Survey of Internet-of-Things platforms

Milan Zdravković[*], Miroslav Trajanović[*], João Sarraipa[**], Ricardo Jardim-Gonçalves[**], Mario Lezoche[***], Alexis Aubry[***], Hervé Panetto[***]

[*] Laboratory for Intelligent Production Systems (LIPS),
Faculty of Mechanical Engineering, University of Niš, Niš, Serbia

[**] CTS, Uninova, Dep. de Eng. Electrotécnica, Faculdade de Ciências e Tecnologia, FCT ,
Universidade Nova de Lisboa , Caparica , Portugal

[***] Research Center for Automatic Control of Nancy (CRAN), Université de Lorraine, UMR 7039, Boulevard des Aiguillettes B.P.70239, 54506 Vandoeuvre-lès-Nancy, France.

milan.zdravkovic@gmail.com, miroslav.trajanovic@masfak.ni.ac.rs
{jfss, rg}@uninova.pt
{mario.lezoche, alexis.aubry, herve.panetto}@univ-lorraine.fr

Abstract **– Internet-of-Things (IoT) platform (often referred to as IoT middleware) is a software that enables connecting the machines and devices and then acquisition, processing, transformation, organization and storing machine and sensor data. The objective of the research behind this paper is to establish a state of the art in the development of IoT platforms. In specific, here we present the main conclusions regarding the functional and design perspective to current IoT platforms and related research. The focus was made on cloud-based IoT platforms with significant user base and successful record of exploitation. We also consider the relevant theoretical foundations of IoT platform research, mostly by taking into account the results of European research. The conclusions and respective discussion are meant to be used in the development of a formal runtime model-driven IoT platform, which conceptual design decisions are shortly presented.**

I. Introduction

Internet of Things (IoT) has attracted attention of major players in industrial landscape and it is currently one of the most expected emerging technologies. According to the 2015's Gartner's Hype Cycle for Internet of Things [1], IoT is currently at the "peak of inflated expectations". Although some of its technologies and enablers are getting closer to the so-called Plateau of Productivity (Wireless Healthcare Asset Management, ZigBee, RFID for logistics and transportation), IoT platforms are still at inception phase of development. The potential market impacts are considered as quite significant.

The development of IoT platforms is driven by the need to facilitate machine-to-machine (M2M) connectivity, which is emerging at unprecedented rate. Machina Research [2] predicts that M2M connections will rise from two billion in 2012 to 12 billion in 2020. Cisco [3] values IoT market to 19 trillion USD. According to the same source, only 0.6% of the physical objects the potential candidates for IoT are currently connected. Different sources refer to estimated 50 billion objects online on 2020.

The previously mentioned Gartner's study [1] suggests that the need for IoT platforms is most precisely articulated in the sector of consumer-centered enterprises, which are currently classified into early adopters of IoT. According to Gartner, in 5 years, 2-3 of 10 homes will be connected homes, with about 500 connectable devices. In service sector, the increase in overall effectiveness of employees and workplaces has been sought as the key expectation from IoT. Intelligent business operations, machine learning and RFID for logistics and transportations are foreseen as the earliest facilitators (within 5 years). Industrial sectors expect a significant impact of IoT on factory floors; this impact will be driven in a short term by enterprise manufacturing intelligence and facilities energy management (within 5 years).

Obviously, all these devices and technologies need a platform which will act as a command centre in homes, workplaces and factory floors. Google Nest and Apple Homekit are some of the examples of "home" aggregators, IoT platforms capable to implement home automation functionality. IoT platform (often referred to as IoT middleware) is thus defined as a software that enables connecting the machines and devices and then acquisition, processing, transformation, organization and storing machine and sensor data.

There exists a strong need for quite diverse sets of methodologies and tools for effective deployment of IoT solutions on different scales. In fact, large-scale deployment of IoT projects under realistic conditions has been considered only recently [4]. The set of methodologies and tools deployed in the specific domain, as the specific IoT solution, is typically referred to as an integrated IoT platform. IoT platform will be built within the complex ecosystem of machines, software and people, dealing with different relevant issues, spanning from M2M connectivity, to data analytics and visualization, IoT application development and others. The most important common features of the different domains are massive scaling and security.

Main objective of this paper is to define the key directions in the development of a formal model-based, generic IoT platform. These directions are presented in the conclusions of the paper (Section 5). They are made based on the discussion (Section 4) of the theoretical foundations for IoT platforms (presented in Section 2); and existing IoT platforms (Section 3).

