2015 3rd International Conference on Future Internet of Things and Cloud

ALMANAC: Internet of Things for Smart Cities

Dario Bonino*, Maria Teresa Delgado Alizo*, Alexandre Alapetite†, Thomas Gilbert†,
Mathias Axling‡, Helene Udsen§, Jose Angel Carvajal Soto¶, Maurizio Spirito*
* Istituto Superiore Mario Boella
Turin, Italy
{bonino,delgadoalizo,spirito}@ismb.it
† Alexandra Instituttet
Copenhagen, Denmark
{alexandre.alapetite,thomas.gilbert}@alexandra.dk
‡ CNet Svenska AB
Stockholm, Sweden
mathias.axling@cnet.se
¶ Fraunhofer FIT
Bonn, Germany
jose.angel.carvajal.soto@fit.fraunhofer.de
§ In-JeT Aps
Copenhagen, Denmark
heu@in-jet.dk

***Abstract*—Smart cities advocate future environments where sensor pervasiveness, data delivery and exchange, and information mash-up enable better support of every aspect of (social) life in human settlements. As this vision matures, evolves and is shaped against several application scenarios, and adoption perspectives, a common need for scalable, pervasive, flexible and replicable infrastructures emerges. Such a need is currently fostering new design efforts to grant performance, reuse and interoperability while avoiding knowledge silos typical of early efforts on similar topis, e.g. automation in buildings and homes. This paper introduces a federated smart city platform (SCP) developed in the context of the ALMANAC FP7 EU project and discusses lessons learned during the first experimental application of the platform to a smart waste management scenario in a medium-sized, European city. The ALMANAC SCP aims to integrate Internet of Things (IoT), capillary networks and metro access networks to offer smart services to the citizens, and thus enable Smart City processes. The key element of the SCP is a middleware supporting semantic interoperability of heterogeneous resources, devices, services and data management. The platform is built upon a dynamic federation of private and public networks, while supporting end-to-end security and privacy. Furthermore, it also enables the integration of services that, although being natively external to the platform itself, allow enriching the set of data and information used by the Smart City applications supported.**

I. Introduction

Worldwide interest in Smart Cities is strong and increasing, fostered by the need to find effective solutions to the major challenges foreseen for the next years, which are expected to strongly affect the current urban landscapes. The main economic and social challenges to be addressed cover: climate change, energy supply and demand, scarcity of fuel and natural resources, population growth, healthcare, urbanization, food traceability and safety, decline of the natural ecosystem, etc. In such a context, cities play an increasingly important role, as they are currently inhabited by nearly half the world's population [1] (68% in Europe[1]), consume 80% of the world's energy production and roughly produce 70% of the total carbon dioxide [2]. The efforts to facilitate viable living conditions are increasing [3] accordingly, and they are attracting major attention from both research and industries.

According to Pike Research [4] a smart city is the integration of technology into a strategic approach to sustainability, citizen well-being, and economic development. Therefore, viable smart city models must be multi-dimensional, encompassing different aspects of smartness and stressing the importance of integration and interaction across multiple domains. Solutions are urgently needed, and quickly advancing technologies may just be the answer. In fact, computer-science solutions and technologies have the potential, to affect most aspects of city ecosystems, from environment (waste management, transportation, governance of natural resources and production of energy) to social (education, security, urban planning, housing) integration.

The ALMANAC project, funded by the European Commission, aims at designing and developing an innovative service delivery platform, with corresponding technology solutions, that integrates typical Internet of Things (IoT) edge networks (termed capillary networks) with metropolitan networks thus enabling an integrated Smart City Information System for green and sustainable applications.

This paper introduces the initial design and experimentation of a new Smart City Platform (SCP), showing how its modules enable the collection, aggregation, and analysis of live data, from appliances, sensors and actuators (e.g., smart meters), etc., regardless of the specific technologies and communication paradigms adopted in each platform deployment. The proposed platform supports policy definition, for

[1] According to Eurostat, http://ec.europa.eu/eurostat/documents/3217494/5728777/KS-HA-11-001-EN.PDF, last visited on March 5, 2015.

978-1-4673-8103-1/15 $31.00 © 2015 IEEE
DOI 10.1109/FiCloud.2015.32

IEEE computer society

addressing federation and data access issues, and implements intelligent control of city processes and devices, through capillary networks integrated with M2M (Machine-to-Machine) management platforms. Data natively fed to the the platform, i.e., collected from sensors and devices specifically deployed as part of the city "smartification" process are integrated, mashed-up and put in context with externally generated data sets, including existing open data initiatives, unstructured social data, etc.

Preliminary results show that the proposed architecture is able to address the huge amount of data generated by a smart city and the high number of sensors deployed within. Moreover, semantics-based city modelling employed in the platform enables addressing several domains, from smart waste management to water control, and seamlessly linking of data and models with other similar initiatives distributed worldwide.

