



HAL
open science

Towards Bridging IoT and Cloud Services: Proposing Smartphones as Mobile and Autonomic Service Gateways

Roya Golchay, Frédéric Le Mouël, Stéphane Frénot, Julien Ponge

► **To cite this version:**

Roya Golchay, Frédéric Le Mouël, Stéphane Frénot, Julien Ponge. Towards Bridging IoT and Cloud Services: Proposing Smartphones as Mobile and Autonomic Service Gateways. Actes des 7ème Journées Francophones de la Mobilité et Ubiquité (UbiMob'2011), Jun 2011, Toulouse, France. pp.45–48. inria-00610537

HAL Id: inria-00610537

<https://hal.inria.fr/inria-00610537>

Submitted on 24 Jul 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards Bridging IoT and Cloud Services: Proposing Smartphones as Mobile and Autonomic Service Gateways

Position Paper

Roya Golchay
Université de Lyon, INRIA
INSA-Lyon, CITI
F-69621, Villeurbanne, France
roya.golchay@insa-lyon.fr

Frédéric Le Mouël
Université de Lyon, INRIA
INSA-Lyon, CITI
F-69621, Villeurbanne, France
frederic.le-mouel@insa-lyon.fr

Stéphane Frénot
Université de Lyon, INRIA
INSA-Lyon, CITI
F-69621, Villeurbanne, France
stephane.frenot@insa-lyon.fr

Julien Ponge
Université de Lyon, INRIA
INSA-Lyon, CITI
F-69621, Villeurbanne, France
julien.ponge@insa-lyon.fr

ABSTRACT

Computing is currently getting at the same time incredibly in the small with sensors/actuators embedded in our everyday objects and also greatly in the large with data and service clouds accessible anytime, anywhere. This Internet of Things is physically closed to the user but suffers from weak run-time execution environments. Cloud Environments provide powerful data storage and computing power but can not be easily accessed and integrate the final-user context-awareness. We consider smartphones are set to become the universal interface between these two worlds. In this position paper, we propose a middleware approach where smartphones provide service gateways to bridge the gap between IoT services and Cloud services. Since smartphones are mobile gateways, they should be able to (re)configure themselves according to their place, things discovered around, and their own resources such battery. Several issues are discussed: collaborative event-based context management, adaptive and opportunistic service deployment and invocation, multi-criteria (user- and performance-oriented) optimization decision algorithm.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*Internet of Things*; C.2.4 [Computer-Communication Networks]: Distributed Systems—*Cloud computing*; D.2.11 [Software Engineering]: Software Architectures—*Service-oriented architecture (SOA)*

General Terms

Algorithms, Management, Performance, Reliability

Keywords

Cloud Computing, Internet of Things, Service-oriented Applications, Middleware, Smart and Autonomic Gateways, Smartphone

1. INTRODUCTION

Computing is currently getting at the same time incredibly in the small with sensors/actuators embedded in our everyday objects and also greatly in the large with data and service clouds accessible anytime, anywhere. This Internet of Things is physically closed to the user but suffers from weak run-time execution environments. Cloud Environments virtualize and provide powerful data storage and computing power but can not be easily accessed and integrate the final-user context-awareness.

In parallel with these run-time environment evolutions, the service-oriented programming and architectures are also important evolutions in software engineering. These paradigms allow to easily split applications, enable a great reuse and (sometimes) handle (some) communication protocol issues. New run-time challenges are still raised especially with IoT and Cloud environments such as service deployment mechanisms, service placement, service invocation scheduling, etc.

We consider smartphones are set to become the universal interface between Internet of Things and Cloud Computing worlds, as shown in Figure 1. In this position paper, we propose a middleware approach where smartphones provide service gateways to bridge the gap between IoT services and Cloud services. Since smartphones are mobile gateways, they should be able to (re)configure themselves according to their place, things discovered around, and their own resources such battery. Several issues are discussed to implement a smart and autonomic service gateway: collaborative event-based context management, adaptive and opportunis-

Figure 1: Smartphone Service Gateways Bridging IoT and Cloud Services

tic service deployment and invocation, multi-criteria (user- and performance-oriented) optimization decision algorithm.

