This work-in-progress paper was presented as part of the main technical program at IFIP WMNC'2013

A Cloud Based - Architecture for Cost-Efficient Applications and Services Provisioning in Wireless Sensor Networks

Roch Glitho
Concordia University
Montreal
Canada

Monique Morrow
CISCO Systems
Zurich
Switzerland

Paul Polakos
CISCO Systems
New Jersey
USA

Abstract: **More and more wireless sensor networks (WSNs) are being deployed. However, provisioning WSN applications and services in a cost efficient manner remains an uphill task. In the current state of affairs, WSN applications and services make little re-use of third party modules due to the lack of appropriate discovery mechanisms. Furthermore, they are bundled with the WSNs in which they are provisioned and this precludes the use of the same WSN by other applications and services. Novel approaches are indeed required. Cloud computing is an emerging paradigm with three key aspects (Software as a Service – SaaS, Platform as a Service – PaaS, and Infrastructure as a Service – IaaS) and several inherent benefits (e.g. efficient use of resources, easy introduction of new applications and services). It can potentially aid in tackling the cost efficiency issue in WSN applications and services provisioning. Unfortunately, this area has not yet received much attention in the research community. This position paper pinpoints the challenges, reviews the state-of-the-art, sketches a new architecture and proposes a research agenda.**

Keywords—wireless sensor networks (WSN), cloud computing, Infrastructure as a Service, overlay networks, Platform as a Service, Software as a Service, applications and services provisioning, service delivery platforms

I. Introduction

Wireless Sensor Networks (WSNs) are composed of nodes that sense, compute and communicate [1]. More and more WSNs are deployed nowadays in a wide range of domains (e.g. health, agriculture). However, provisioning (i.e developing, deploying and managing) WSN applications and services in a cost efficient manner remains an uphill task. There are still several stumbling blocks.

Cloud computing is a promising paradigm for applications and services provisioning. It has many inherent advantages, such as the easy introduction of new applications, resource efficiency, and scalability. There is not yet a standard technical definition for cloud computing. However there is a consensus around the three key facets: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) [2].

This position paper sketches a novel cloud-based architecture for cost efficient applications and services provisioning in WSNs and proposes a research agenda. The next section discusses the overall challenges and illustrates them with a concrete WSN application scenario: brush fire detection and management. The shortcomings of the cloud architectures proposed to date for tackling the challenges are pinpointed. The fourth section sketches the cloud architecture we envision for the future and discusses the research agenda.

II. The Overall Challenges and the Brush Fire Detection and Management Application as Illustration

It is no easy task to provision WSNs' applications and services in a cost efficient manner. A first challenge is to decouple the applications and services from the WSNs that host them. Applications and services are usually bundled with the WSN infrastructure in which they are deployed; thereby precluding the use of the infrastructure by several other applications and services that could have reasonably been deployed in it. This is certainly not sustainable in the long term because it will lead to the proliferation of redundant WSNs.

Another challenge is to speed up WSNs' applications and services development and deployment. The process is still slow. Third-party applications have almost no opportunity to be re-used because there is virtually no framework that allows their discovery for composition purposes. Moreover the development and deployment environments usually target a limited range of developers who need to be acquainted with highly specialized and non-standard development and deployment environments and tools.

Yet another challenge is to make WSNs' applications and services available from any device, anywhere and anytime. Most WSN applications and services are developed using the classical client/server paradigm, and the client part is quite often too "heavy" for small footprint devices such as smart phones. Many of these applications and services are only accessible on-site. Furthermore, most of them are accessible by end-users only via a human-machine interface. Today's WSN applications and services do not offer the appropriate programmatic interfaces that would make them accessible by other applications via a machine-machine interface.

Let us illustrate these challenges by the hypothetical brush fire detection and management application. The application detects fire eruption and displays the fire contour, among other things. Reference [3] describes the semantic and discusses how the fire detection sensors deployed by private individuals could eventually be re-used by the city authorities if they decide to deploy such an application. We provide below our own analysis of how such an application would be provisioned today and thereby illustrate the inefficiency of the process and the challenges we just discussed.

In the current state of affairs, the authorities will have to deploy sensors all over the city, in both public places (e.g. streets, parks) and private places (e.g. private houses, private business premises). This will most probably lead to the deployment of redundant WSNs because some (if not most) of the private places may have already deployed their own fire-alert WSNs. Unfortunately, these WSNs cannot be re-used because they cannot host additional applications /tasks.