II. Theoretical Foundations for IoT Platforms

Stankovic [5] highlighted the following research directions for IoT:

- massive scaling (addressing, discovery, architectural models that can support the expected heterogeneity),
- architecture and dependencies (IoT apps, deployment, resolving interference problem in using the utility device from different apps by some kind of multiplexing, dependencies across applications especially for safety critical apps or when actuators can cause harm),
- creating knowledge and big data (real-time data interpretation, knowledge formation, new inference techniques, trusting data by using confidence levels, reliable data associations),
- robustness and openness,
- security (detection and diagnosis of attack and deployment of countermeasures),
- privacy (evaluate requests against policies, reconciliation of the different policies) and
- humans in the loop (modeling human behaviors, human use and control).

Obviously, each of the listed directions is highly relevant for the IoT platform. In fact, the design of one IoT platform must take into account all these directions and its development approach must be defined so IoT platform becomes an enabler of each of the above factors.

As the technologies and standards used for IoT device manufacturing and communication are still in very early phase of development and adoption, IoT platforms should embrace a role of IoT experimentation facilities. Gluhak et al [4] identified the requirements for a next generation of experimental research facilities for the IoT:

- scale (supporting thousands of nodes: minimized human intervention, maximized plug-and-play configuration, automatic fault management),
- heterogeneity (management of devices, easy programmability of heterogeneous devices),
- repeatability (across different test beds: agreements on standards),
- federation (with other test beds, or other experiments: common framework for authentication, interoperability),
- concurrency (virtualization of devices, multiple experiments for one device),
- experimental environment (robustness towards the environmental conditions),
- mobility (handling system dynamics, movement of devices) and
- user involvement and impact (multi-modal mechanisms for user feedback, automated detection of situations where user behavior influences the data validity).

III. Cloud-based IoT platforms and services

In this section 16 different cloud-based IoT platforms and services are presented, with short overview of their distinguishing architectural designs and functionality. Initial selection of platforms was done by a Google search. Then, based on the analysis of the website content, the selection is filtered to the platforms with a significant customer base and partnerships with device manufacturers and system integrators.

Some platforms without M2M connectivity features are included, considered as IoT support platforms. Domain-specific platforms are not presented in this paper. The platforms are listed in alphabetical order.

Arrayent. The IoT platform is composed of four components. Connect Agent is a firmware, a lightweight agent deployed in devices (bulk firmware updates enabled). It exchanges data with the Connect Cloud by using 128-bit AES encryption. Each of these devices has its own digital copy in Connect Cloud which hosts the virtual devices to which mobile apps can connect. Mobile Framework is used for development of apps which manage connected devices. It uses also engine for managing and sending triggered alerts that could also then trigger response actions in the product which generated the alert. Finally, Insights provides secure access to data via dashboards, batch exports, data streaming and data connectors.

Axeda. Connectivity middleware facilitate connecting machines and devices to the cloud. Application enablement platform simplifies development of IoT apps, with capabilities such as data management, scripting engine, integration framework, SDKs and web services for accessing data and apps in the cloud. Connected machine management applications facilitate remote monitoring, management, service and control of remote devices. Capabilities also include software (client, firmware) distribution and configuration management.

Bugswarm is a lightweight platform that can acquire data from and control devices using JavaScript or plain HTTP. It defines a “swarm” – system of resources which can communicate to other resources within the system, according to the defined access policy. A resource is considered as anything that can communicate through HTTP, not only devices but also web or mobile applications. Device-specific, client-side applications, device connectors are available for use, to connect existing device as a resource to a swarm. When the specific device is connected, it sends the private message to all swarm members, with the list of its capabilities or services that the device can provide (feeds). Other resources interested in these services could send a feed request to a device, which then responds with a feed response (typically, with sensed data).

Carriots. The platform is an aggregator; it enables connecting any type of device with web connectivity which can send a stream of data, by using MQTT, cURL, hURL or Poster, to Carriots REST API. For each of the protocols, a client installation is needed on the device. Then, Listener or Trigger component can be developed and deployed on platform to perform operations with data. Device control and maintenance is enabled (checking status, managing configurations, updating firmware). All development is being done by using Java Carriots SDK, by putting a code to the specific fields in Carriots Control Panel web application (Java code interpretation/execution). Free use is enabled with limited functionality (up to 10 devices).

Evrythng is natively a digital identity management platform, often referred to as “Product Relationship Management” (PRM) platform. Semantic data store is used to customize dynamic data profiles – digital identities of the products, so they can exchange data with authorized applications.