The remainder of the paper is organized as follows: Section II provides an overview of the current state of the art of smart city IoT platforms, and technologies, and discusses the existing approaches with respect to the one adopted in the ALMANAC platform definition. Section III introduces the ALMANAC platform architecture starting from a general, functional, overview and continuing with design choices and specific details characterizing the platform design. Section IV presents the platform integration and deployment in a context of federated smart cities exchanging data and functionalities, according to specified agreements. In this respect, the paper concentrates more on technical aspects, as regulatory choices and policies are clearly out of scope. Finally, Section V reports preliminary results, proving both the viablity of the approach and the initial scalability of the platform, while Section VI draws conclusions and discusses future works to be carried in the next year of the project.

II. Related Works

Over the last few years, the research community has reached a quite clear consensus about the necessity and usefulness of making cities smarter, and many studies have been carried out on market potential [5],possible architectures (e.g., I-architecture [6], IoT-A [7]) and deployments, and roadmaps for real world exploitation [8], among others. Nevertheless, smart city research is still a very active field, with currently available solutions still are still experiencing difficulties demonstrating the full potential of such a new generation of human settlements.

To address the issues, and challenges emerging in this domain, researchers from industry and academia are proposing approaches and platforms for supporting the "smartification" of existing cities. On the industrial side, big firms such as IBM [9], [10], HP, BOSCH, Siemens and Cisco are currently pursuing Smart City solutions with the aim of enabling cities to become "smarter" by intelligently connecting city events and data, in each city jurisdiction, possibly exploiting integrated cloud solutions.

On the research side, several approaches are available addressing different aspects of smart city design and exploitation, from sustainable deployment and growth to next-generation, semantics-powered smart city platforms. In their paper, Vilajosana et al. [11] investigate the underlying reasons for the current slow adoption of smart cities, proposing a procedure to make smart cities happen based on big data exploitation through the so-called "API stores" concept. They discuss a viable approach for scaling business within city ecosystems and describe the currently available ICT technologies, exemplifying all findings by means of a sustainable smart city application. While the approach proposed by Vilajosana et al. can be considered to describe the motivations for Smart Cities and establish a general framework, the proposed ALMANAC framework is more oriented to the actual implementation of a federated smart city platform.

Bellini et al. [12], propose a system for data ingestion and reconciliation of smart city related aspects such as: road graphs and services, traffic sensors, etc. The system allows managing a large volume of static and dynamic data coming from a variety of sources. This data is mapped to a smart-city ontology, called KM4City (Knowledge Model for City), and stored into an RDF store where it is available for applications via SPARQL queries, thus providing new services to the users via specific applications for public administration and/or enterprises. Examples of potential uses of the knowledge base produced by the Bellini et al. system are also offered, and they are accessible from the RDF store and related services. The framework and platform architecture presented in this paper shares several design principles with the approach of KM4City, i.e., semantic modeling of smart-city entities and big-data approach to data capture. However, while the former is more focused on ontology and dataset population, ALMANAC is more concerned with federation of several city platforms and direct exploitation of platform services by third-party application developers.

Several joint research projects are currently investigating aspects related to smart city design and adoption both in Europe and US, among them:

- The Smart Santander project[2] provides a city-scale experimental research facility in support of typical applications and services for a smart city. Current ALMANAC solutions already integrate data managed by Smart Santander thanks to a strong collaboration between the two projects;
- The UrbanWater project[3], is installing smart meters in Almería (Spain) to achieve greater efficiencies in water management;
- The Open IoT project[4] defines an open source cloud solution for the Internet of Things, which can be seen as a technological enabler for smart city platforms such the one presented in this paper;
- The Mobosens US research project[5] provides citizens with a platform for collecting and sharing environmental data, from stream quality to drinking water safety.

III. Architecture

The overall logical architecture of the ALMANAC smart city framework is purposely kept general to be easily adaptable to any city. It includes functional blocks well aligned with

[2]http://www.smartsantander.eu/, last visited on March, 2015.
[3]http://urbanwater-ict.eu/, last visited on March, 2015.
[4]http://openiot.eu/, last visited on March, 2015.
[5]http://nanobionics.mntl.illinois.edu/mobosens/, last visited on March, 2015.

the IoT-A reference architecture specification [7] and mainly addresses four different concerns:

(a) Enabling external applications to exploit functions and services offered by the platform (e.g., through cloud-based REST APIs);
(b) Enabling efficient handling of high-cardinality, high-frequency event data, context data and metadata associated to managed entities (e.g., sensors);
(c) Mapping low-level data representations and communication paradigms into a common, shared and machine understandable set of models, and data-exchange paradigms, and, finally,
(d) Supporting effective communication inside and outside the platform.

This high-level specification is materialized into a platform architecture organized around four layers (see Figure 1): the API, Virtualization, Data Management and Smart City Resource Adaptation.

