Open IoT Ecosystem for Sporting Event Management

SYLVAIN KUBLER[1], JÉRÉMY ROBERT[1], AHMED HEFNAWY[2], KARY FRÄMLING[3], CHANTAL CHERIFI[2], AND ABDELAZIZ BOURAS[4], (Member, IEEE)

[1]Interdisciplinary Centre for Security, Reliability and Trust, University of Luxembourg, L-2721 Luxembourg City, Luxembourg
[2]DISP Lab, Université Lyon 2, F-69621 Villeurbanne, Lyon, France
[3]School of Science, Aalto University, FI-00076 Aalto, Finland
[4]CSE, College of Engineering, Qatar University, Doha 2713, Qatar

Corresponding author: Sylvain Kubler (sylvain.kubler@uni.lu)

ABSTRACT By connecting devices, people, vehicles, and infrastructures everywhere in a city, governments and their partners can improve community well-being and other economic and financial aspects (e.g., cost and energy savings). Nonetheless, smart cities are complex ecosystems that comprise many different stakeholders (network operators, managed service providers, logistic centers, and so on), who must work together to provide the best services and unlock the commercial potential of the so-called Internet of Things (IoT). This is one of the major challenges that faces today's smart city movement, and the emerging "API economy." Indeed, while new smart connected objects hit the market every day, they mostly feed "vertical silos" (e.g., vertical apps, siloed apps, and so on) that are closed to the rest of the IoT, thus hampering developers to produce new added value across multiple platforms and/or application domains. Within this context, the contribution of this paper is twofold: 1) present the strategic vision and ambition of the EU to overcome this critical vertical silos' issue and 2) introduce the first building blocks underlying an open IoT ecosystem developed as part of an EU (Horizon 2020) Project and a joint project initiative (IoT-EPI). The practicability of this ecosystem, along with a performance analysis, is carried out considering a proof-of-concept for enhanced sporting event management in the context of the forthcoming FIFA World Cup 2022 in Qatar.

INDEX TERMS Internet of Things, smart city, interoperability, ecosystem, open innovation, API economy.

I. INTRODUCTION

New Internet of Things (IoT) applications that leverage ubiquitous connectivity, system interoperability and analytics, are enabling Smart City initiatives all over the world [1], [2]. These new applications introduce tremendous new capabilities such as the ability to connect, manage, and optimize complex sets of disparate information systems, sensors, devices, people and software solutions into a "System-of-Systems" (SoS) like fashion [3], [4].

Although the smart city paradigm paves the way for societal and economic opportunities, for example to reduce costs for societies, increase the service for the citizens in a number of areas, foster a sustainable economic growth, they also pose architectural and structural issues that must be addressed for businesses to benefit [5], [6]. One of the most critical obstacles is the *vertical silos'* model that shapes today's IoT, which hampers developers – *due to the lack of interoperability and*

openness – to produce new added value across multiple platforms (data is "siloed" in a unique system, cloud, domain, and stays there) [7], [8]. This is all the more true in a smart city environment, as it is a complex ecosystem that comprises a wide range of interacting and cooperating actors such as users, software and network providers, financial institutions, logistic centers, and so on [9], [10], which is why cities are usually built on vertically-oriented closed systems that are difficult to interconnect.

Several organisms and standardization fora understood this critical challenge and started to build up consortia and IoT initiatives to address it. Let us cite, for example, the *Web of Things* initiative at W3C that aims to create open ecosystems based upon open standards, including identification, discovery and interoperation of services across platforms [11]; the *Alliance for Internet of Things Innovation* (AIOTI) launched by the EU with the aim of strengthening links and

building new relationships between the different IoT players (industries, SMEs, startups) [12]; the *Open Platform 3.0*^{TM} at The Open Group that focuses more on organization applications and practices [13]; the *OneM2M* global standards initiative that involves eight standard bodies for Machine to Machine (M2M) communications [14]; the *IEEE Internet of Things (IoT)* initiative [15] or still the *International Technical Working Group on IoT-Enabled Smart City Framework* developed at NIST [16]. Although most of those initiatives promote various types of standards and specific technology enablers, they all share the same vision about relying as much as possible on open and interoperable standards to foster emergence of open ecosystems, and unlock the commercial potential of the IoT.

Within this context, the research work presented in this paper aims to present one framework that enables IoT service stakeholders (either service publishers or consumers) to join, contribute and benefit from an open IoT ecosystem developed in the context of an ongoing EU H2020 project named bIoTope.[1] This project contributes both to the AIOTI initiative and a joint project initiative named IoT-EPI[2] (European Initiative for IoT platform development) that aims to build a vibrant and sustainable IoT-ecosystem in Europe. Our research work is also part of the Open Platform 3.0 initiative since the messaging protocols used in bIoTope are the ones published by The Open Group, as will be discussed in this paper. Section II provides a more detailed view on such initiatives, especially with regard to EU's vision and ambition. Following this, section III provides a first overview of the building blocks underlying the bIoTope's ecosystem. Section IV develops and evaluates the practicality of these technical building blocks with regard to a sporting event management scenario defined in the framework of the forthcoming FIFA World Cup 2022; conclusion and discussion follow.

It is important to note that this article is an extension of the conference paper published in the proceedings of the *13th Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services* [29], whose extended content includes: (i) a more in-depth presentation of the EU initiatives, along with a discussion of how the API Economy could play a key role in solving the problem of "vertical silos" in the IoT, (iii) an introduction of the IoT service marketplace designed in bIoTope, which acts as a Web of Things-like environment (including multimodal search and discovery IoT data/services), (iv) a presentation of how bIoTope fosters easy service composition using IoT visual programming tool.

II. EUROPEAN IoT VISION & AMBITION
A. TOWARDS OPEN IoT ECOSYSTEMS
While in the US, IoT ecosystems are created around big, multinational players such as Google, Amazon, Facebook

and Apple – *the so-called GAFA [30]* – the EU's strength is rather in smaller and agile companies. Several past EU initiatives gave rise to a multitude of IoT platforms in various domains [31], let us cite the IERC cluster in which IoT-A, OpenIoT, BUTLER, *etc.*, were developed, or still the Future Internet-PPP programme that contributed to the development of the FI-WARE cloud-based infrastructure that offers a number of general- and specific-purpose functions in multiple sectors (farming, manufacturing, mobility, pervasive game...). All these projects/platforms were funded and developed in the FP7 framework (2007-2015) that constituted the ignition phase of the IoT program approach. The second phase - *started in 2016* – aims to foster the emergence of open IoT (or business) ecosystems enabling and incentivizing communities of citizens, SMEs, and other public-private institutions, to join and contribute to the growth and sustainability of these ecosystems. To achieve this mission, the EU recently launched the AIOTI alliance with the aim of making recommendations for future collaborative work in the context of the IoT Focus Area in the H2020 EU program. TABLE 1 provides a short overview of the 13 Work Group (WG) composing the AIOTI alliance, their respective focus area, and recent report(s)/white paper(s) that have been published by each WG.

Within this alliance, and as part of WG1, seven Research and Innovation (R&I) projects funded under the ICT30 cluster (2016-2019) have undertaken research technology developments with the aim to turn the above-mentioned platforms, and other IoT solutions developed at the european and international levels, into economically viable IoT ecosystems. One particularity of this cluster, compared with previous EU initiatives, is that these seven R&I projects are part of the joint IoT-EPI project initiative, aiming to maximize opportunities for platform development, interoperability and information sharing across projects and use case pilots. This initiative and underlying projects are discussed in the next section.

B. IoT-EPI: A JOINT PROJECT INITIATIVE TO FOSTER A VIBRANT AND SUSTAINABLE IoT-ECOSYSTEM IN EUROPE
The seven R&I projects carried out in the ICT30 cluster aims to improve horizontal interoperability and provide viable proofs-of-concept about how existing platforms for connected smart objects can easily, safely, and reliably be integrated for a multiplicity of novel IoT applications. TABLE 2 provides an overview of what are the focus of each project, namely:

- *Integration of devices:* this topic refers to M2M communications capabilities, where turn-key M2M solutions and components are developed and easy to be deployed. For example, *TagItSmart* will develop innovative optical tags (using a new QR code ink technology) and associated Cloud services for enhanced product tracking; *INTER-IoT* and *symbIoTe* aim to use a common M2M service layer specifications (based on the ETSI's oneM2M standard); *AGILE* proposes a gateway

---

[1] http://www.biotope-project.eu, last accessed Apr., 2017.
[2] http://iot-epi.eu, last accessed Apr., 2017.

TABLE 1. AIOTI working groups (WG).

WG	Main Focus	Reports
WG1 - IoT European Research Cluster	Compare EU-funded innovation research and development programmes, with the aim of defining a common vision of IoT technology and addressing european research challenges.	[17]
WG2 - Innovation Ecosystems	Aim at designing actions to develop innovation ecosystems by stimulating startups, encouraging the use of open IoT platforms, enabling Large Scale Pilots (LSPs), and linking large and small companies through open innovation.	[18]
WG3 - IoT Standardisation	Identify and, where appropriate, makes recommendations to address existing IoT standards, analyses gaps in standardisation, and develops strategies and use cases.	[19], [20], [21]
WG4 - Policy Issues	Identify and makes recommendations to address existing and potential barriers that prevent or hamper the take-up of IoT in the context of the Digital Single Market.	[22]
WG5 - Smart Living Environment for Ageing Well	Deliver white papers, recommendation reports, innovative use cases susceptible to improve the quality of life of Elderly people using the latest IoT technologies.	[23]
WG6 - Smart Framing	Deliver reports of use cases that allow monitoring and control of the plant and animal products life cycle "from farm to fork".	[24]
WG7 - Wearables	Focus on IoT solutions that integrate key technologies (e.g. nano/organic electronics, embedded software...) into intelligent systems to bring new functionalities into clothes and other body-mounted devices.	[25]
WG8 - Smart Cities	Refer to IoT solutions that allow for increased multi-modal mobility, more efficient traffic management, a dynamic road infrastructure, automated road tolling, usage based insurance, and improved policy making.	[26]
WG9 - Smart Mobility	Focus on IoT solutions that improve water management efficiency by monitoring and controlling surface water retention, flooding, and so forth.	[27]
WG10 - Smart Water Management	Focus on IoT solutions that bring together information, technology and human ingenuity to achieve a rapid revolution in the development and application of manufacturing intelligence.	–
WG11 - Smart Manufacturing	Focus on IoT solutions that bring together information, technology and human ingenuity to achieve a rapid revolution in the development and application of manufacturing intelligence to every aspect of business.	[28]
WG12 - Smart Buildings and Architecture	Focus on IoT solutions deployed by various companies along the value chain (i.e., energy companies, supply, traders, *etc.*) to allow the performance optimisation of their energy asset portfolios.	–
WG13 - Smart Buildings and Architecture	Focus on IoT technologies and solutions deployed in buildings and districts of buildings to improve life of the occupant by addressing and optimising elements such as comfort, light, air quality, water, nourishment, fitness, and energy usage.	–

TABLE 2. H2020 projects developed under the ICT-30 programme (2016-2019).