Exosite is cloud-based IoT platform offering M2M connectivity and data visualization tools and services. Open API is available for advanced data processing and integration with enterprise applications.

GrooveStreams is data analytics cloud platform, allowing data collection from multiple platforms, including IoT devices. Open API can be used to send data streams at a fixed (up to 10 second) or random interval or as a point stream of a fixed value. Data analytics tools are offered with near real-time performance. Data can be redistributed as Derived Steams, or visualized with customizable charts and graphs. The platform is open access. Premium features are available, related to number of organizations, users and increased (scalable) data I/O rate.

Ifttt (If This Then That) is not a native IoT platform; it is an interoperability-as-a-service platform which allows users to create chains of simple conditional statements, called "recipes", which are executed upon the particular events recorded from the different services. It is the platform which enables users to create their own recipes, which can also include events from the different devices. Some of the existing examples of IoT related recipes are: "delay watering your garden if it's going to rain tomorrow", "receive and emergency call if smoke is detected", and others. There also exist alternatives to this service, such as Zapier and Yubnub.

Kaaproject is open-source IoT middleware platform. It enables management and maintenance (firmware updates distribution) of device inventory and near real-time communication between the devices. It is transport-agnostic and promotes use of structured data. It provides endpoint SDKs that can be embedded into devices. Complete solutions already exist for Android, IoS, Raspberry Pi and other platforms. It is pre-integrated with existing data processing solutions, such as mongoDB, Hadoop, Oracle and others.

LinkSmart open source middleware platform is a framework and a service infrastructure for creation of IoT applications, originally developed by Hydra EU project. The project is hosted by Fraunhofer FIT. It includes Device Connector for integrating devices (with different implementations for specific devices), Resource Catalog for managing devices and resources they expose, Service Catalog (services used to access devices and resources) and GlobalConnect tunneling service that enables access to devices beyond the boundaries of a private (routable) network.

Mbed platform aims at even tighter integration, by treating all its connected devices as embedded devices. They all have in-house mbed open-source Operating System, event-driven single-threaded architecture (scales down to the simplest, lowest cost, lowest power consumption devices). Mbed supports only devices based on ARM Cortex-M microcontroller. Key principles are security, connectivity and manageability (uses OMA Lightweight M2M, a popular protocol for monitoring and managing embedded devices). The architecture consists of mbed OS, Device connector (works with REST APIs), TLS (includes cryptographic and SSL/TLS capabilities in embedded products), client (library that connects devices to device connector service and mbed server – free, high-level C++ API) and server (essentially, a middleware, also hosted as a cloud service, connects IoT devices to web applications).

Nimbits is data logging service and rule engine platform for M2M connectivity. It provides nimbits.io open source Java library for developing Java, web and Android solutions that use Nimbits Server as a backend platform. Backend platform collects geo and time-stamped data and executes rules on this data, such as calculations, email alerts, xmpp messages, push notifications and others. Free and Enterprise editions of the server are available.

Particle.io (former Spark.io) offers hardware development kits for building the firmware for the devices, by using web-based IDE and deploying this firmware over the air. Then, ParticleJS and Mobile SDK libraries can be used to build web and mobile apps, based on the collected data.

Autodesk SeeControl is IoT cloud service which virtualizes machines, links them with reporting devices and use analytics to unlock their data. No-coding, drag-and-drop approach is implemented. Platform is focused to the needs of the manufacturing industries, in specific to generating product performance data, predicting a product failure, performing maintenance and optimizing supply chain and material replenishment costs. It provides a large library of existing protocol/vendor device adapters. Service also includes light ERP modules and business management tools.

SensorCloud is a cloud IoT platform for acquisition, visualization and analysis of data. The platform natively supports connectivity with LORD MicroStrain's wireless and wired sensors. Visualization tools are available. It is possible to setup simple alerts, triggered by the data threshold values. MathEngine analysis tools are provided, with a simple interface which facilitates common operations such as FFTs, smoothing, filtering and interpolation.

PTC ThingWorx. In the platform, each device is represented by so-called Thing Template. Template defines properties (for example, temperature), services (for example, posting to Facebook) and events (for example, malfunction). Devices use agents to connect to IoT platform; different agents are used for the different types of devices. Composer application is used to model the things, business logic, visualization, data storage, collaboration and security required for IoT application. Mashup can be assembled by using different thing templates, namely, UI widgets which are pre-wired to the thing templates. The mashups are then used as interactive IoT applications, real-time dashboards, collaborative workspaces and mobile interfaces. BPM component is included to enable definition and execution of the processes, starting with an alert or event from a remote connected device. Device asset management tool is also included to facilitate remote diagnostics, control and scheduled software update of things. Free use is possible, with limited functionality.