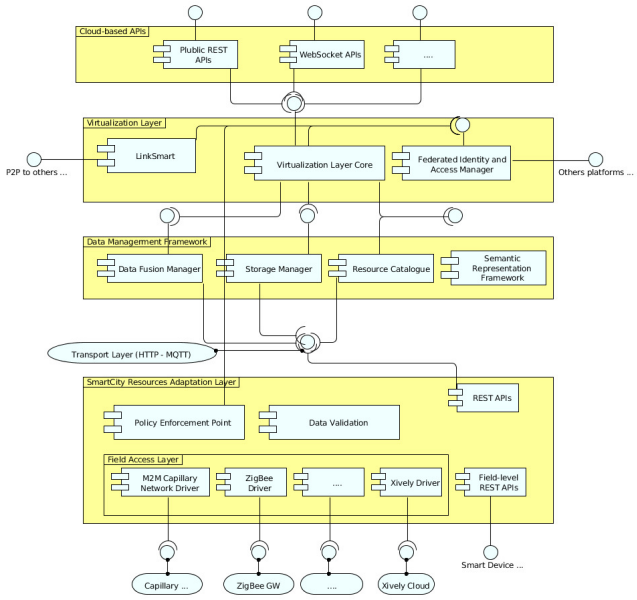

Fig. 1. The ALMANAC Smart City Platform architecture

A. API

The API layer encompasses modules offering endpoints for third party applications to integrate services provided by the ALMANAC platform. Currently included modules comprise: a RESTful API endpoint for client/server interactions, e.g., searching for specific device types; and a WebSocket API for enabling effective management and delivery of live data streams, e.g., measures taken at the city premises. Access to both is subject to authentication and authorization and might involve role and policy sharing between different platform instances (see Section IV).

For example, functions exposed through the API layer include:

- Semantic Library services, allowing external applications to search for IoT resources (e.g., devices) based on specific metadata requirements, e.g., by resource type (ontology classes), by location, by application domain (e.g., waste management), etc.
- Historical Data services, offering access to time series of events and measures associated to IoT resources managed by the platform (or by other federated platform instances). Such resources can be either "direct", i.e., corresponding to a tangible service or device deployed in the smart city, or "derived" as result of data-fusion and complex event processing.
- Data fusion services enabling applications to define their own operators and queries using complex event processing languages (e.g., the Esper EPL[6]). These operators can be applied to process both local data and information available at the federation level, i.e., produced in different platform instances.
- Resource services enabling direct control of connected physical devices and platforms; mainly for querying data and measures, but also actuation is supported.
- Provisioning and management, allowing, for example, to programmatically add new objects to the set of entities interfaced and abstracted by a given platform instance.

B. Virtualization

The Virtualization layer represents the highest level of an ALMANAC platform instance and hosts modules coordinating the activities of the core platform components. Inter-platform communication, secure handling of data and enforcement of access policies is enabled by this layer, both at the platform and at the federation level. Involved modules encompass: the *Virtualization Layer Core*, *LinkSmart*, the *Federation Identity* and the *Access Manager*.

The *Virtualization Layer Core* (VLC) coordinates functions and services offered by the core platform components. It virtualizes several resources and processes by means of proxying, routing, wiring, and payload transformation services. In particular, the VLC: (a) proxies requests and responses (as well as event streams) to/from different internal modules, acting as front-end towards the API layer; (b) routes incoming data and outgoing information through the proper modules be they internal or belonging to other federated platforms; (c) wires and coordinates the activities of different core modules with the aim of effectively fulfilling complex application requests, e.g., involving a directory search, a data fusion query and an historical data extraction; (d) transforms requests/responses payload formats to better support interoperability with different systems and smart city platforms. Translation services depend extensively on ontology models and metadata representations managed by the platform semantic modules.

The *LinkSmart* module is based on the well-known, and widely adopted (at EU-level), LinkSmart [13] middleware. It ensures inter-platform connectivity through peer-to-peer data exchange, and seamlessly exposes remote platform services as part of the local platform. Moreover, it enforces trust and security between platforms thanks to a standard Policy Enforcement Point built into the Federated Identity and Access Manager of the ALMANAC platform .

[6]EsperTech: Esper - Reference Documentation, http://esper.codehaus.org/esper-5.1.0/doc/reference/en-US/html/index.html, last visited on February 27th, 2015

The *Federated Identity and Access Manager* (FIAM) is the module in charge of implementing, exposing and, enforcing trust and security inside both single platforms and platform federations. Inter-platform communication is secured through standard TLS encryption whereas access policies are enforced at the platform boundaries; reference standards include the OASIS Security Assertion Markup Language, SAML [14] and the eXtensible Access Control Markup Language (XACML) [15]. The FIAM module builds upon two main sub-modules, namely the Federated Identity Manager and the Access Manager.

The Federated Identity manager (FIM) enables federation among several ALMANAC Platform Instances, and with other platforms (e.g. the Smart Santander Platform) by providing the following features:

1) Web single sign-on, based on the SAML standard so that users only have to sign into the federation once;
2) Dynamic retrieval of user attributes among domains to facilitate access control decisions, thus enabling unified access among the federation.

