HAL
open science

Extending Ambient Intelligence to the Internet of Things: New Challenges for QoC Management

Pierrick Marie, Thierry Desprats, Sophie Chabridon, Michelle Sibilla

To cite this version:

Pierrick Marie, Thierry Desprats, Sophie Chabridon, Michelle Sibilla. Extending Ambient Intelligence to the Internet of Things: New Challenges for QoC Management. International Conference on Ubiquitous Computing and Ambient Intelligence - UCAmI 2014, Dec 2014, Belfast, United Kingdom. pp. 224-231. hal-01148836

HAL Id: hal-01148836

https://hal.science/hal-01148836v1

Submitted on 5 May 2015

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.

Open Archive TOULOUSE Archive Ouverte (OATAO)
OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in : http://oatao.univ-toulouse.fr/
Eprints ID : 13258

To link to this article : DOI :10.1007/978-3-319-13102-3_37
URL : http://dx.doi.org/10.1007/978-3-319-13102-3_37

To cite this version : Marie, Pierrick and Desprats, Thierry and Chabridon, Sophie and Sibilla, Michelle *Extending Ambient Intelligence to the Internet of Things: New Challenges for QoC Management*. (2014) In: International Conference on Ubiquitous Computing and Ambient Intelligence - UCAmI 2014, 2 December 2014 - 5 December 2014 (Belfast, United Kingdom).

Any correspondance concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr

Extending Ambient Intelligence to the Internet of Things: New Challenges for QoC Management

Pierrick Marie[1], Thierry Desprats[1], Sophie Chabridon[2], and Michelle Sibilla[1]

[1] IRIT UMR 5505 Université Paul Sabatier, 31062 Toulouse, France
{Firstname.Name}@irit.fr
[2] Institut Mines-Télécom/Télécom SudParis, CNRS UMR 5157, 91011 Évry, France
Sophie.Chabridon@telecom-sudparis.eu

Abstract. Quality of Context (QoC) awareness is recognized as a key point for the success of context-aware computing solutions. At a time where the Internet of Things, Cloud Computing, and Ambient Intelligence paradigms bring together new opportunities for more complex context computation, the next generation of Multiscale Distributed Context Managers (MDCM) is facing new challenges concerning QoC management. This paper presents how our QoCIM framework can help application developers to manage the whole QoC life-cycle by providing genericity, openness and uniformity. Its usages are illustrated, both at design time and at runtime, in the case of an urban pollution context- and QoC-aware scenario.

Keywords: quality of context, quality criterion, context management, meta-modeling, information model.

1 Introduction

Context-aware applications become more and more common. These applications consume context information extracted from local ambient data, users profiles, heterogeneous and spatially distributed sensors. The behaviour of context-aware applications is strongly affected by the quality of the context information (QoC). QoC-aware applications require a fine and efficient management of the QoC they rely on. QoC is related to any information that describes the quality of context information as defined by [4]. A solution to handle the QoC is to use context managers. Context managers support context information throughout their life cycle. The life cycle of a piece of context information starts when it gets collected by a sensor and ends at its consumption by a context-aware application. A bad quality of context could lead applications to wrong decisions and irrelevant reactions. That is why context managers must take into account QoC at each step of the context information life cycle.

Extending the scope of context managers from local ambient environment to the Internet of Things (IoT) introduces a spatio-temporal decoupling between context producers, such as embedded sensors, and context consumers such as context-aware applications running on mobile devices. Consequently new

challenges arise in order to guarantee the effectiveness and the efficiency of the new generation of context managers, corresponding to Multiscale Distributed Context Managers (MDCM). Such MDCMs must be deployed over various devices or servers, spread across heterogeneous networks or clouds. In addition to the classical key points used for the successful determination of the behaviour of context-aware applications, QoC can contribute to both the effectiveness and efficiency of these kind of context managers.

MDCMs being distributed over the Internet, this implies that there is no more possibility to establish a kind of "one-to-one QoC-based contract" between context producers and consumers, but QoC requirements and QoC guarantees still exist. Context consumers express QoC requirements about the quality they need and symmetrically, context producers express QoC guarantees about their capabilities. According to the expectations of the consumers, MDCMs have to deliver context information with the appropriate QoC. It requires the ability to express filters based on well-defined QoC criteria that may be used within MDCMs to optimize context data routing. Because one of the main functionalities of context managers is to process context information (by aggregation, inference, ...), they also have to manipulate QoC during the execution of these operations. That is why MDCMs should be extensible by enabling the definition and the computation algorithm of any QoC criterion.