The application would certainly be developed and deployed by highly specialized developers. They will use the proprietary development and deployment environment that comes with the newly-purchased WSN. There will be little or no third-party application re-use because the existing development environments do not have the features to enable a large-scale discovery of potential building blocks that can be composed. More importantly, the application will not be accessible anywhere, anytime and from any device by other stakeholders (e.g. fire fighters, insurance companies) who could improve their efficacy by visioning the fire contour. It will be confined to the city authorities' premises.

III. TODAY'S CLOUD ARCHITECTURES FOR WSN APPLICATIONS AND SERVICES PROVISIONING

Cloud architectures can certainly aid in tackling the challenges discussed in the previous section. The proposed architectures focus on specific facets (i.e. IaaS, PaaS, SaaS) of cloud computing and do not even meet all the requirements associated with the specific facet they focus on. To the best of our knowledge, there has been no attempt to design an integrated architecture that tackles all the challenges. We first introduce cloud computing basics. The proposed IaaS, PaaS and SaaS centered architectures are then reviewed.

A. *The basics of cloud computing*

Cloud Computing offers a pay as you go model and relieves end-users from acquiring and hosting hardware and software resources. It refers to a framework that supports IaaS, PaaS, and SaaS. IaaS enables the on-demand acquisition of the physical and virtualized extents of computing, storage, network and software resources. PaaS supports the on-demand access to middleware service components (such as a message queuing or software development and testing platforms) that enables the development and management of applications and services. SaaS allows on-demand access to an instance of an application or service by end-users or other applications. Essentially, cloud computing has emerged as a technology that is changing the IT, datacenter, networking and applications and services' landscapes in a major way. A cloud service offers cloud services out of one or more data centers, where computing, storage, and network resources are offered on-demand to cloud service consumers.

However, it is important to stress that cloud computing is not necessarily applicable in its current form to WSNs. Let us illustrate this point by the IaaS aspect. IaaS implies the on-demand assignment of a physical resource (or set of resources) from a large pool of equivalent resources at the request of the cloud service consumer. In the typical model, the consumers are granted the use of the physical resource until their task is completed. The resource is then relinquished and returned to the pool. In WSNs, a distinction may exist in that while a set of sensors may be functionally identical (e.g. smoke detectors, heat sensors), they may not be operationally equivalent. For an illustration, it would not make sense to assign a sensor resource deployed to detect fire at location X to a consumer located at location Y, even if it is available for use.

B. *IaaS centered architectures*

A typical example is the Sensor-Cloud infrastructure proposed in reference [4]. The overall goal is to allow the transparent use of sensors (i.e. without being aware of the sensors' location and detailed specifications). There are three actors in the business model: the sensor owner, the infrastructure administrator and the end-user. The sensor owner allows the use of her/his sensors (by other end-users) through the Sensor-Cloud infrastructure. The administrator manages, among other aspects, the portal server, and the templates for registering physical sensors. The end-user owns applications and services that use the sensor data.

It is quite clear that this architecture cannot handle complex scenarios such as fire brush detection and management. In Sensor-Cloud, sensors can be used by a given application only if they have been freed by the application that was previously using them. End-users have to explicitly free sensors when they have finished using them. In the brush fire detection and management scenario however, sensors at private places will have to be used concurrently by the owners' application and the city authorities' application.

It should be stressed that there is a large body of literature on WSN virtualization, a topic highly pertinent to cloud IaaS. This literature covers both node level virtualization (e.g. [5, 6]) and network level virtualization (e.g. [7, 8]). Node level virtualization can be defined as the mechanisms that enable multiple applications to reside in and run concurrently on a single WSN node [9]. Network level virtualization on the other hand can be defined as mechanisms for enabling virtual sensor networks (VSNs). A VSN is a subset of nodes of a WSN dedicated to a certain application at a given time [7].

WSN virtualization as a concept is certainly more promising than Sensor-Cloud for tackling the IaaS aspects of scenarios such as the brush fire detection and management example. However it is still in its infancy and there are several open research issues. We will further discuss these issues in our research agenda, as WSN virtualization is one of the main items.

C. *PaaS- and SaaS-centered architectures*

PaaS- and SaaS-centered architectures remain the most popular options. The key idea is to collect data from the WSN that is outside the clouds, transfer that data to the cloud, and use the PaaS of the cloud to develop and manage the applications and services. These applications and services may then be offered as SaaS. So far, only relatively simple applications have been considered for these architectures.

There are several examples from the healthcare domain. In reference [10], patients wear textile sensors that collect bio-signals (e.g. heart rate, temperature) and motion data. The data is forwarded to the cloud via a mobile phone. The mobile phone and the sensors communicate by Bluetooth. Google Apps Engine is used at the PaaS level to develop and offer very simple applications (e.g. alert management, data overview) as SaaS to end users.