Acronym	Project Name	Integration of devices	Creation of platforms	Interoperable APIs	Autonomous reasoning	Website
AGILE	Adoptive gateways for diverse multiple environments	✓	✗	✓	✗	www.agile-iot.eu
BIG IoT	Bridging the Interoperability Gap of the Internet of Things	✗	✓	✓	✓	www.big-iot.eu
bIoTope	**Building an IoT oPen innovation ecosystem for connected smart objects**	✗	✓	✓	✓	www.biotope-project.eu
INTER-IoT	Interoperability of heterogenous IoT platforms	✓	✓	✓	✗	www.inter-iot-project.eu
symbIoTe	Symbiosis of smart objects across IoT environments	✓	✓	✗	✗	www.symbIoTe-h2020.eu
TagItSmart	Smart Tags driven service platform for enabling ecosystems of connected objects	✓	✓	✗	✗	www.tagitsmart.eu
VICINITY	Open virtual neighbourhood network to connect intelligent buildings & smart objects	✗	✓	✓	✗	www.vicinity2020.eu

access point that should integrate key IoT modules such as modularity, extensibility, privacy and development toolkit management;

- *Creation of platforms:* this topic refers to the definition, specification and extension of platforms, either Cloud-based or local (or both), depending on the pilot needs and requirements. For example, *TagItSmart* and *symbIoTe* are developing Cloud-based services (*TagItSmart* will e.g. re-use available FIWARE components); *bIoTope* and *VICINITY* put particular emphasis on edge nodes (e.g., based on Fog computing and distributed analytics);

- *Interoperable APIs:* this topic refers to standardized and open APIs that must cope with the IoT peculiarities and requirements, e.g. to support efficient data publication, consumption and composition of heterogeneous information sources from across various platforms. Those APIs must provide the necessary messaging interfaces, along with generic content description models for IoT data representation. Each project will investigate, adopt (or develop) such open API solutions, although one challenging task amongst the projects will be the convergence towards the use of a common set of standards;

- *Autonomous reasoning:* this topic refers to context-aware and self-adaptation capabilities of the system/ ecosystem [32].

Two coordination support actions (CSA) are supporting the R&I projects, namely UNIFY-IoT focusing on scientific aspects, and Be-IoT focusing on long-term impact-, community- and ecosystem-building success.
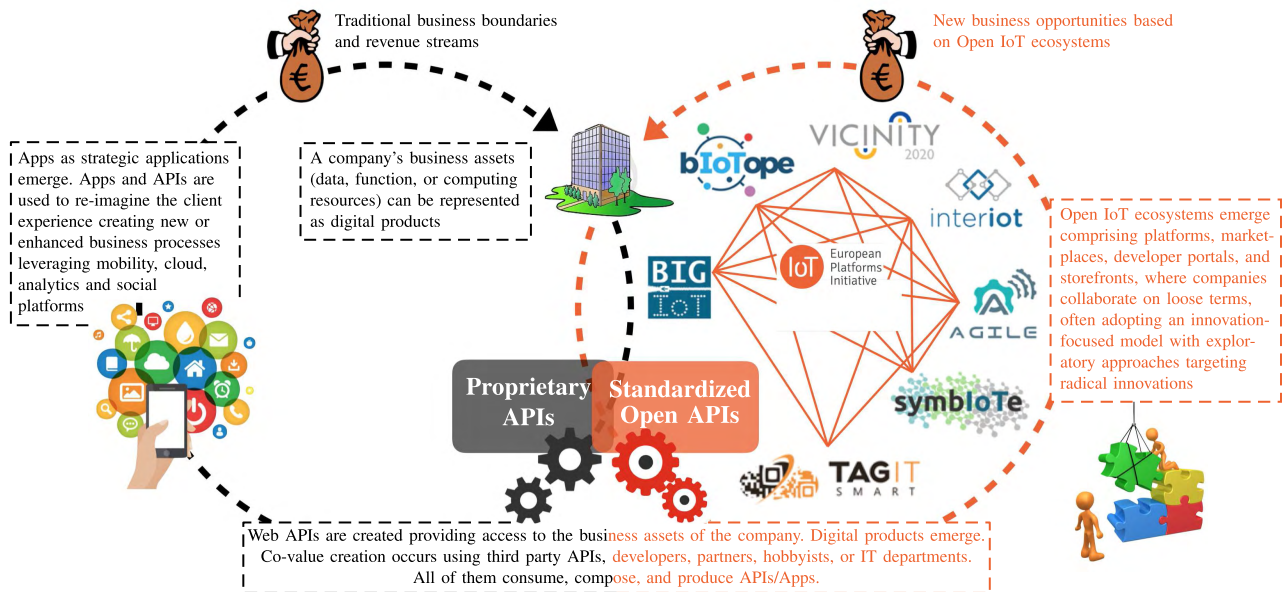
Fig. 1. Towards more open IoT ecosystems for joint service co-creation, delivery, and radical innovation practices.

These two CSA projects are actually leading the IoT-EPI initiative to foster cooperation and convergence between the seven R&I projects. To better understand the ambition of IoT-EPI, Fig. 1 provides an at-a-glance overview of the desired impact from the API economy perspective. As illustrated in this figure, while companies release digital products and services through proprietary APIs, new opportunities arise with the emergence of open ecosystems built upon standardized open APIs, allowing companies to reimagine their business processes and customer experience. Such types of ecosystems are intended to support joint offerings, ad-hoc collaboration, co-creation and co-invention, adopting exploratory approaches targeting radical innovations [6], [33]. As stated in the recent IBM's report, entitled *"The Power of the API Economy"* [34]:

> *"The API Economy has changed how we think about building applications (think apps) and how we deploy software (think cloud). The largest impact of this change for business is speed: Business processes and data are no longer locked inside applications. The result is the **death of data and application silos**."*

In this paper, our research work focuses on the standardized open API developed and used in the bIoTope project, notably the Open Messaging Interface[3] (O-MI) and Open Data Format[4] (O-DF) standards.

III. OVERVIEW OF bIoTope ECOSYSTEM BUILDING BLOCKS

The following ecosystem building blocks are introduced and discussed in sections III-A to III-C, respectively dealing with: (i) the IoT service marketplace enabling the publication and discovery of IoT data and/or services in the ecosystem, (ii) O-MI providing a generic Open API for implementing RESTful IoT information systems, and (iii) O-DF providing a generic content description model for Things in the IoT.

A. IoT SERVICE MARKETPLACE

bIoTope is evolutionary insofar as it lays a solid foundation to allow existing communities of developers (businesses, system integrators, *etc.*) and end-users (citizens, institution and other legal entities) to join an open, easy-to-use and secure IoT ecosystem that fosters new relationships. In keeping with this visionary action, the core concepts and building blocks underlying the bIoTope ecosystem are summarized in Fig. 2 (a few real-life solution providers are referenced in an effort to ease the understanding). First, parts denoted by ① in Fig. 2 stress the fact that the project leverages the available platforms and cloud endpoints such as weather station solution providers (as denoted by "0"), car manufacturers, *etc.*, using them to create smart cities, industries, and homes. The second step, denoted by ②, emphasizes the fact that bIoTope provides the necessary tools to enable any legal entity (citizens, businesses, municipalities, *etc.*) to expose – *using O-MI/O-DF standards as basic interoperability layer* – personal IoT data or[5] IoT services, while providing them with the possibility of (i) choosing what personal data items can be shared with peer systems/developers; (ii) deciding for which purpose personal data can be used, for how long, and at what cost;

---

[3]https://www2.opengroup.org/ogsys/catalog/C14A, last accessed Apr., 2017.

[4]https://www2.opengroup.org/ogsys/catalog/C14B, last accessed Apr., 2017.

[5]A distinction is made between IoT data and IoT services in this document, "IoT data" referring to IoT data streams coming from sensors or other systems generating or holding data (databases, files...), while an "IoT service" refers to the call of a web service that takes, as inputs, one or more parameters and imply a processing stage to return the expected result.
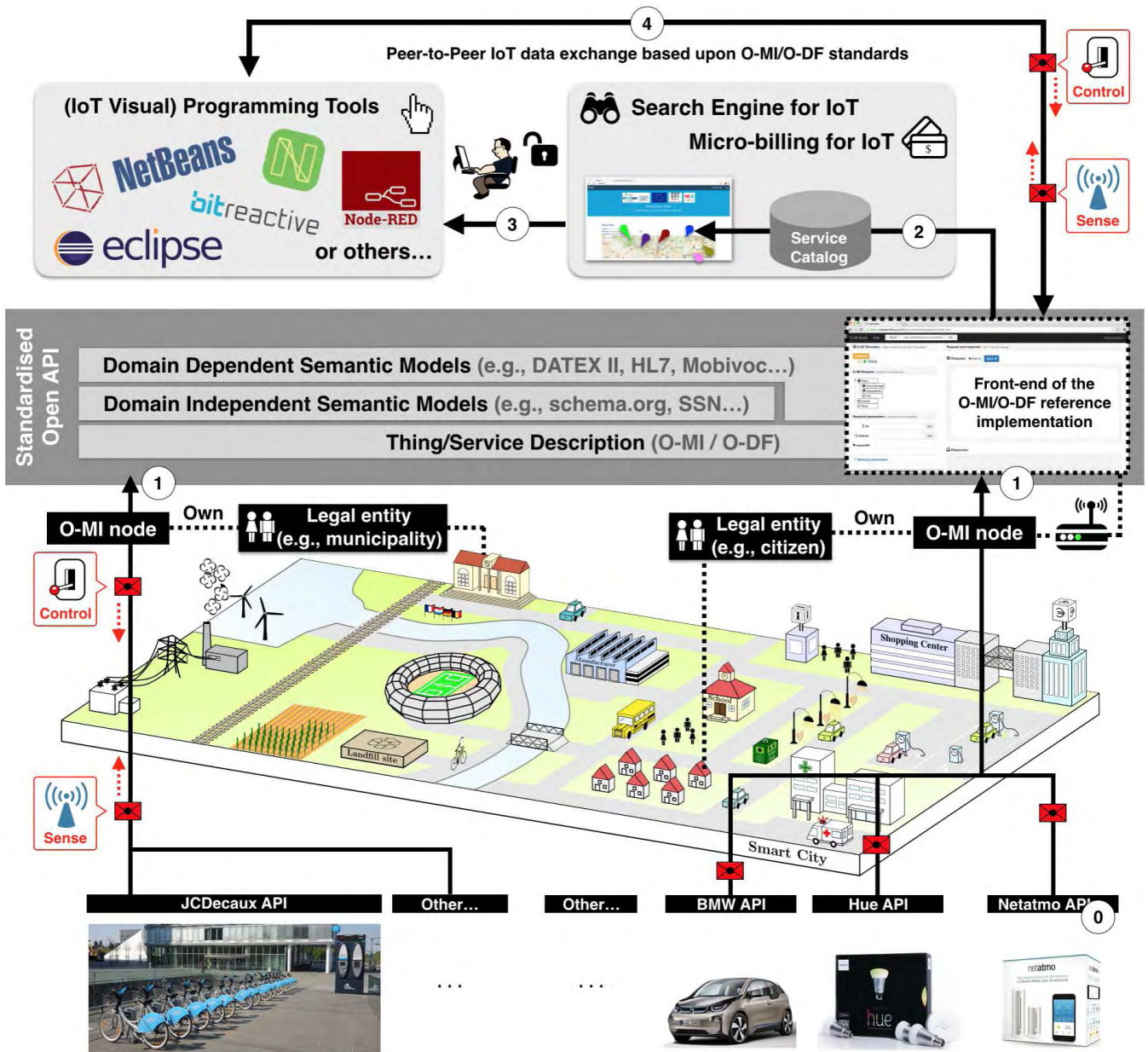
Fig. 2. IoT-based Smart Parking System for Sporting Event Management using O-MI & O-DF standards.