Together with the Federated Identity Manager, the Access Manager (AM) centrally manages authentication and authorization policies for different types of ALMANAC users and resources. The AM provides the following features:

1) Policy-based access control, ensuring that only authorized users can access certain data based on their access rights, possibly across multiple applications and domains;
2) Centralized policy management, reducing the complexity associated with managing authentication and authorization policies separately across multiple ALMANAC Platform Instances or internally across modules.

C. Data Management

The Data Management layer encompasses components for storing, retrieving and managing both (semantic) metadata and live data gathered from the smart city. Data filtering, aggregation and fusion of measurements gathered from sensors deployed in the city territory are performed at this level, thanks to the so-called Data Fusion Manager module (DFM). Such data is then stored and made accessible to applications through the higher ALMANAC layers. Resources deployed in the city can be retrieved and queried thanks to the joint work of the Resource Catalogue, which effectively handles descriptions of "real" devices and systems (instances), and of the Semantic Representation Framework, which provides metadata and context in widely adopted Semantic Web standards (e.g., JSON-LD and OWL) supporting SPARQL-based CRUD operations. These components are further detailed in the following paragraphs.

The *Data Fusion Manager* (DFM) supports processing of live data and events (i.e., Complex Event Processing [16] operations) fed by the underlying SCRAL layer (see Section III-D) or generated by other platform components (e.g., as a result of a Data Fusion process). It can exploit several event processing backends (e.g., Esper, WSO2, StreamInsight) and offers a uniform operation definition language (Data Fusion Language - DFL) to support easy definition of CEP construcs at the API level. With respect to standard EPL languages, the DFL adopted in this module offers roughly the same expressivity, as it bridges / exposes the features of the underlying event-processing backends, but offers additional support to composition of complex event elaboration chains (e.g., similarly to [17]), possibly defined through visual tools. Figure 2 shows the DFM inner architecture including the processing backends and the DFL interpreter.

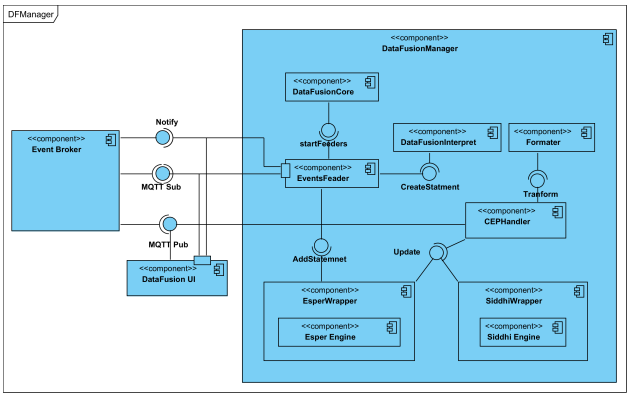

Fig. 2. The Data Fusion Manager architecture

The *Storage Manager* module provides persistence for data generated by devices and systems deployed in the "managed" smart city (raw data) and by the Data Fusion Manager (derived or fused data). It supports storage and querying of large numbers of time-stamped events, and measurements, and exploits several different storage backends depending on specific installation requirements. This module adopts the microservices[7] design pattern to ensure both horizontal and vertical scalability. In this sense, it basically acts as a transparent proxy towards groups of storage solutions, whose specific technology and deployment (e.g., local vs cloud) may be different for any ALMANAC platform instance. The corresponding logical architecture is shown in Figure 3, and includes sub-modules for: de-coupling high-frequency feeds from inner (and slower) storage operations and for adapting different input formats to its inner data model (NoSQL is the default storage-level selection, but relational schemas can be adopted).

The *Resource Catalogue* provides a "directory" service for devices and systems managed by a single ALMANAC platform instance. It offers a REST-based interface to select and retrieve resources and services, with filtering capabilities. Every resource is described according to the IoT-A Architectural Reference Model [7] and represented through an extension of the standard UPnP SCPD (Service Control Point Document) format[8]. Resource descriptions can either

[7]http://microservices.io/patterns/microservices.html, last visited on March 3rd, 2015.

[8]Currently, an on-going effort is pushing forward the adoption of widely recognized sensor representation formats, e.g., the OGC SensorML or the IETF SenML for easier data exchange.

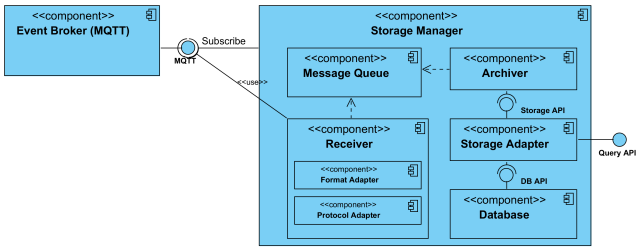

Fig. 3. The Storage Manager inner architecture.