Reference [11] provides another example from the health domain. Patients' vital data is collected via a WSN connected to legacy medical devices. The data is then sent to the medical center's cloud for storage, processing and distribution. IEEE 802.11 is used between the sensor nodes and the cloud. A prototype has yet to be built and the plan is to build it using an open source, Open Nebula.

There is no lack of examples from other fields. Reference [12] discusses a tailing dam monitoring and pre-alarm system in mines. Relevant data is collected and transferred to a cloud, as in the health domain applications. A PaaS is then used to develop and manage the applications just as in the health domain.

None of the previously discussed architectures can satisfactorily tackle the PaaS and IaaS challenges. PaaS's (including Google Apps Engine) target experienced developers and novice developers cannot use them. The provided applications and services barely re-use third party applications. None of the PaaS systems (including Google Apps Engine) enables the large-scale discovery of third party applications (including applications that may reside in clouds) that may be composed to provide the targeted application. Furthermore, none of the applications and services offers a programmatic interface that will enable its use by other applications.

IV. OUR VISION FOR THE FUTURE: ARCHITECTURAL SKETCH AND RESEARCH AGENDA

Let us go back to the brush fire detection and management scenario. In our vision the city should be able to deploy its application, without re-deploying sensors in privatelocations. Furthermore, it should be possible for other stakeholders to deploy other new applications and services that re-use the very same sensors already deployed in public and private places. An example might be an application that detects fire, observes its contour and automatically dispatches robots to suppress it.

All these applications and services should be easily developed by both experienced and novice developers by composing building blocks discovered on the fly. Even a novice from an insurance company should, for instance, be able to develop an application that provides statistics on fire eruption in the city by simply "dragging and dropping". The fire contour display application developed by the city should of course be accessible anywhere, anytime by fire fighters. It should also be accessible via a programmatic machine to machine interface to the statistics application developed by an insurance company employee.

A. *Architectural sketch*

Figure 1 sketches the envisioned architecture. It is assumed that all third party applications are provided as cloud applications. This makes the architecture a full cloud-based architecture where we have WSN applications and services, cloud providers and third-party applications' cloud providers. An inter-cloud interactions framework allows the interactions between these clouds.

In the figure, there is a WSN cloud and there are other clouds. The other clouds provide the third-party applications that may be re-used by the WSN applications and services that reside in the WSN cloud. These WSN applications and services are developed and managed using a PaaS adapted to WSN reality. They are also exposed to end-users using an SaaS adapted to WSN reality. Furthermore, the IaaS adapted to WSN reality enables an efficient usage of resources, for instance, taking into account the fact that sensors residing in the cloud may be functionally equivalent but not operationally equivalent as mentioned earlier in this paper.

The inter-cloud interaction framework enables publication (e,g, third party application providers need to publish their applications), discovery (the WSN cloud provider needs to discover the applications published by third-party applications for applications' composition purposes) and a few other functionalities such as enabling the WSN cloud provider to instantiate a third-party application it utilizes.

Let us illustrate this by an application that detects fire and dispatches a fleet of robots to suppress it. The application will reside in the WSN cloud. It will be built by composing the following blocks: a fire detection fire contour algorithm processor and a robot dispatcher. These blocks will reside in other clouds. The WSN cloud, a fire detection cloud and the fire contour algorithm processor cloud will interact for publication, discovery and other purposes using the cloud interaction framework.

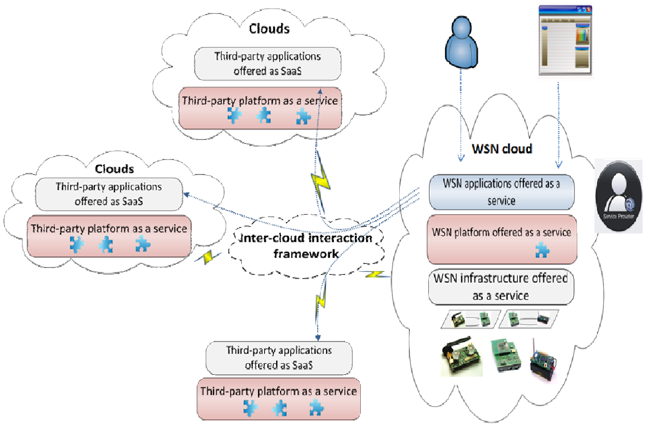

Figure – A cloud based architecture for cost efficient Applications and Services Provisioning in WSN.