(iii) being informed whenever data is used/called, and by whom.

Although more details about O-MI and O-DF will be provided in sections III-B and III-C, it should be stressed that O-MI provides a generic Open API for implementing RESTful IoT information systems, while O-DF provides a generic content description model for Things in the IoT that can, and should be extended with more specific semantic web vocabularies. Stage denoted by ① in Fig. 2 illustrates this extension principle, where either/both domain-independent vocabularies (such as iot.schema.org, SSN, *etc.*) or/and domain-dependent vocabularies (such as DATEX II or MobiVoc for the mobility sector, HL7 for healthcare, *etc.*) can be used as extension of O-DF. Such vocabularies can

be found through LOV-like repositories[6] [35]. Overall, when used together, O-MI and O-DF provide the necessary tools for "any" IoT information systems to interoperate successfully in ad-hoc manners. Getting back to the main subject of exposing IoT data/services, bIoTope aims to develop an IoT service marketplace[7] (including search engine and smart contract capabilities) whose primary goal is to put IoT data/service

---

[6]LOV (Linked Open Vocabularies) initiative gathers and makes visible indicators such as the interconnections between vocabularies and each vocabulary's version history, along with past and current editor (individual or organization).

[7]A first version of the IoT service marketplace has been released at the following URL: https://otaniemi3d.cs.hut.fi/IoTBnB/, last accessed Apr., 2017.

publishers and potential third party consumers in relation with each other. This is illustrated through stages ② and ③ in Fig. 2, where only the "description" of what data/services are available and how to call/query them is defined. In this respect, the marketplace's IoT search engine should support multimodal search like:

- *Spatial/Temporal-based search:* one may want to search for services within a geographical area;
- *Keyword-based search:* one may want to search for services falling in a specific application area (e.g., mobility, healthcare, *etc.*);
- *Reputation-based search:* one may want to search for a service ensuring a certain level of quality, which comprises various dimensions such as *(i) data quality:* quality of sensor data streams, or of more advanced services (e.g., how accurate a failure prediction algorithm is), *(ii) service owner reputation:* third party developers being able to leave a review about whether the data stream works fine, the publisher of the accessed data stream is reactive when asking question, *etc.*;
- *Contractual term or Technology-based search:* one may want to search only for IoT data/service publishers that make them available for free or are compliant with a specific crypto currency, or data/services that are compliant with specific IPR policies (e.g., license type, *etc.*);

Once a third party developers identify relevant IoT data or services that they would like to access to, agree on the contract terms (potentially leading to monetary transactions), the IoT service marketplace "delivers" one or more API access security tokens and/or certificates (valid for specific durations, access rights, *etc.*) that will enable them to access the data/service from the remote O-MI node/server, whose communications between the consumer and publisher is achieved in an ad-hoc manner (see stage ④). At this stage, third party developers can rely on their own preferred IDE (Integrated Development Environment) for accessing, combining, and processing the accessed data streams or services, meaning that bIoTope does not impose the use of any specific IDE tool, as illustrated in Fig. 2. For example, in section IV, the IoT visual programming tool "Node-Red" (developed by IBM) is used for service composition purposes. The next section provides greater insights into O-MI and O-DF standard specifications.

B. O-MI: A GENERIC AND STANDARDIZED OPEN API

O-MI and O-DF standards emerged out of past EU FP6 and FP7 projects (e.g., PROMISE FP6, LinkedDesign FP7...), where real-life industrial applications required the collection and management of product instance-level information for many domains involving heavy and personal vehicles, household equipment, phone switches, *etc.* [36]–[38]. Information such as sensor readings, alarms, assembly, disassembly, shipping event, and other information related to the entire product life cycle needed to be exchanged between products and systems of different organizations. Based on the needs of those real-life applications, and as no existing standards could

be identified that would fulfil those requirements without extensive modification or extensions, the partner consortia started the specification of new messaging interfaces [36]. Those specifications have since then been further developed and published by the IoT WG of The Open Group. O-MI provides a generic Open API for any RESTful IoT information system, meaning that in the same way that HTTP can be used for transporting payloads in formats other than HTML, O-MI can be used for transporting payloads in nearly any format. In resume, O-MI and O-DF are independent entities that reside in the OSI Application layer, where O-MI is specified at the "communication" level and O-DF at the "format" level.
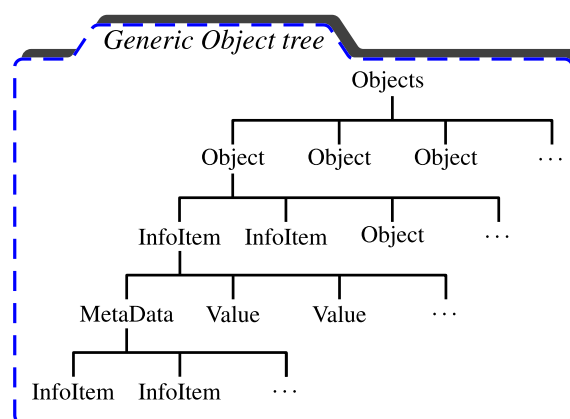

Fig. 3. Open data format: generic "Object" tree.

C. O-DF: A GENERIC CONTENT DESCRIPTION MODEL FOR THINGS

O-DF is defined as a simple ontology, specified using XML Schema – *which might currently be the most common text-based payload format due to its flexibility, thus providing more opportunities for complex data structures [32]* – that is generic enough for representing "any" object and information that is needed for information exchange in the IoT. It is intentionally defined in a similar manner as data structures in object-oriented programming. O-DF is structured as a hierarchy with an "Objects" element as its top element (see Fig. 3), which can contain any number of "Object" sub-elements. "Object" elements can have any number of properties, referred to as InfoItems, as well as "Object" sub-elements. The resulting Object tree can contain any number of levels (*cf.*, Fig. 3). Every Object has a compulsory sub-element called "id" that identifies the Object. The "id" should preferably be globally unique or at least unique for the specific application, domain, or network of the involved organizations. The proof-of-concept developed in section IV will facilitate the understanding of O-DF and the associated Object's tree/hierarchy. As previously mentioned, O-DF can and should be extended – *whenever relevant* – with domain-dependent and domain-independent web vocabularies, consisting in adding some relevant vocabulary tags in the O-DF payload (e.g., as O-DF Object's or InfoItem's id, name, metadata, *etc.*).

TABLE 3. Main messaging interfaces specified in the O-MI standard.

Operations	Description
1 - Write	Used to send information updates to O-MI nodes.
2a - Read (One time)	Used for immediate retrieval of information from an O-MI node.
2b - Read (Subscription)	Two types of subscriptions can be performed: *subscription **with** callback address:* the subscribed data is sent to the callback address at the requested interval. Two types of intervals can be defined: ***interval-based*** or ***event-based***; *subscription **without** callback address:* the data is memorized on the subscribed O-MI node as long as the subscription is valid. The memorized data can be retrieved/polled by issuing a new O-MI read request by using the subscription ID.
3 - Cancel	Used to cancel a subscription before it expires.

Regarding O-MI, one of its defining characteristics is that nodes may act both as "servers" and as "clients", and therefore communicate with each other or with back-end servers in a peer-to-peer manner. Typical examples of exchanged data are sensor readings, lifecycle events, requests for historical data, notifications, *etc.* One of the fundamental properties of O-MI is that O-MI/O-DF messages are "protocol agnostic" so they can be exchanged using HTTP, SOAP, SMTP, or similar protocols. Four operations are supported, as summarized in TABLE 3. Another important feature of O-MI is that messages are "self-contained" in the sense that all the necessary information to enable the recipient to handle the message is contained in the message itself (e.g., actions to be performed, callback address, TTL...). Other relevant interfaces are presented in more details in [36], [39], and [40] such as the "publication and discovery" mechanisms for data, services and meta-data using the "RESTful" URL-based queries. There are several IoT messaging standards comparable with O-MI, and *vice-versa* (e.g., MQTT, AMQP...). Nonetheless, each standard is designed to address different IoT communication requirements. In fact, there are four distinct IoT communication models according to the RFC-7452 for networking of smart objects [41], which are all illustrated in Fig. 4 and described hereinafter:

D2D: Device-to-Device **D2C:** Device-to-Cloud
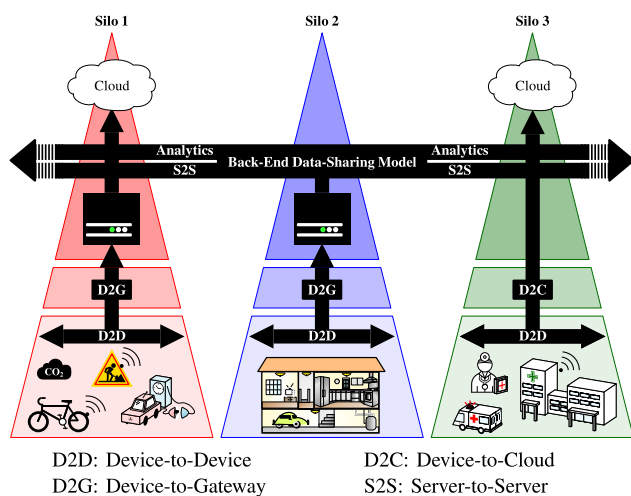**D2G:** Device-to-Gateway **S2S:** Server-to-Server

Fig. 4. IoT communication models (RFC-7452).