The rest of this paper is organized as follow. Next section presents related works. An overview of our proposed solution is given in section 3 and smart and autonomic mechanisms are detailed in section 4. Finally the last section states our current and future works.

2. RELATED WORK

While talking about mobile environment the first thing that attracts attention is resource-poor devices compared with fixed stations. Mobile phones are limited in computation power, available memory and battery life. The wireless connections with different available bandwidth and probability of disconnection and reconnection are another important point in these environments. There is a considerable amount of research on how to address these mobility issues by migrating data and computation to distributed systems [12]. This issue is now raised by Internet of Things where things are even more constrained. To address the low energy power issue, [11] proposed using cloud computing. According to this article sending computation to the cloud will save energy on smart phone. The amount of saved energy is depends on the wireless bandwidth, the amount of data to be transmitted to cloud and also the application type if it is real-time data application or not. In order to use the cloud more efficiently, how to distribute the application is very important. Wishbone [13] is a system that takes a data-flow graph of operators and produces an optimal partitioning. The partitioning problem is to find a cut of data-flow graph, where its vertices are stream operators and its edges are streams. Edge and vertex weights represent bandwidth and CPU utilisation respectively. The partitioning algorithm models the program as an integer linear program that minimizes a linear combination of network bandwidth and CPU load. The notion of data-flow graph is also used in [8] where the presented middleware platform can automatically distribute different layers of an application between the phone and the cloud. In this graph the vertexes are software modules where service dependency between them represents by

the edges. The optimal cut of the graph will minimize or maximizes the objective function that can be the end-to-end interaction time between a phone and a server, the amount of exchanged data or complete the execution in less than a predefined time. The layered architecture is also considered in [9] where presented tenet architecture for tiered sensor networks networks simplified application development. This architecture composed of two tiers: a lower tier consisting of motes, which enable flexible deployment of dense instruction, and upper tier containing fewer, relatively less constrained nodes. Another three tiers management architecture is used in [7] where a service-oriented framework is introduced to simplify the development and run-time adaptive support of autonomic pervasive applications. A middleware is proposed to serve as a framework to host autonomic home applications. A distributed resource model and management tailored for deployment of adaptive services in a mobile environment is also discussed in [1]. This research is focuses on the resources directly surrounding one mobile user. The presented distributed resource management framework is used to monitor the context and dynamically respond to changes by selecting the most appropriate application variant which called self-adaption mechanism. Although the distributed powerful resources in the cloud seems useful to cover the resource constraints of small components in Internet of Things, absence of context-awareness in the cloud make it difficult to communicate with it for the users. Several bridging approaches exist [5] but a link is required to handle communication between almost physical environment of IoT and virtual environment of the cloud.

3. SOLUTION OVERVIEW

We consider service-oriented applications modeled as a graph where nodes are services and vertices are service dependencies. Two service dependencies are possible: static deployment dependency and dynamic run-time invocations.

Figure 2: Smartphone Service Gateway - Conceptual Model

For the smartphone service gateway we propose, as shown in Figure 1, the major challenge consists in deploying, invoking, (and eventually replicating) services of an application by integrating (i) existing light-services on communicating objects from IoT, (ii) mid-services on the smartphones closer to

the end-users, (iii) heavy-services distributed in the Cloud.

The proposed conceptual model in Figure 2 has this service graph in input, applies a multi-criteria decision algorithm and realizes deployment/invocation/(replication) actions to schedule/coordinate the service choreography. The decision algorithm takes into account different context inputs extracted from the services (size, performance, etc), from platforms (mobility, location, etc) and from the user (preferences, profile, etc).