B. Research agenda

1) Applicability of the cloud paradigm to WSNs

As briefly discussed in this paper, some of the assumptions behind cloud computing (e.g. resources that are functionally equivalent are operationally equivalent) do not hold in the WSN world. We will provide an exhaustive study of these assumptions and revisit the cloud paradigm as appropriate.

2) Inter-cloud interactions for WSN applications and services provisioning

Overlays' technologies could be used for the interactions. Reference [13] proposes the use of overlays for inter-cloud interactions. However, the work is at a very early stage, and moreover the focus is more on interactions for authentication, routing, delivery and registration. These interactions, although pertinent, do not correspond to the interactions in our proposed architecture. Other technologies such as service oriented architecture (SoA) could also be considered.

3) WSN offered as a Service (i.e. WSN virtualization)

WSN virtualization, a pillar of the proposed architecture, is a very rich research area. The business model is a motivator for eventual commercial acceptance and needs to be researched. Roles (e.g. physical WSN provider, virtual WSN provider, Individual sensor provider in crowd sensing) have to be defined and interactions sketched. Another research item is node level virtualization. The solutions provided so far in the literature are geared toward specific applications domains. Generic solutions need to be investigated. Yet another research item is network level virtualization. Most of the approaches proposed to date only consider the sequential execution of different applications. They do not address simultaneous execution. Combining node level and network level virtualization is also an interesting research avenue.

4) WSN platform as a Service – PaaS and WSN applications as Services – SaaS

Several research items need to be investigated at this level. A first item is platforms that are geared towards WSN application development and management and that cater to the needs of developers with different levels of expertise. Existing platforms (e.g. Google Apps Engine, Microsoft Azure) do not account for the specifics of WSN applications and services. Furthermore, they solely cater to the needs of somewhat experienced developers. This needs to be addressed. Yet another research item is the provision of programmatic interfaces to WSN applications and services to allow their use by other applications. Web services could be considered as enabling technologies.

V. CONCLUSIONS

This position paper has sketched a cloud-based architecture for cost efficient WSN applications and services provisioning. We have also proposed a research agenda. The challenges were identified and related work reviewed. The architecture we sketched is fully cloud-based and can potentially meet all the challenges. We will tackle the research issues of our agenda in future work.

REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, ”Wirelesssensor networks: a survey”, Computer Networks, vol. 38, no. 4, 2002,

[2] L. M. Vaquero et al., “A Break in the Clouds: Towards a Cloud Definition”, ACM SIGCOMM ComputerCommunication Review, Vol. 39, No1, January 2009pp. 393-422G.

[3] Amiya Bhattacharya, Meddage S. Fernando, and Partha Dasgupta.,”Community sensor grids: virtualization for sharing across domains”, 1st Workshop on Virtualization in Mobile Computing (MobiVirt ’08), ACM, New York, 2008, pp. 49-54

[4] M. Yuriyama and T. Kushida, Sensor-Cloud Infrastructure, 2010 13th International Conference on Network Based Information Systems

[5] P. Levis and D. Culler: ”Mat: A tiny virtual machine for sensor networks” In ASPLOSX: Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems, 2002, pp. 8595.

[6] J. Koshy and R. Pandey, ”VMSTAR: Synthesizing scalable runtime environments for sensor networks”, In SenSys05: Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, 2005 pp. 243254

[7] A. P. Jayasumana, Q. Han, and T. Illangasekare, ”Virtual sensor networksa resource efficient approach for concurrent applications,” In Proc. ITNG 2007, Las Vegas, Apr. 2007

[8] H. M. N. Dilum Bandara, A. P. Jayasumana and T. Illangasekare,”Cluster Tree Based Self Organization of Virtual Sensor Networks”, In Proc. IEEE Globecom workshop on Wireless Mesh and Sensor Networks, New Orleans, Nov. 2008

[9] Y. Yu et al., ”Supporting concurrent applications in wireless sensor networks”, In proceedings of the 4th International Conference on Embedded Networked Sensor Systems, SenSys06, Boulder, Colorado, 2006, pp.139-152

[10] C. Doukas and I. Maglogiannis, Managing Wearable Sensor Data through Cloud Computing, 2011 Third IEEE International Conference on Cloud Computing Technology and Science

[11] C.O Rolim et al., A Cloud Computing Solution for Patient’s Data Collection in Health Care Instutitions, Second International Conference on e-Health, Telemedecine and Social Medecine, 2010

[12] Sun et al., The Internet of Things (IOT) and Cloud Computing (CC) Based Tailings Dam Monitoring and Pre Alarm System in Mines, Elsevier Safe Science, Forthcoming (Electronic version available on the journal site)

[13] C. Shan et al, Inter-cloud Operations via NGSON, IEEE Communications Magazine, January 2012