- *Device-To-Device (D2D):* two or more devices directly connect and communicate between one another rather than through an intermediary application server (*cf.*, inside silos 1, 2 and 3 in Fig. 4);

- *Device-To-Gateway (D2G):* the IoT device connects to a local gateway device that may either (i) be connected to a Cloud service provider (*cf.*, inside silo 1) or (ii) store and process device-related data at the edge (*cf.*, inside silo 2);

- *Device-To-Cloud (D2C):* the IoT device connects directly to an Internet Cloud provider to exchange data and services (*cf.*, inside silo 3). Frequently, the device and Cloud service are from the same vendor (commonly referred to as "vendor lock-in");

- *Back-End Data-Sharing (S2S):* this model plays a key role in improving horizontal interoperability across vertical silos (*cf.*, Fig. 4). More concretely, this model shall facilitate *Server-To-Server* (S2S) information exchange based on open and standardized IoT interfaces, but shall also provide provisions for Analytics services, e.g. to filter, aggregate and analyze cross-domain and cross-platform information.

TABLE 4. IoT standards *vs.* communication models.

	DDS	MQTT	AMQP	CoAP	XMPP	O-MI
D2D model	✓	✓		✓		
D2G model	✓	✓	✓	✓		
D2C model	✓	✓	✓			
S2S model	✓				✓	✓

TABLE 4 gives insight into well known IoT messaging standards [42], highlighting for which IoT communication model(s) they have been primarily thought and designed for. Our study reports CoAP (developed by IETF), MQTT (developed by IBM), AMQP (developed by OASIS), Data Distribution Service – DDS (developed by the Object Management Group), and XMPP (developed by Cisco). As emphasized in this table, O-MI is primarily aiming at improving horizontal interoperability across vertical silos (S2S). Although this paper is not intended to carry out a technical and thorough comparison between O-MI and the above-mentioned standards, a few striking differences and cornerstones of this standard can nonetheless be pointed out: O-MI uses text-based representations (XML, JSON...) instead of binary formats and can use any of the 'Communication' and 'Transport' level standards as its underlying protocol; O-MI provides a "RESTful" URL-based query mechanism and, like DDS, is "Data-centric" meaning that middleware can understand the data (e.g., object identity, hierarchy...). This table highlights that three messaging protocols have the necessary provisions for Back-End Data-Sharing communications (DDS, XMPP, O-MI), although this may be a debatable topic; this is why
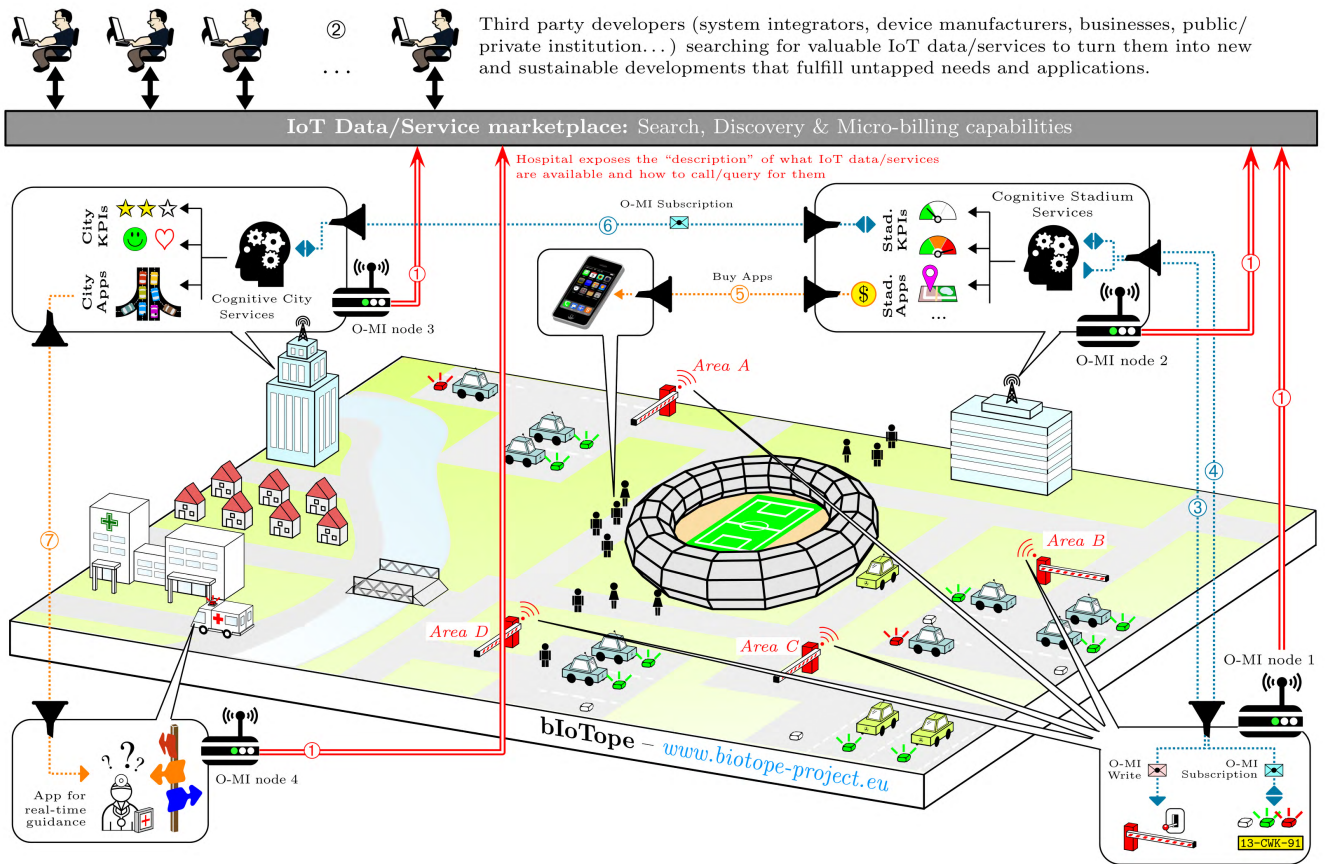
Fig. 5. IoT-based Smart Parking System for Sporting Event Management using O-MI & O-DF standards.

a more in-depth analysis between all these communication protocols should be carried out in the future.

IV. PROOF-OF-CONCEPT: SMART PARKING

The Qatari government that is closely working with Qatar University, expressed an interest in exploring and developing first proofs-of-concept using the O-MI/O-DF standards [43]. In this respect, two usage scenarios have been developed and tested, respectively focusing on smart parking management of (1) cars arriving at stadium and in case of incidents inside the stadium (enabling smart emergency services during sporting events), and (2) city bikes to optimize citizens' and visitors' life during sporting events. Sections IV-A and IV-B focus on scenario 1, including a performance analysis of the O-MI/O-DF standards, while scenario 2 is detailed in section IV-C with a focus on service composition built using both Node-Red and the bIoTope technical building blocks previous described.

A. SCENARIO 1: SMART PARKING MANAGEMENT – INFORMATION PUBLICATION, DISCOVERY AND CONSUMPTION

In our scenario, each spectator has a unique profile that holds personal information, payment tools, and booked stadium seat numbers. Parking spots are booked in-advance through an online booking system that optimizes the spot allocation (e.g., to enable a car owner to be as close as possible to his/her stadium seat). Upon parking spot allocation, users may enter their car plate number to get fast track access to the stadium, which has several outer gates (see Fig. 5). Fast track gates have sensors to read the car plate numbers and check their eligibility to get in. Another sensor located at each parking spot reads the car plate number to check whether the car is or not at the right spot. If not, a signal as a warning (e.g., light or acoustic) will be issued to notify the user about the disturbing situation (*cf.*, red lights in Fig. 5). In this proof-of-concept, we consider a simplified parking that is composed of four parking spot areas, respectively denoted by Areas *A* to *D* in Fig. 5. Those areas are respectively composed of 3, 6, 3 and 3 parking spots denoted by $P_{i,j}$ where i is the area index ($i \in \{A..D\}$) and j the corresponding spot index (e.g., $j \in \{1..6\}$ for $i = B$). Given the parking configuration, several O-MI edge nodes (see O-MI nodes 1 to 4 in Fig. 5) have been implemented.