ThingSpeak is another IoT platform, with features very similar to SensorCloud. It features open channels with available data from different devices, published by the users. Platform enables actuation, namely talking back to the device, which is done over HTTP.

IV. Discussion

Today's IoT platforms are typically cloud based, delivering PaaS (Platform-as-a-Service). Communication with devices is made by installing or developing light clients, which only purpose is to facilitate connectivity to the central platform. Pervasive (over any communication channel, including cellular, WiFi and satellite) and reliable (where connection quality is considered as a primary criteria) connectivity is a key feature.

Typical features of IoT platforms are: connectivity as a service, monitoring and maintenance of devices (including firmware updates), data visualization, data analytics, basic application logics through alerts and triggers.

Connectivity as a service is achieved by enabling the unconditional (with installed client) access to devices, no matter if they are located behind the firewall, a NAT or mobile network router. The service should work with any device that provides a TCP socket.

The following categories of existing IoT platforms were identified:

- Domain-specific platforms are the IoT platforms which facilitate specific domain scenarios. Often, these platforms are built on the top of the generic M2M connectivity providers. Examples of such platforms are rachio for smart irrigation, nest for home automation, getcleverpet, fishbed and others.
- Technology-specific platforms are the platforms which take into account only specific set of devices. These platforms are sometimes closed, in the sense that they are based on the devices with proprietary technology. Examples of such platforms are Mbed, which supports only devices based on ARM Cortex-M microcontroller, Zatar, Nest and others.
- M2M connectivity providers offer connectivity as a service as a core service, with only a few other features, mostly related to data analytics. Their primary objective is data acquisition and analysis.
- Full scale generic IoT middlewares (for example, ThingWorx) provide full range of connectivity services, but they also facilitate the application development, based on data collected by the devices and transformed by analytical tools. Such development is possible by using integrated development environments (IDE), API's or even language interpreters.
- Some platforms offering supporting services were considered as important. They do not offer M2M connectivity services, so they are not IoT platforms. However, they offer functionality which can be useful for IoT scenarios. Examples of such platforms are GrooveStreams, a data analytics cloud platform, and ifttt, interoperability-as-a-service platform.

With the rise of IoT platforms, cross-platform interoperability and reuse is starting to emerge. There are cases where the domain-specific IoT platforms are made by using M2M connectivity providers. For example, getcleverpet, groovelabs.io and fishbit are implemented by using particle.io. Similarly, first cases of collaboration between platforms appear, with interoperability solutions. For example, ThingSpeak platform is connected to realtime.io. Further collaboration may be facilitated by increasing number of stable open-source solutions. In fact, they already exist on market (Kaaproject, LinkSmart) offering significant opportunities for development of complex systems over existing core communication management and communication platform. Finally, IoT ecosystem will certainly benefit from the further development of supporting services (interoperability-as-a-service, storage-as-a-service, data analytics and visualization, etc.).

HTTP and REST will probably be the way to communicate with devices, as well as between platforms. Guinard et al [4] found that RESTful Web Services are more suitable for programming access to IoT devices, than WS-* service architecture. However, it is highlighted that the latter are better choice when complex requirements related to advanced security and QoS are considered.

IoT at scale is a crucial technical requirement for further development, scalable performance (concurrency), storage and connectivity being the most important topics. Scaling at a client side is considered with least priority; devices are still the most sensitive and weakest components of IoT architecture in terms of reliability (power consumption).

A. *Issues and challenges of current cloud IoT platforms*

Centralized approach to managing IoT ecosystem may pose challenges to the devices' reusability in multiple contexts/applications, due to potential conflicts between clients. It may also affect the future of IoT ecosystem architecture and pose an approach characterized by exclusive ownership over a device, where services and data of this device would be offered through a central platform that controls that device, forming a network of networks. This may lead to application silos, with potential risk of interoperability issues. Is P2P communication between devices worth considering as an alternative, even through their digital identities stored in a cloud (like in Arrayent platform)? Obviously, current devices' energy consumption issue makes the case for centralized approach, but for how long?