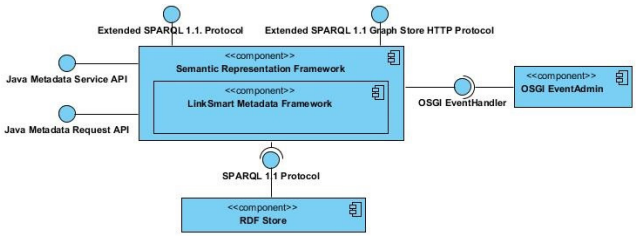

Fig. 4. The Semantic Representation Framework inner architecture.

TABLE I. SAMPLE QUERIES TO THE RESOURCE CATALOGUE.

Query	Description
`/*`	List all available resources
`//IoT:gateway[.='gateway1']`	List all available resources at the gateway 1
`//IoT:gateway[.='gateway1']/services/*`	Lists all the services offered by gateway 1

be provided by the resource abstraction layer (SCRAL) or by the provisioning API. In both cases, supported discovery paradigms encompass: explicit registration, via suitable REST calls, or implicit discovery carried out locally through UPnP. It must be noted that since the latter process cannot go beyond the local network boundaries, it cannot be exploited to discover real-devices deployed in a given Smart City. Instead, it is leveraged to dynamically handle their "software counterparts" exposed by the platform modules handling physical networks (e.g., by the SCRAL). Table I reports some sample queries to the catalogue and describes the corresponding operations.

The *Semantic Representation Framework* is responsible for managing and querying structured, graph-based, domain models and meta-data. It exposes a range of native (Java) and remote (HTTP) service interfaces, allowing for various interaction modes that are either focusing on invocation of persistent queries (Remote Procedure Calls, RPC) or on exchange of semantic data resources (REST). In the context of the ALMANAC architecture, the SRF logically complements the Storage Manager component. While the former excels in querying of structure-rich data, enhanced by semantic inference, the latter is optimized for bulk storage and querying of large volumes of measurement data and events.

The SRF core modules are developed as part of the open-source LinkSmart Metadata Framework. The Metadata framework acts as a transparent proxy to any state-of the art RDF store compatible with the SPARQL 1.1 Protocol and considerably augments the standard repository (e.g. Virtuoso[9]) functions and interface coverage, e.g. by supporting persistent SPARQL queries and updates and, RESTful management of graph fragments (semantic resources). Figure 4 provides an overview of the inner framework architecture.

[9]http://virtuoso.openlinksw.com/, last visited on March 3rd, 2015.

D. Smart City Resource Adaptation

The Smart City Resource Adaptation Layer (SCRAL) provides a REST-based uniform and transparent access to physical devices, capillary networks, systems and services for monitoring and actuation in a Smart City context. Peculiar device functionalities are uniformed, abstracted and mapped to a well-known set of functions and primitives complying with (device) models handled in the semantic framework of the ALMANAC platform. Moreover, due to its nature of interface between the ALMANAC platform and the real world, the SCRAL offers primitives for applying access-control, data-validation and role-based policies on field-level data sources. While typical SCRAL instances are distributed near to physical devices, meaning that more than one SCRAL instance is usually adopted in a single ALMANAC Platform Instance (PI), at least one cloud instance is typically available in a PI to support connection of smart devices, i.e., of devices able to natively exchange data conforming to the ALMANAC data model. In such a case, the main SCRAL duty is to enforce access rights, perform data validation and, support needed provisioning primitives.

The SCRAL inner architecture (see Figure 5) is mainly organized in three layers named API, Core and Field-Access, respectively. The topmost layer exploits the SCRAL Connector component which exposes REST resources to the upper layers of the ALMANAC platform, and the MQTT data source, which feeds the ALMANAC PI broker with purposely uniformed, abstracted and validated real-time data. The core layer hosts components such as the Event-Delivery module, the Policy Enforcement Point, the Data Validation module and the Metadata Generation module. Finally, the Field Access layer integrates components (drivers) to wrap and isolate device-specific and technology-specific implementations used to access real physical devices and systems.

IV. FEDERATION

ALMANAC defines a framework in which multiple PIs can participate in federations offering cross-city, cross-nation and cross-entity services. Such federations are typically built around service-exchange agreements that consider the involved political and legal issues. From a technical standpoint, they are deployed by establishing sets of trust domains including two or more PIs. Platform instances exchange data and distribute tasks according to roles and privileges defined in the federation agreement (e.g., in a simplistic on/off case, all users/platforms belonging to the federation are allowed to perform tasks, whereas external platforms/users cannot exploit the federated services), and managed through state-of-the-art solutions for

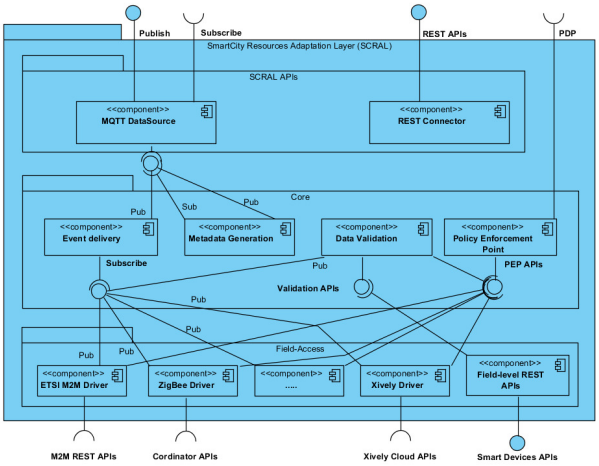

Fig. 5. The SCRAL inner architecture.