In our usage scenario, the four O-MI edge nodes' owners can take the decision to expose to the service marketplace one or more of the IoT data/service published by these edge
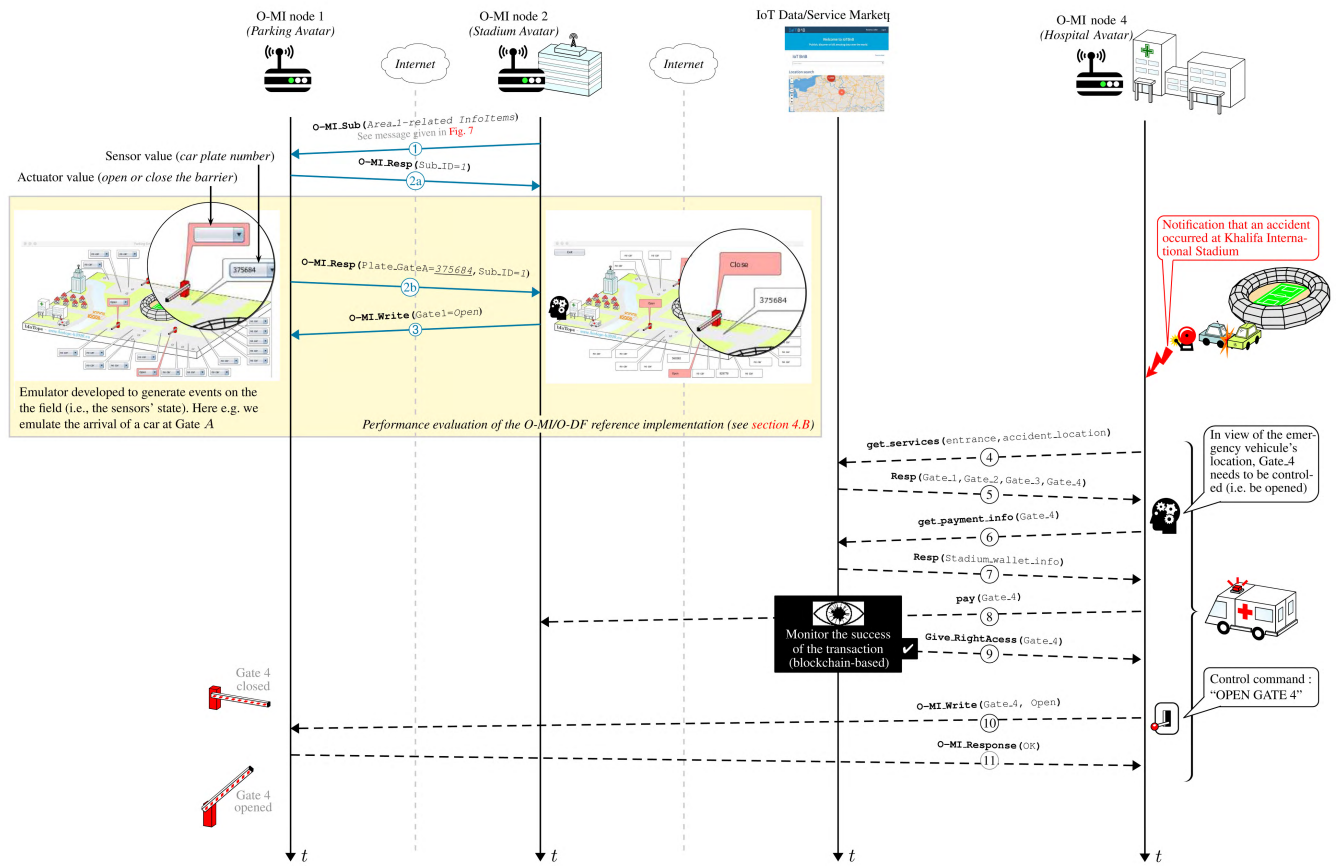
Fig. 6. Smart parking scenario combining a parking emulator and monitoring tool based on the O-MI/O-DF reference implementation.

nodes. This stage is illustrated through arrows denoted by ① in Fig. 5, therefore allowing third party developers to search for, access and benefit from valuable IoT data/services in the city. Such a stage is not detailed in this paper, although an example of how a third party developer can innovate on top of this service marketplace is presented in section IV-C. In this first scenario, we assume that various system integrators, namely the stadium's, city's and hospital's system integrators have developed, based on the information exposed/published by these four O-MI edge nodes, new services for smart sporting event management. Arrows denoted by ③ to ⑦ illustrate the different network communications and interactions between the different O-MI node servers and city stakeholders.

A more detailed view of these network communications between *O-MI node 1* and *O-MI node 2* is given in Fig. 6 (see arrows denoted by "1" and "2a"), whose associated O-MI/O-DF subscription message ("1") is given in Fig. 7. Rows 1 to 5 detail the O-MI message interface where the operation is set to "read" with an interval set to "−1" and a specific callback address (see row 4), meaning – *according to the standard specifications* – that the subscribed data values must be returned in an event-based manner to the stadium office (i.e., O-MI node 2). Rows 6 to 26 detail the message

payload, or to be more exact, the part of the O-DF hierarchy that is subscribed to (the summarized hierarchy view in Fig. 7 helps to better understand how this information hierarchy has been thought/designed for this specific use case). In this example, the stadium office (O-MI node 2) subscribes to *Plate_Number_Readers* information related to Area 1 ($P_{A,1}$ to $P_{A,3}$, Gate1...). Given the message interface setting, the stadium office receives a notification every time that an InfoItem value changes. This is illustrated in Fig. 6 through arrow denoted by "2b", where a car whose plate number is 375684 arrived in front of Gate 1; a notification is then automatically pushed to the stadium office (O-MI node 2) that decides to open (or not) the Gate. In the scenario depicted in Fig. 6, the car is authorized to get in the parking and, accordingly, an O-MI write request is sent to O-MI node 1 (see arrow denoted by "3" in Fig. 6). The information collected by the stadium office can therefore be further processed and turned into (i) new key performance indicators (KPIs) such as the number of free parking spots, car queue length in front of each gate, *etc.*; or still into (ii) stadium free or fee-based Apps (see arrows denoted by ⑤ in Fig. 5) that could potentially inform world cup spectators about how busy a drink and food sale booth is. Fig. 5 highlights through the communication between *O-MI node 3* (city) and *O-MI node 2* (stadium avatar)
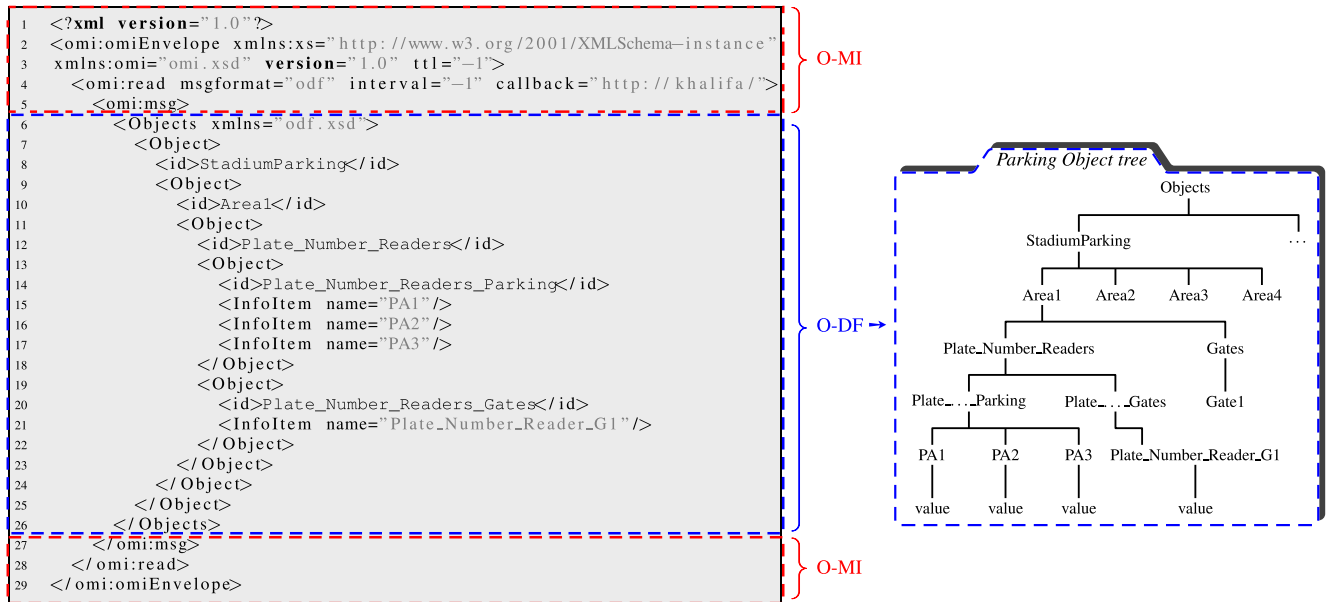
Fig. 7. O-MI/O-DF message and associated information hierarchy when subscribing to Area 1-related data.

how the city can discover, access and use the stadium KPIs for various purposes (e.g., to provide indicators on the city health, citizens' well-being, number of free parking spots and real-time traffic state in the whole city, and so on).

A cross-domain scenario considering an emergency situation in the stadium is proposed and depicted in Fig. 6, where a notification about the emergency is sent to the city hospital. The hospital system sends an emergency vehicle to the stadium site. In order to be aware of whether there are controllable entrances/gates around the area of the accident, the emergency vehicle – *or hospital's back-end system* – calls the service marketplace's API requesting for the set of available IoT data/services that meet this multimodal search demand (see arrow denoted by ④). The marketplace's IoT search engine identifies four InfoItems/Gates in the service catalog, which are returned as a list to the emergency vehicle (see ⑤). In view of the vehicle's location and trajectory in the city, the vehicle predicts that Gate 4 will need to be opened when reaching the stadium. To this end, the vehicle – *or hospital's back-end system* – sends a new request to the service marketplace asking the information needed to pay for accessing and controlling Gate 4. Although this micro-billing process is out-of-scope of this paper (see [44]), the core idea is to establish a smart contract (e.g., based on blockchain-like technologies) between the vehicle/hospital and the "owner" of Gate 4-related data/services (stadium in our scenario) and, for this to happen, the service marketplace provides the vehicle/hospital with the necessary information, while monitoring that the contract/transaction is successfully agreed/made between both parties (publisher-consumer). Once confirmed, the necessary access rights to communicate and control Gate 4 are sent to the hospital/vehicle. All this is illustrated through arrows denoted by ⑥ to ⑨, while the last two stages/arrows emphasize the

O-MI Write request sent by the hospital/vehicle to open Gate 4. Although it is obvious that it makes little sense to charge an emergency vehicle to get in the stadium parking, this scenario was presented first and foremost to describe the automated discovery and micro-billing stages, which are needed in many other IoT applications.

B. SCENARIO 1: A PERFORMANCE ANALYSIS OF O-MI/O-DF

A first version of the O-MI and O-DF reference implementation has been released[8] and used as foundation of our smart parking system's proof-of-concept. As a complement of this reference implementation, a smart parking emulator and monitoring tool has been developed for both emulating the sensor/actuator events occurring on the field (e.g., the arrival of a car at a specific parking area…) and – *from the client side* – for visualizing the current state of the parking. Two screenshots of the parking emulator and monitoring tool are shown in Fig. 6. From an implementation perspective, and according to the O-MI/O-DF reference implementation guidelines, it is necessary to develop a software agent that periodically pushes the emulated data to an internal database (internal to the reference implementation). This data is then published and made available (depending on the access rights) for any peer O-MI node.[9]

Along with this emulator/monitoring tool, we propose to study the performance of the O-MI/O-DF reference implementation in terms of 'efficiency', which refers in this study to the amount of data (called 'data load' in the following) produced by the reference implementation to access one or

---

[8]Github: https://github.com/AaltoAsia/O-MI, last accessed Apr., 2017.