Though convenient, turnkey solutions, such as the group of full scale generic IoT platforms may affect the development of IoT market. Namely, when considering current cloud computing applications, the only feature that distinguishes IoT platforms from the other cloud services platforms is M2M connectivity as a service. All the other services, such as analytics, visualization and application development are value-added, non-IoT core services and they may be provided by the third-parties. However, the use of non-core services by the M2M connectivity providers is not at the significant rate.

Finally, the third major issue is lack of support to complex data structures and business logics (beyond the level of simple triggers or rules), to be used for the development of applications, based on collected and/or transformed data.

V. Conclusions

Although Gartner's analysis of the emerging technologies positions IoT platforms at the very early phase of development, experiences from this survey show that cloud-based M2M connectivity services offer is well established. Some characteristic market niches are already recognizable, namely, M2M connectivity, data storage and analysis, data visualization, Interoperability-as-a-Service and others. What is clearly missing at this point is IoT ecosystem application building environment.

While the objective of this paper was to identify the gap in the current state of art of IoT platforms, comparing to the theoretical foundations and vision of IoT, its motivation was to setup the novel design of IoT platform which core feature will be exactly application development.

Based on the survey, following main principles for the development of formal model-driven IoT software execution platform (InoTEP) are defined:

- InoTEP is web application for devices in IoT which enables composition and realization of IoT scenarios, by using peer-to-peer approach (multiple InoTEP instances installed on multiple devices, communicating over REST).
- InoTEP provides Application-as-a-Service service which will interpret any formal model (RDF/RDFS/OWL ontology) in a runtime and deliver CRUD (create/read/update/delete data) application.
- InoTEP enables formal definition of the device's capability to sense and/or actuate, by using Capabilities ontology.
- InoTEP uses RDF as a transport protocol for communication between devices (over REST).
- InoTEP tries to match any data received through its own REST interface (external data), with domain and capabilities ontologies.

The above listed principles are further used in selection of the enablers of the key components of InoTEP. Application-as-a-Service component will be implemented by using OntoApp system [6]; W3C Sensor ontology [7] is being extended to develop a Capabilities Ontology; Active Semantic Model [8] approach will be used for a matching engine.

ACKNOWLEDGMENT

This work presented in this paper has been co-funded by the projects VIHOS (III41017, supported by the Government of Republic of Serbia), H2020-644715 "AquaSmart", supported by the European Commission and "Towards Context-aware Smart Cyber-Physical Ecosystems" (451-03-01765/2014-09/23, supported by Bilateral scientific co-operation programme - Serbia and Portugal).

REFERENCES

[1] Stankovic, J.A. (2014) Research Directions for the Internet of Things. Internet of Things Journal, 1(1)3-9

[2] Machina Research press release, Machine-to-Machine connections to hit 12 billion in 2020, generating EUR714 billion revenue. https://machinaresearch.com/static/media/uploads/machina_research_press_release_m2m_global_forecast_analysis_2010_20.pdf

[3] Nair, S., Why is "Internet of Everything" a future growth driver for Cisco? http://finance.yahoo.com/news/why-internet-everything-future-growth-130020843.html

[4] Gluhak, A., Krco, S., Nati, M., Pfisterer, D., Mitton, N. et al. A Survey on Facilities for Experimental Internet of Things Research. IEEE Communications Magazine, Institute of Electrical and Electronics Engineers, 2011, 49 (11), pp.58-67

[5] Stankovic, J.A. (2014) Research Directions for the Internet of Things. Internet of Things Journal, 1(1)3-9

[6] Zdravković, M., Trajanović, M. (2015) On the Runtime Models for Complex, Distributed and Aware Systems. 5th International Conference on Information Society and Technology (ICIST 2015), March, 8-11 2015, Kopaonik, Serbia. In: Zdravkovic, M., Trajanovic, M., Konjovic, Z. (Eds.): ICIST 2015 Proceedings, pp.235-240, 2015

[7] Compton, M., Barnaghi, P., Bermudez, L., Garcia-Castro, R., Corcho, O., Cox, S., Graybeal, J., Hauswirth, M., Henson, C., Herzog, A., Huang, V., Janowitz, K., Kelsey, D., Le Phuoc, D., Lefort, L., Leggieri, M., Neuhaus, H., Nikolov, A., Page, K., Passant, A., Sheth, A., Taylor, K.: The SSN Ontology of the W3C Semantic Sensor Network Incubator Group, Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 17, 25-32. (2012)

[8] Stojkovic, M., Trifunovic, M., Misic, D., Manic, M. (2015) Towards Analogy-Based Reasoning in Semantic Network. Computer Science and Information Systems, 12(3) 979–1008