federated identity and role management, for example based on the SAML specification. Data and tasks are exchanged in a peer-to-peer manner, either through standard Internet protocols or through LinkSmart-controlled channels, hence overcoming address translation issues. Each PI can be part of more than one federation at a time, thus providing the basis for a strictly interconnected deployment of the ALMANAC services across cities and nations (Figure 6 displays a high-level view of such a federation approach).

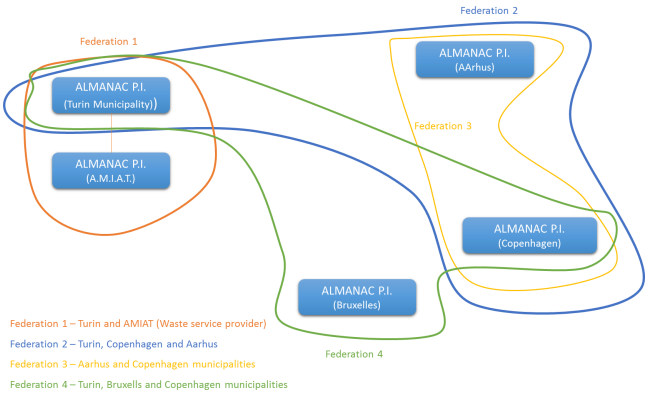

Fig. 6. Multiple federations between ALMANAC Platform Instances.

Routing of requests and data through the different PIs belonging to a federation is managed at the Virtualization Layer, exploiting a sound naming / addressing scheme based on standard DNS names and CNAMEs. In fact, the ALMANAC platform defines a set of guidelines and conventions for naming instances and components within. This scheme is federation centric and adopts a twofold approach:

- Inside a platform instance, components are addressed with Internet domains structured as follows: `component.platform.federation.domain`. For example, the Storage Manager component located in the Copenhagen PI of Federation 4 (see Figure 6) will be identified by the following DNS name: `storage-manager.copenhagen.federation4.almanac.eu` where the almanac.eu part is the authoritative domain, which may change depending on the federation owner.
- Outside of a PI, i.e., for external applications exploiting the ALMANAC services, components will be identified as follows: `platform.federation.domain/component`, thus hiding the platform complexity to external stakeholders.

A. Deployment

While being mainly involved with PIs and their interrelations, the deployment of ALMANAC also tackles the issues and design decisions related to deployment of components inside a single PI. Two main design choices govern the platform design in this context:

Firstly, it is assumed that not all components defined in the ALMANAC platform specification must be part of a specific platform instance deployed in the real world. This accounts for different requirements depending on the deployment purpose (legacy instance or public utility installation), context (small city, utility, large administrative regions), and so on. For instance, under these conditions it is perfectly possible that single, isolated PIs do not include the LinkSmart connection modules, whereas PIs deployed on low-power devices might not include full versions of the Data Fusion Manager.

Secondly, components participating in a single platform instance do not need to be singletons nor to be physically deployed on the same server in the same location. Thanks to the DNS-based naming scheme discussed in the previous paragraphs, components can in fact be safely identified, together with the platform instance they belong to, thus moving the loosely coupled architecture of ALMANAC a step further towards a completely distributed approach to Smart City Platforms.

V. Experimental Results

The ALMANAC platform has been developed as a joint collaboration of seven international partners participating in the homonymous project funded by the EC. It exploits several different technologies including Java, OSGi, C#, NodeJS, M2M protocols, non-relational databases, etc. Each module has been carefully designed and implemented according to the relative, specific requirements, exploiting different technical solutions and paradigms depending upon its respective roles, and tasks, in the platform. Early versions of the ALMANAC Platform have been tested in two hackathons held in 2014 at the IoTWeek (in London) and at the IoT360 conference, in Rome. An active deployment[10] is currently under test (see Figure 7), demonstrating a quite good reliability and fault tolerance, with over 5 months of up-time with no intervention. The platform instance used for experiments is focused on the waste collection domain and exploits a light-weight ontology model (and instances) developed in the context of the project.