[9]A web-interface is supported facilitating the use of the O-MI operations (read, write…): http://biotope.sntiotlab.lu:8080/html/webclient/index.html

(a) Data Load (*bytes*) and efficiency ratio of the web interface with O-DF payload

(b) Data Load (*bytes*) and efficiency ratio of the REST interface with payload as text-plain value (HTTP GET)

(c) Data Load (*bytes*) and efficiency ratio of the REST interface with O-DF payload (HTTP POST)
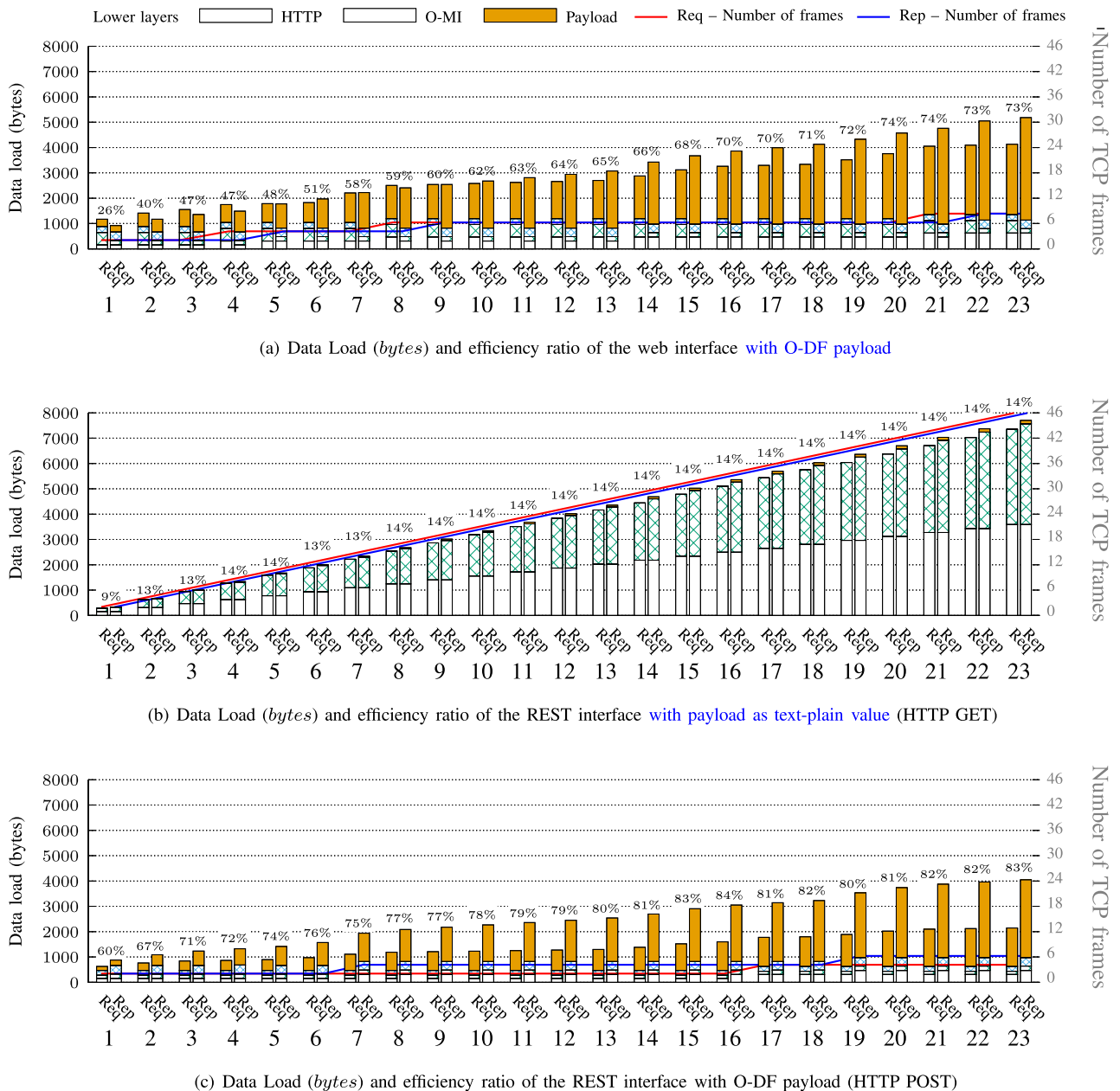
Fig. 8. O-MI/O-DF reference implementation analysis with respect to the standard specifications. (a) Data Load (*bytes*) and efficiency ratio of the web interface with O-DF payload. (b) Data Load (*bytes*) and efficiency ratio of the REST interface with payload as text-plain value (HTTP GET). (c) Data Load (*bytes*) and efficiency ratio of the REST interface with O-DF payload (HTTP POST).

more InfoItems and its efficiency ratio. To this end, a network analyser (Wireshark) has been used to analyze the network traffic considering the smart parking use case, which contains 23 InfoItems in total. At a more concrete level, these 23 InfoItems are read on an incremental basis, i.e. one read request/response including 1 InfoItem (and associated Objects' tree) is performed, then one request/response including 2 InfoItems, and so on. The result of this analysis is given in Fig. 8, where data load is composed of:

- a constant part related to the sum of the Ethernet protocol (26 *bytes*), the IP and TCP headers (respectively 20 and 32 *bytes*) for each request/response and their

respective acknowledgment (78 *bytes* in the reference implementation). The transient states of the TCP opening and closing operations have not been considered. (encapsulation being represented in white color in the histogram). If the O-MI/O-DF message payload needs to be fragmented into a set of frames (according to the Maximum Transmission Unit equals to 1500 *bytes* in our implementation), the sum of the encapsulation is multiplied by the number of frames so as to obtain the global data load;

- a variable part related to the type of request/response. Indeed, when using the reference implementation *via*

a web interface (see Fig. 8(a)), the application protocol HTTP is constant (462 *bytes* for the request and 173 *bytes* for the response) and the message payload is growing depending on the O-DF information hierarchy; the higher the number of levels, the higher the size of the message as demonstrated in Fig. 8(a).

In the performance evaluation process, O-DF is considered as payload in the reference implementation. However, one may argue that the payload is only the 'useful' data for the application, which can be either the value itself or the semantic (O-DF) structure depending on whether the application needs to understand the data before using it (reason for annotating data using O-DF and other web semantic annotations). As a consequence, we decided to study these two considerations that implies using, on the one hand, the reference implementation that conveys the whole or part of the O-DF tree depending on the request (associated performance results are given in Fig. 8(a)) and, on the other hand, the REST interface whose payload consists only of the URL in the request[10] and the value in the response (associated performance results are given in Fig. 8(b)). As can be observed in both figures, using the REST interface needs to send as many frames as InfoItems when reading more than one InfoItem; this is why the data load is continuously growing in Fig. 8(b), and *vice-versa*, using an aggregation strategy (i.e., using the O-DF payload) results in more efficient data load than sending one frame per infoItem. Looking deeper at both histograms, this aggregation strategy is paying off when embedding more than 7-8 InfoItems in a single message (cf. Fig. 8(b)), even though this is true only for this use case since it depends on the O-DF structure and content.

The number of frames and their size can impact the reliability and performance of the application depending, among other things, on the environment in which the application is being run. If the environment is noisy with a high bit/frame error rate (e.g., wireless network or in industrial environments), then it may be more sensible to send smaller frames (i.e., to adopt the REST interface strategy) at the expense of the global data load that increases when reading more than 7-8 InfoItems in this specific case. Indeed, the higher the packet size, the higher the probability that an error occurs, which has a non negligible impact on the efficiency performance due to erroneous frames retransmission. However, the aggregated strategy (i.e., making use of O-DF) adds a generic semantic/vocabulary that is key to automate the reasoning in IoT applications, which is more than essential for enhanced interoperability in the "Back-End Data Sharing" communication model, whereas the REST interface has the advantage of minimizing the load related to the HTTP layer as evidenced when comparing the first request/response between Fig. 8(a) and 8(b). These two advantages could potentially be combined in future reference implementation versions by sending to the REST interface (*via* HTTP POST)

an O-DF payload. Furthermore, since REST-based messages are intended to be processed by machines/devices, we could even suggest to optimize the O-DF payload by removing human-readable constraints imposed by the web-interface of the reference implementation (i.e., spaces and carriage return feeds). Such an hybrid strategy has been set up, whose performance results are given in Fig. 8(c). It can be noted that the size of the frames (and thus the number of frames) and the global data load decrease compared with the web-version (*cf.*, Fig. 8(a)), and the efficiency ratio increases.

In summary, even if the data load generated by the initial version of the O-MI/O-DF reference implementation is non negligible, it remains acceptable for non real-time or critical time applications. Nonetheless, as explained in section III and evidenced through IoT-based smart parking use case, O-MI/O-DF standards have not been designed for such time-constrained applications, but rather to improve interoperability across distinct systems and organizations. Regarding the final smart parking infrastructure, the Qatar government has not yet decided on the technologies to be used/deployed on site, but we believe that our findings can help to decide how to use or properly adapt the O-MI/O-DF reference implementation to the final decisions and expectations. For example, if more automated services (without human in the loop) would need to be developed, we could propose more advanced frameworks that would take full advantage of the REST interface (i.e., both HTTP GET and POST – Fig. 8(b)-8(c)), while taking into account the overall environment and selected technologies (e.g., if the network suffers from high packet loss rates, *etc.*). The self-adaptation capabilities of such frameworks could even take into consideration the final O-DF tree for deciding to switch, when reading a certain number of InfoItems, between the HTTP GET and POST depending on whether or not the aggregation strategy is paying off.

C. SCENARIO 2: SMART PARKING MANAGEMENT – SERVICE COMPOSITION WORKFLOW USING IOT VISUAL PROGRAMMING TOOL

The objective of this second scenario is to show that third party developers – *after having identified, paid for, and received the necessary rights to access specific IoT data/services* – can use their own preferred IDE to develop innovative applications. In this scenario, we consider the open source software tool Node-Red,[11] which allows developers to wire together devices, APIs, and other online services as part of the IoT. In this respect, the bIoTope consortium has developed and released a Node-Red's nodes[12] covering the O-MI and O-DF standard specifications (see bottom-left of Fig. 9).

Overall, Fig. 9 provides an overview of the different flows and nodes used to create a new cross-platform service for

---

[10]Embedded in the HTTP protocol as a plain-text, which means that the size of HTTP varies according to the URL (i.e., the number of digit, e.g. in the string `Objects/StadiumParking/Zone1/Gates/Gate1`).

[11]https://nodered.org, last accessed Apr., 2017.