This process is autonomic because we think a smartphone has especially to be self-managed, without the user intervention. It is however important the decision algorithm to be user-oriented because a smartphone is really an end-user terminal. The process includes so a retro-action loop from service uses/monitoring feedbacks. This retro-action loop allows to automatically modify the user profile, to change the algorithm inputs by reannotating the service graph, or to adapt the decision algorithm it-self.

4. SMART AND AUTONOMIC GATEWAY MECHANISMS

After analysing the properties of autonomic systems, we propose several mechanisms implemented in a service-oriented middleware approach (cf Figure 3).

Figure 3: Smartphone Service Gateway - Architecture Overview

Aware property

1. **Event-based context management:** The context management is crucial in an autonomous system while these inputs are conditioning the decision. In our approach, we focus the context monitoring on services: their performance, location, mobility, etc. In very highly changing environments such as IoT and Cloud ones, to have a constantly reevaluating algorithm, we use a *Listener/Event Notifier* approach for each changes. Pertinently qualifying the moment to send an event is critical on a smartphone because a too active monitoring leads to a battery run off and a too passive monitoring do not produce a accurate view of the physical world. We will examine context aggregation and filtering such as in [14].

2. **Collaborative services repositories:** Services running on a smartphone can be available for other smartphones in vicinity. Each smartphone holds a local *Repository Service* that can be discovered by using different discovery protocols. Smartphones can then collaborate, for example, to use all the same service and reduce battery consumption of participants. Aimed collaborative approaches are not those of distributed systems with strong consensus or transaction problems, either are geographically-closed and short-lived collaborations with only local stability.

Adaptive property

3. **Adaptive, delayed and opportunistic service deployment and invocation:** Deployment and coordination mechanisms need to be adaptive to allow migration or call redirection according to smartphones mobility and/or disconnections. We will use a DTN communication stack implementation (Disrupted-Tolerant Network) in the *Communication Service* to provide on-demand deployment and most of all - spontaneous and opportunistic deployment such as in [10]. The opportunistic deployment can for instance be implemented by caching services prioritarily from well-known users or from users sharing common interests such as in Social Networks.

Automatic property

4. **Smart multi-criteria decision algorithm:** Our Service Gateway bridging several worlds - IoT, Cloud, Smartphones and End-users -, the decision algorithm implemented in the *Choreography Engine Service* has to consider contextual events from the physical environment such as the service location or mobility, from the virtual environment such as the service execution performances, from platforms such as the battery level and from user such as the service use statistics. This problem of solving multiple constraints is NP-complete but several approaches seems very interesting for mobile environments such as approximation algorithms for constrained knapsack problems [6] or biologically-inspired algorithms [15, 2]. Using the feedback control loop, we will particularly enrich them with a smart learning of the user profile.

5. CURRENT AND FUTURE WORK

A prototype - AxSeL¹, A conteXtual SErvice Loader - was implemented based on a Felix², one Java implementation of the OSGi service-oriented specification³ [3, 4]. This prototype can run on constrained devices and locally manages service-oriented application deployment according to hardware resources. It has been tested with a PDF reader application.

As immediate extensions, we will improve the prototype and focus on Smart Building domain with the INSA Lyon

¹<http://amazones.gforge.inria.fr/>, INRIA Amazones Team Forge

²<http://felix.apache.org/>

³<http://www.osgi.org/>

project: “The Smart Chappe Building: a Platform for Contextual Services”⁴. This platform will allow us to demonstrate the integration and detail (i) the interaction between smartphones and temperature/presence sensors in the building, (ii) the interaction between smartphones and static Wifi hotspots and/or mobile service gateways and (iii) the storage and computation migration of sensed data collected in the building to a cloud.