This ontology focuses on waste-related issues with an open, extensible and scalable approach. According to this design

[10]reachable at http://almanac-showcase.ismb.it:8000/

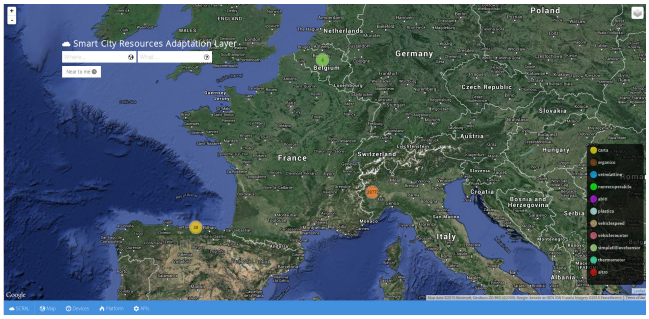

Fig. 7. The home page of the test platform instance showing managed sensors.

goal, no assumption is performed on the remaining smart city data, and modelling is restricted to the aspects specifically related to waste collection and management, e.g., by representing waste bins, type of disposables generated by the citizenship and so on. All represented concepts leverage existing, well-known definitions of properties (and classes/super-classes) with a typical Linked Open Data approach. For example, the city concept is directly linked (owl:equivalentClass) to the corresponding schema[11] and places[12] concepts. The current ontology version mainly represents waste bins, waste types and the geographical/administrative context in which they are deployed. It is organized along three main hierarchies (see Figure 8) rooted in the classes WasteBin, City and Waste, respectively.

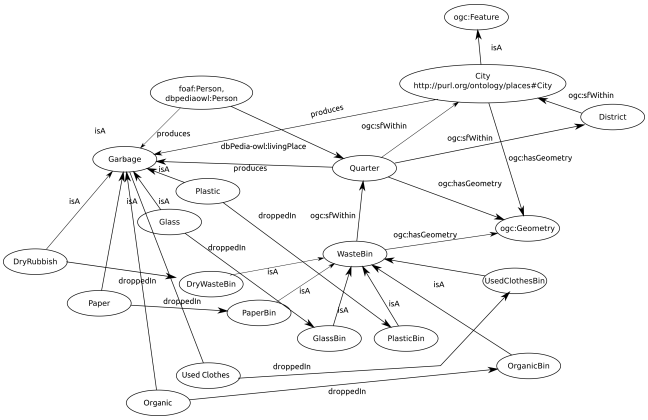

Fig. 8. The ALMANAC waste ontology exploited in the preliminary test deployment.

From a more "technical" standpoint, the ontology includes 20 concepts, 6 object properties and 3 data-type properties. Featuring a SHIQ(D) DL expressivity, it is interconnected with 6 different and widely recognized vocabularies including: the geonames ontology[13], the places ontology, the GeoSPARQL vocabulary [18], the Schema.org vocabulary, the VCard [19], Good Relations [20] and the Unit of Measurement (MUO) [14] ontologies. The current ALMANAC deployment exploits a full instantiation of the waste ontology representing waste bins available in Turin. It includes around 28,000 waste bins, one city, 10 districts and 25 quarters. Additional sample sensors from other cities are included, specifically Santander (ES) and Bruxelles (BE). This increases the number of handled sensors to around 30,000.

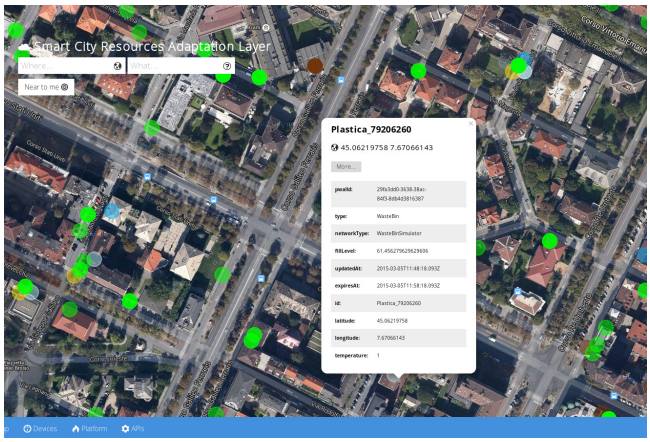

Fig. 9. A snapshot of the ALMANAC platform interface showing the details of a managed waste bin.

These sensors produce data at a rate varying from one sample every 10 minutes (fill level of waste bins) to 1 sample every minute for speed sensors and transit counters. The overall ALMANAC infrastructure demonstrates ability to gracefully sustain such high sensor cardinality and data rates, even when storing raw data, without performing any aggregation.

VI. Conclusions and Future Works

In this paper we presented the current architecture of the ALMANAC Smart City Platform being designed and developed as part of the EU FP7 ALMANAC project. Differently from most of the current platforms, ALMANAC is among the few currently available approaches to smart cities where federation between different administrations and companies, trust, security and data sharing are explicitly tackled as core features. Project activities in the last year lead to a first working prototype of the platform which, given the first experimentation outcomes, has the potential to address several challenges of the Internet of Things in the smart city domain. Future works will include: (a) stronger federation capabilities and systematic addressing of related trust and service agreement issues, (b) self-describing, machine-understandable API design to foster easier adoption of the platform, (c) support tools for spreading the platform among EU cities and potentially worldwide, (d) citizen-centric applications support, to involve the citizenship in virtuous cycles that, thanks to the platform features, should trigger tangible improvements in the city quality of life, with a particular focus on the waste and water management domains.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n.609081.