[12]The corresponding files (*.js* and *.html*) are available at the following GitHub repository, but not yet in the Node-RED Library: https://github.com/skubler, last accessed Apr., 2017.
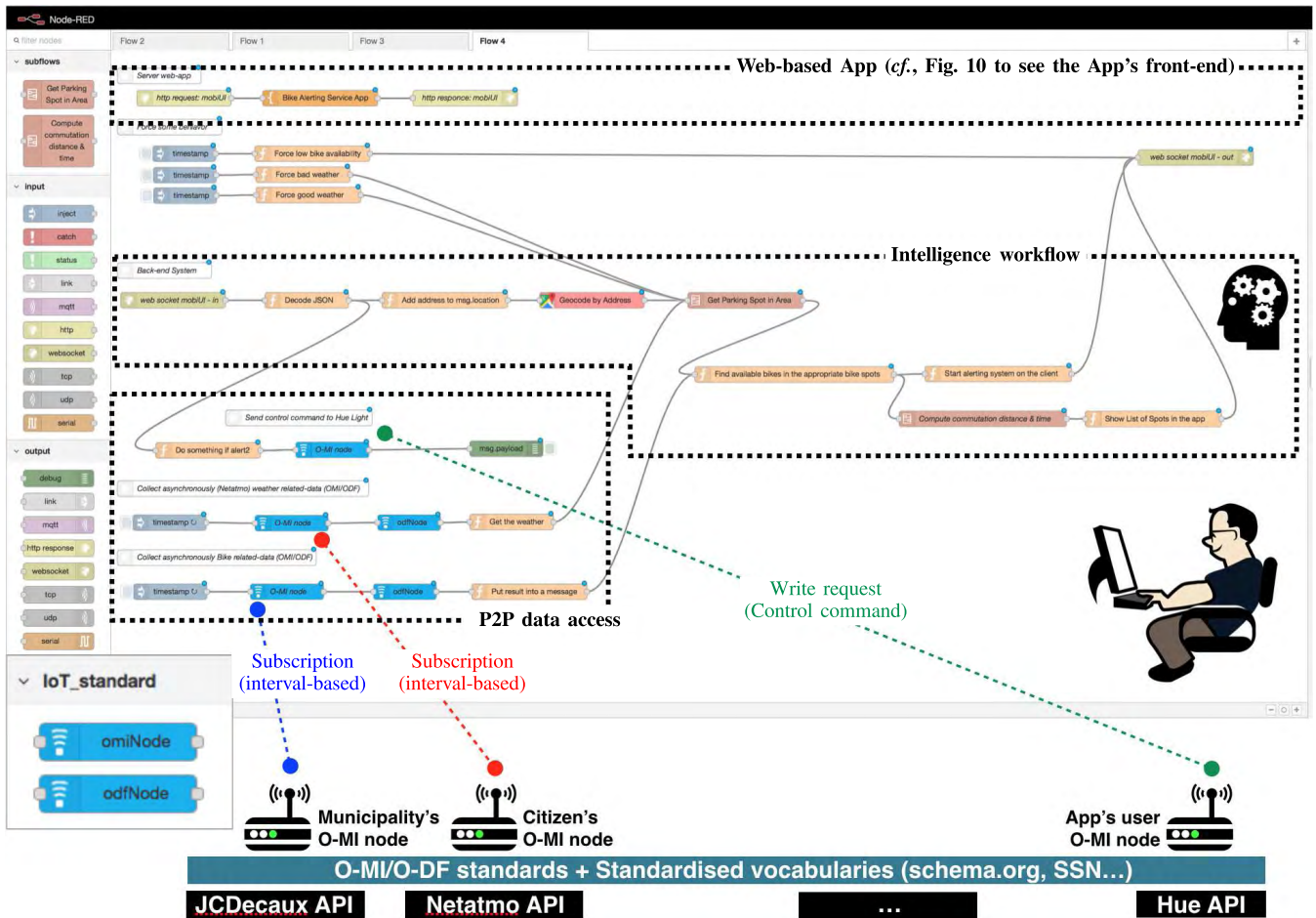
Fig. 9. Innovative Cross-domain & Cross-platform IoT service developed based upon the bIoTope's open IoT ecosystem developed and Node-Red's IoT visual programming tool.

citizen or visitors during the FIFA World Cup 2022 event. This service is intended to enable a citizen/visitor to provide his/her home/hotel location as input parameter in order to be notified about the best bike parking spot to go and pick up a bike, taking into consideration both the weather forecast as well as the number of bikes available per spot. A screenshot of the resulting App that the municipality wants to develop is provided in Fig. 10. For such a development, the municipality's developer identifies via the data/service marketplace that (i) bike parking spot-related data can be accessed for free via the municipalities O-MI node (see Fig. 9); similarly regarding (ii) weather station-related data in the city, where some owners charge for the access; and let us assume that (iii) the App's end-user home or hotel is equipped with a smart bulb that can be controlled using O-MI, which serves in our scenario to turn the light to a specific color when the end-user is about leaving according to the current situation (e.g., turn the light to red if there are less than *n* bikes on all the surrounding parking spots, thus notifying end-users that they have to hurry up or potentially change their plan). Fig. 9 shows the service composition flow that the municipality's developer has developed using Node-Red, which consists of three layers:

- *P2P data access:* this is may be the most important layer, or at least the key message we want to convey through this paper, namely that municipality's developer only has one standard to use for both understanding and accessing IoT data or services in the city, regardless of whether the "underlying" platforms come from hundreds or thousands of vendors. In this scenario, the municipality's developer subscribe to all bike parking spot-related data streams surrounding the end-user's home/hotel (namely the number of available bikes on all spots over the city) and to the weather station that is the nearest of end-user's home/hotel;
- *Intelligence workflow:* this layer contains the set of algorithms that compose the service workflow, fulfilling the service logic/behaviour above-described;
- *Web-based App:* this layer deals with the App's UI;

A recent tutorial and showcase video has been edited and uploaded to Youtube,[13] showing all the elements discussed in this paper, meaning the use of (i) the bIoTope service marketplace, (ii) the O-MI and O-DF nodes in Node-Red,

---

[13]https://www.youtube.com/watch?v=OueY3o-Rf_4&t=36s, last accessed Apr., 2017.

Fig. 10. Citizen App developed based on Node-Red (see Fig. 9).

and (iii) the App developed based on Node-Red. Readers can potentially watch this video to obtain additional information and a better understanding of the bIoTop's vision and ambition.

V. CONCLUSION

The IoT is playing an ever-important role in this new digital landscape, offering us ways to make our world smarter and more interconnected than it has ever been before. Having said that, there are still important challenges ahead that need to be addressed to enable businesses to make the most of the IoT. A crucial challenge is to overcome the "vertical silos" that shape today's IoT (e.g., vendor-lock in/siloed Apps), which are closed to the rest of the IoT and hamper developers to produce new added value across multiple platforms. The EU has taken this challenge very seriously by launching several and complementary IoT Programmes. This paper offers an overview of such EU programmes initiatives. This paper further focuses on the vision, ambition and first technical building blocks developed in the bIoTope H2020 project, which makes use of recent Open API standards named Open Messaging Interface (O-MI) and Open Data Format (O-DF) to fulfill requirements for "Back-End Data-Sharing' communications (RFC7452). Further insights into these standard specifications are provided in this paper, along with

a proof-of-concept – *developed based upon O-MI/O-DF and other ecosystem building blocks* – for enhanced sporting event management in the context of the forthcoming FIFA World Cup 2022 in Qatar.

Although bIoTope like initiatives provide the necessary foundation to create technically and economically viable IoT ecosystems in Europe, there are still challenges to be solved, particularly to leverage semantic web technologies for the IoT (also called "Semantic Web of Things") to converge heterogeneous data sources in a smart ecosystem. The answers that will be given will not put into question the O-DF standard since it only provides a generic content description models for IoT data representation description, which needs to be annotated using other vocabularies (iot.schema.org, SSN, DATEX II, HL7, *etc.*), as can be found on Linked Open Vocabularies (LOV) like repositories that make it possible to lookup vocabularies to annotate and parse messages.

Finally, although this deliverable has not focused on security and privacy aspects, the bIoTope consortium is investigating and developing the necessary building blocks to make the ecosystem robust and resilient against cyber-attacks and/or failures (including identification and authentication, data protection and prevention against threats at both the device and system levels. The ecosystem must provide end-users (either developers who make use of the IoT service marketplace or Apps' end-users) with tools and supports to give them back control over of their personal data/services, e.g. to help them to (i) decide sharing (or stop sharing) personal data with third parties; (ii) know what personal data are exposed and its actual content (ii) audit who are accessing and processing personal data (iii) have automated vulnerability notification mechanisms to ensure that the ecosystem cannot be too much affected by harmful intent (malware, viruses, hackers), *etc.*

REFERENCES

[1] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of Things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, Mar. 2016.

[2] P. Neirotti, A. De Marco, A. C. Cagliano, G. Mangano, and F. Scorrano, "Current trends in smart city initiatives: Some stylised facts," *Cities*, vol. 38, pp. 25–36, Jun. 2014.

[3] K. M. Alam and A. El Saddik, "C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems," *IEEE Access*, vol. 5, pp. 2050–2062, 2017.

[4] M. Naphade, G. Banavar, C. Harrison, J. Paraszczak, and R. Morris, "Smarter cities and their innovation challenges," *Computer*, vol. 44, no. 6, pp. 32–39, Jun. 2011.

[5] K. Pahlavan, P. Krishnamurthy, and Y. Geng, "Localization challenges for the emergence of the smart world," *IEEE Access*, vol. 3, pp. 3058–3067, 2015.

[6] H. Schaffers *et al.*, "Smart cities as innovation ecosystems sustained by the future Internet," Tech. Rep., 2012, p. 65. [Online]. Available: https://hal.inria.fr/hal-00769635/document

[7] S. Zhao, L. Yu, and B. Cheng, "An event-driven service provisioning mechanism for IoT (Internet of Things) system interaction," *IEEE Access*, vol. 4, pp. 5038–5051, 2016.

[8] S. Kubler, K. Främling, and A. Zaslavsky, "bIoTope: Building an IoT open innovation Ecosystem for connected smart objects," in *Digitising the Industry—Internet of Things Connecting the Physical, Digital and Virtual Worlds*. River Publishers, 2016, pp. 270–274. [Online]. Available: http://www.internet-of-thingsresearch.eu/pdf/Digitising_the_Industry_IoT_IERC_2016_Cluster_eBook_978-87-93379-82-4_P_Web.pdf

[9] K. A. Paskaleva, "The smart city: A nexus for open innovation?" *Intell. Build. Int.*, vol. 3, no. 3, pp. 153–171, 2011.