6. REFERENCES

- [1] M. Alia, S. Hallsteinsen, N. Paspallis, and F. Eliassen. Managing distributed adaptation of mobile applications. In *Proceedings of the 7th IFIP WG 6.1 international conference on Distributed Applications and Interoperable Systems*, DAIS '07, pages 104–118, Berlin, Heidelberg, 2007. Springer-Verlag.
- [2] E. Altman, P. Dini, D. Miorandi, and D. Schreckling, editors. *Paradigms for Biologically-Inspired Autonomic Networks and Services – The BIONETS Project eBook*. BIONETS Consortium, 2010. Deliverable D0.2.3.
- [3] A. Ben Hamida, F. Le Mouël, S. Frénot, and M. Ben Ahmed. A graph-based approach for contextual service loading in pervasive environments. In *Proceedings of the 10th International Symposium on Distributed Objects and Applications, DOA '2008*, volume 5331 of *Lecture Notes in Computer Science*, pages 589–606. Springer Verlag, Monterrey, Mexico, Nov. 2008.
- [4] A. Ben Hamida, F. Le Mouël, S. Frénot, and M. Ben Ahmed. Déploiement adaptatif d’applications orientées services sur environnements contraints. *Computer Science and Technology / Technique et Science Informatiques*, 30(1):59–91, 2011.
- [5] T. Bissyandé, L. Réveillère, Y.-D. Bromberg, J. Lawall, and G. Muller. Bridging the gap between legacy services and web services. In *Proceedings of the ACM/IFIP/USENIX 11th international conference on Middleware, Middleware '10*, volume 6452 of *Lecture Notes in Computer Science*, pages 273–292. Springer Berlin / Heidelberg, 2010.
- [6] G. Borradaile, B. Heeringa, and G. T. Wilfong. Approximation algorithms for constrained knapsack problems. *CoRR*, abs/0910.0777, 2009. Revised version 2 (Feb 2010).
- [7] J. Bourcier, A. Diaconescu, P. Lalanda, and J. A. McCann. AutoHome: An autonomic management framework for pervasive home applications. *ACM Trans. Auton. Adapt. Syst.*, 6:8:1–8:10, February 2011.
- [8] I. Giurgiu, O. Riva, D. Juric, I. Krivulev, and G. Alonso. Calling the cloud: Enabling mobile phones as interfaces to cloud applications. In *Proceedings of the ACM/IFIP/USENIX 10th international conference on Middleware, Middleware '09*, volume 5896 of *Lecture Notes in Computer Science*, pages 83–102. Springer Berlin / Heidelberg, 2009.
- [9] O. Gnawali, K.-Y. Jang, J. Paek, M. Vieira, R. Govindan, B. Greenstein, A. Joki, D. Estrin, and E. Kohler. The tenet architecture for tiered sensor networks. In *Proceedings of the 4th international conference on Embedded Networked Sensor Systems*, SenSys '06, pages 153–166, New York, NY, USA, 2006. ACM.
- [10] F. Guidec, N. L. Sommer, and Y. Mahéo. Opportunistic software deployment in disconnected mobile ad hoc networks. *IJHCR*, 1(1):24–42, 2010.
- [11] K. Kumar and Y.-H. Lu. Cloud computing for mobile users: Can offloading computation save energy? *Computer*, 43(4):51–56, april 2010.
- [12] F. Le Mouël, F. André, and M. Segarra. AeDEn: An adaptive framework for dynamic distribution over mobile environments. *Annals of Telecommunications*, 57(11–12):1124–1148, Nov. 2002.
- [13] R. Newton, S. Toledo, L. Girod, H. Balakrishnan, and S. Madden. Wishbone: profile-based partitioning for sensornet applications. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, NSDI '09, pages 395–408, Berkeley, CA, USA, 2009. USENIX Association.
- [14] D. Riboni and C. Bettini. COSAR: hybrid reasoning for context-aware activity recognition. *Personal and Ubiquitous Computing*, 15(3):271–289, 2011.
- [15] H. Wada, J. Suzuki, Y. Yamano, and K. Oba. Evolutionary deployment optimization for service-oriented clouds. *Softw. Pract. Exper.*, 41:469–493, April 2011.

⁴<http://telecom.insa-lyon.fr/content/plateforme-smart-chappe-building>