[11] https://schema.org/, last visited on March, 2015

[12] http://vocab.org/places/schema.html, last visited on March, 2015

[13] http://www.geonames.org/ontology/documentation.html, last visited on March, 2015

[14] http://idi.fundacionctic.org/muo/, last visited on March, 2015

REFERENCES

[1] "State of world population 2007, unleashing the potential of urban growth," U.N. Population Fund (UNFPA), Tech. Rep. [Online]. Available: http://www.unfpa.org/public/publi-cations/pid/408

[2] "Sustainable smart cities - building sustainable business value in changing cities," KPMG Asset Management Competence Centre, Tech. Rep., 2012.

[3] M. Kinver. (2006, June) The Challenges Facing an Urban World. online-article. BBC News website. [Online]. Available: http://news.bbc.co.uk/2/hi/science/nature/5054052.stm

[4] "Smart cities. intelligent information and communications technology infrastructure in the gov- ernment, buildings, transport, and utility domains." Pike Research, Tech. Rep., 2011. [Online]. Available: http://www.navigantresearch.com/research/smart-cities

[5] "Smart Cities Market by Smart Home, Intelligent Building Automation, Energy Management, Smart Healthcare, Smart Education, Smart Water, Smart Transportation, Smart Security, & by Services - Worldwide Market Forecasts and Analysis (2014 - 2019)," marketsandmarkets.com, Tech. Rep., 2015. [Online]. Available: http://www.marketsandmarkets.com/Market-Reports/smart-cities-market-542.html

[6] "ICT architecture supporting daily life in three smart cities," Smart Cities EU Project, Tech. Rep., 2012. [Online]. Available: http://www.smartcities.info/files/ICT_architecture_supporting_service_delivery_in_Smart_Cities.pdf

[7] M. Bauer and al., "Final architectural reference model for the IoT v3.0," Internet of Things Architecture IoT-A, Tech. Rep., 2013. [Online]. Available: http://www.iot-a.eu/public/public-documents/d1.5/

[8] H. V. K. K. M. T. e. a. O. Vermesan, M. Harrison, "The Internet of Things - Strategic Research Roadmap", Cluster of European Research Projects on the Internet of Things," 2009.

[9] P. Fritz, M. Kehoe, and J. Kwan, "Ibm smarter city solutions on cloud," IBM, White Paper, 2012. [Online]. Available: http://www01.ibm.com/software/industry/smartercities-on-cloud

[10] J. Hogan and al., "Using standards to enable the transformation to smarter cities," *IBM Journal of Research and Development*, vol. 55, no. 12, p. 110, Jan - Mar 2011.

[11] I. Vilajosana, J. Llosa, B. Martinez, M. Domingo-Prieto, A. Angles, and X. Vilajosana, "Bootstrapping smart cities through a self-sustainable model based on big data flows," *Communications Magazine, IEEE*, vol. 51, no. 6, p. 128134, June 2013.

[12] P. Bellini, M. Benigni, R. Billero, P. Nesi, and N. Rauch, "Km4city ontology building vs data harvesting and cleaning for smart-city services," *Journal of Visual Languages & Computing*, vol. 25, no. 6, p. 827839, 2014, distributed Multimedia Systems {DMS2014} Part I. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1045926X14001165

[13] M. Eisenhauer, M. Jahn, F. Pramudianto, T. Sabol, P. Kostelnik, and J. Hreno, "Towards a generic middleware for developing ambient intelligence applications," in *CIB W78-W102 International Conference, Sophia Antipolis, France*, October 2011.

[14] *OASIS Security Assertion Markup Language (SAML) V2.0*, http://saml.xml.org/saml-specifications, OASIS Std., Rev. 2.0, March 2005.

[15] *eXtensible Access Control Markup Language (XACML) Version 2.0*, http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf, OASIS Std., Rev. 2.0, February 2005.

[16] D. C. Luckham, *The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2001.

[17] D. Bonino, F. Corno, and L. De Russis, "Real-time big data processing for domain experts, an application to smart buildings," in *Big Data Computing / Rajendra Akerkar*. Boca Raton: CRC Press, 2013, vol. 1, p. 415447. [Online]. Available: http://porto.polito.it/2504150/

[18] M. Perry and J. Herring, "Ogc GeoSPARQL - A Geographic Query Language for RDF Data," Open Geospatial Consortium, Tech. Rep., 2012.

[19] P. S., *vCard Format Specification*, Internet Engineering Task Force (IETF) Std. [Online]. Available: http://tools.ietf.org/html/rfc6350

[20] M. Hepp, "GoodRelations: an ontology for describing products and services offers on the web," in *Proceeedings of 16th International Conference on Knowledge Engineering and Knowledge Management (EKAW2008)*, vol. 5268. Springer LNCS, September 2008, p. 332347.