[10] M. Abu-Matar, "Towards a software defined reference architecture for smart city ecosystems," in *Proc. IEEE Int. Smart Cities Conf.*, Sep. 2016, pp. 1–6.

[11] D. Raggett, "The Web of Things: Challenges and opportunities," *Computer*, vol. 48, no. 5, pp. 26–32, 2015.

[12] *Alliance for Internet of Things Innovation (AIOTI)*, Eur. Commission, Brussels, Belgium, 2015.

[13] The Open Group. (2014). *The Nexus of Forces in Action—Business Use-Cases of Open Platform 3.0*, accessed on Feb. 1, 2017. [Online]. Available: https://www2.opengroup.org/ogsys/catalog/W145

[14] J. Swetina, G. Lu, P. Jacobs, F. Ennesser, and J. Song, "Toward a standardized common M2M service layer platform: Introduction to oneM2M," *IEEE Wireless Commun.*, vol. 21, no. 3, pp. 20–26, Jun. 2014.

[15] R. Minerva, "Towards a definition of the Internet of Things," White Paper, 2015.

[16] S. Rhee, "Catalyzing the Internet of Things and smart cities: Global city teams challenge," in *Proc. 1st Int. Workshop Sci. Smart City Oper. Platforms Eng. (SCOPE)*, 2016, pp. 1–4.

[17] AIOTI WG01, "Report on Internet of Things applications," Alliance Internet Things Innov., Tech. Rep., 2015.

[18] AIOTI WG02, "Report on innovation ecosystems," Alliance Internet Things Innov., Tech. Rep., 2015.

[19] AIOTI WG03, "Report on high level architecture (HLA), release 2.0," Alliance Internet Things Innov., Tech. Rep., 2015.

[20] AIOTI WG03, "Report on semantic interoperability, release 2.0," Alliance Internet Things Innov., Tech. Rep., 2015.

[21] AIOTI WG03, "Report on IoT LSP standard framework concepts, release 2.0," Alliance Internet Things Innov., Tech. Rep., 2015.

[22] AIOTI WG04, "Report on Policy," Alliance Internet Things Innov., Tech. Rep., 2015.

[23] AIOTI WG05, "Report on smart living environment for ageing well," Alliance Internet Things Innov., Tech. Rep., 2015.

[24] AIOTI WG06, "Report on smart farming and food safety Internet of Things applications—Challenges for large scale implementations," Alliance Internet Things Innov., Tech. Rep., 2015.

[25] AIOTI WG07, "Report on Wearables," Alliance Internet Things Innov., Tech. Rep., 2015.

[26] AIOTI WG08, "Smart city LSP recommendations report," Alliance Internet Things Innov., Tech. Rep., 2015.

[27] AIOTI WG09, "Report on smart mobility," Alliance Internet Things Innov., Tech. Rep., 2015.

[28] AIOTI WG11, "Report on smart manufacturing," Alliance Internet Things Innov., Tech. Rep., 2015.

[29] S. Kubler, J. Robert, A. Hefnawy, C. Cherifi, A. Bouras, and K. Främling, "IoT-based smart parking system for sporting event management," in *Proc. 13th Int. Conf. Mobile Ubiquitous Syst., Comput., Netw. Services*, 2016, pp. 104–114.

[30] J. C. Miguel and M. A. Casado, "GAFAnomy (Google, Amazon, Facebook and Apple): The big four and the b-ecosystem," in *Dynamics of Big Internet Industry Groups and Future Trends*. Springer, 2016, pp. 127–148.

[31] O. Vermesan and P. Friess, Eds., *Internet of Things Applications—From Research and Innovation to Market Deployment*. River Publishers, 2014, pp. 243–286. [Online]. Available: http://www.internet-of-thingsresearch.eu/pdf/IoTFrom%20Research%20and%20Innovation%20to%20Market%20Deployment_IERC_Cluster_eBook_978-87-93102-95-8_P.pdf

[32] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 414–454, 1st Quart., 2014.

[33] L. Atzori, A. Iera, and G. Morabito, "From 'smart objects' to 'social objects': The next evolutionary step of the Internet of Things," *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 97–105, Jan. 2014.

[34] K. Holley *et al.*, *The Power API Economy—Stimulate Innovation, Increase Productive, Develop New Channels, Reach New Markets*. USA: IBM Corporate, 2014. [Online]. Available: http://www.redbooks.ibm.com/redpapers/pdfs/redp5096.pdf

[35] P.-Y. Vandenbussche, G. A. Atemezing, M. Poveda-Villalón, and B. Vatant, "Linked open vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web," *Semantic Web*, vol. 8, no. 3, pp. 437–452, 2017. [Online]. Available: http://www.semantic-web-journal.net/system/files/swj1127.pdf

[36] K. Främling, S. Kubler, and A. Buda, "Universal messaging standards for the IoT from a lifecycle management perspective," *IEEE Internet Things J.*, vol. 1, no. 4, pp. 319–327, Aug. 2014.

[37] K. Främling, J. Holmström, J. Loukkola, J. Nyman, and A. Kaustell, "Sustainable PLM through intelligent products," *Eng. Appl. Artif. Intell.*, vol. 26, no. 2, pp. 789–799, 2013.

[38] D. Kiritsis, "Closed-loop PLM for intelligent products in the era of the Internet of Things," *Comput.-Aided Design*, vol. 43, no. 5, pp. 479–501, 2011.

[39] S. Kubler, K. Främling, and A. Buda, "A standardized approach to deal with firewall and mobility policies in the IoT," *Pervas. Mobile Comput.*, vol. 20, pp. 100–114, Jul. 2015.

[40] S. Kubler, K. Främling, and W. Derigent, "P2P data synchronization for product lifecycle management," *Comput. Ind.*, vol. 66, pp. 82–98, Jan. 2015.

[41] H. Tschofenig, J. Arkko, and D. McPherson, "Architectural considerations in smart object networking," Internet Eng. Task Force, Fremont, CA, USA, Tech. Rep. RFC-7452, Mar. 2015.

[42] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015, doi: 10.1109/COMST.2015.2444095.

[43] A. Hefnawy, A. Bouras, and C. Cherifi, "Integration of smart city and lifecycle concepts for enhanced large-scale event management," in *Proc. Int. Conf. Product Lifecycle Manage. (IFIP)*, Doha, Qatar, 2015, pp. 687–697.

[44] J. Robert, S. Kubler, and Y. Le Traon, "Micro-billing framework for IoT: Research & technological foundations," in *Proc. IEEE 4th Int. Conf. Future Internet Things Cloud*, Aug. 2016, pp. 301–308.

SYLVAIN KUBLER received the M.Sc. and Ph.D. degrees in computer science and engineering from the Université de Lorraine, France, in 2009 and 2012, respectively. He was a Post-Doctoral Researcher with Department of Computer Science, Aalto University, from 2013 to 2015. He is currently a Research Associate with the Interdisciplinary Center for Security, Reliability and Trust, University of Luxembourg. He has broad expertise in Internet of Things, cyber physical systems, intelligent products, context-awareness, product lifecycle information management, multi-criteria decision making, fuzzy logic, and communication networks. He received the Best Thesis Award in automatic control from the IFAC French Work Group GdR MACS/Club EEA.

JÉRÉMY ROBERT received the M.Sc. and Ph.D. degrees in computer science and engineering from the University of Lorraine, France, in 2009 and 2012, respectively. He is currently a Research Associate with the Interdisciplinary Center for Security, Reliability and Trust, University of Luxembourg. He has broad expertise in industrial and embedded networks, such as the use of switched Ethernet embedded in the future space launchers. His research interests include heterogeneous data communication challenges in the Internet of Things and the implementation of messaging services and high-level data formats. As the major research work was conducted in collaboration with the industry, these skills could be therefore applied in the area of the smart factory (industry 4.0) and smart cities.

AHMED HEFNAWY received the master's degree in communications management from the University of Strathclyde, U.K. He is currently a Ph.D. degree student with the Université Lyon 2, France, with a focus on using lifecycle management in the smart city ecosystem, in order to better integrate people, processes, and systems. He has accomplished diversity of technical, regulatory, and policy experience in the ICT industry spans over 15 years. He is currently the ICT Policy Manager with the Ministry of Transport and Communications, Qatar, where he is involved in developing essential ICT policies, strategies and frameworks, and conducts necessary research to identify ICT policy gaps according to international best practices and comparative analysis.

KARY FRÄMLING received the M.Sc. degree in computer science from the Helsinki University of Technology in 1990 and the Ph.D. degree from the Ecole Nationale Suprieure des Mines de Saint-Etienne, France, in 1996. He is currently a Professor of Practice in building information modeling (BIM), Aalto University, Finland. His main research topics are on information management practices and applications for BIM, product lifecycle management in general, distributed systems, middleware, multiagent systems, autonomously learning agents, neural networks and decision support systems. He is currently the Scientific Coordinator of the bIoTope H2020 project (2016-2019), 9.6M€.

CHANTAL CHERIFI received the Ph.D. degree in computer science from Corsica University, France, in 2011. She is currently an Associate Professor with the DISP Laboratory, Université Lyon 2, France. Her current research interests focus on information systems agility with SOA architectures, smart cities, and network science.

ABDELAZIZ BOURAS (M'15) received the Ph.D. and Habilitation degrees in computer science from the Université Lyon 2, France, and the Honoris-Causa Ph.D. degree in science from the HRH Princess Maha Chakri Sirindorn of Thailand. He was a Full Professor (Exceptional Class) with the University of Lyon 2. In 2013, he joined Qatar University as the Chair of the ictQATAR scientific chair position, where he is currently a Professor with the Computer Science and Engineering Department, College of Engineering. He is also the Chair of the IFIP WG5.1 on ICT Lifecycle management and contributes to several international programs. He leaded or participated in over 12 international projects and contributed to the creation of start-ups including Ginkyo, France, and the Digital Incubation Center, Qatar. He has founded or co-founded several international conferences, such as the IFIP PLM International Conference, the IEEE SKIMA International Conference, and the CAD International Conference and scientific journals including the SSMS, the IJPLM, the JMPM, and the IJPD, and consulted for governments, organizations, and companies in Europe, America, and Asia. He is a member of the ASME, the IFIP, the Design Society, the GDR-MACS, the V-LAB, and the MICADO. He received several international awards and grants for his initiatives in the field of ICT and their applications.

● ● ●