This paper presents our QoCIM (Quality of Context Information Model) framework and answers the following questions, highlighted by an urban pollution scenario: (1.) How to take into account the QoC guarantees of producers? (2.) How to apply processes on context and QoC to infer high level information? (3.) How to fulfil the QoC requirements of the context consumers? (4.) How to apply QoC based dissemination to deliver the appropriate context?

The remainder of this paper is structured as follows: Section 2 motivates the use of MDCMs with a fictional scenario inspired from an existing concern: the urban pollution. Related and previous works dealing with context managers and quality of context are presented in Section 3. Section 4 presents the software solution we have built to handle the QoC within MDCMs. Section 5 summarizes our contributions and gives some perspectives.

2 An Urban Pollution Measurement Scenario

This Section proposes a fictional scenario inspired from the citizens' concern on urban pollution and illustrated in Figure 1. A polluted city wants to inform its citizens in real-time about the pollution level in the streets. The purpose is to offer solutions, based on QoC-aware applications, to avoid the most polluted streets when people are walking or riding a bike, for example. To realize it, the city deploys an MDCM to collect as raw data pollution measurements (*e.g.* Carbon Monoxide: 4.6 ppm) and the location of these measurements. Two QoC criteria, the precision defined as "how close together or how repeatable the results from a measurement are" [6] and the freshness defined as "the time that elapses between the determination of context information and its delivery to a

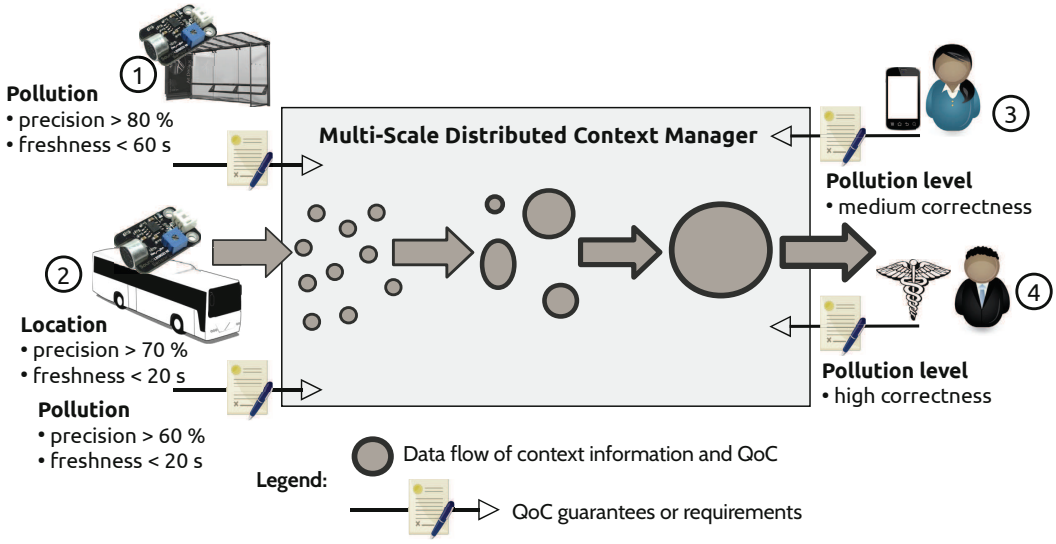

Fig. 1. QoC for the pollution measurement scenario

requester" [4] are also collected by the MDCM as QoC meta-data. The context information, offered by the city services, must be understandable and easy-to-use with a high abstraction level. That is why the raw data are then aggregated by the MDCM to produce a new context information, the pollution level, and a new QoC criterion, the correctness. The pollution level is separately computed for each street with all the available measurements made in the respective street. Inspired from the Air Quality Index[1]the pollution level is divided into six levels: *good, moderate, unhealthy for sensitive groups, unhealthy, very unhealthy and hazardous*. The correctness is defined by [7] as "the probability that an instance of context accurately represents the corresponding real world situation, as assessed by the context source, at the time it was determined" and possesses three values: *low, medium and high*. *Low* means the pollution level is not reliable while *high* means the pollution level is very reliable. The MDCM also tries to predict the coming pollution level and its correctness a few hours in advance, like the forecast weather system to help citizens prepare their travels.

① The city equips its *bus stations* with sophisticated pollution sensors and an embedded dedicated software. The pollution sensor performs accurate measurements ($\geqslant$ 80% of precision), but produces measurements, in the worst case, only every 60 seconds. The software is in charge to evaluate the precision of the measurements, and provide the MDCM the pollution as context information and the evaluation of the precision as QoC indicator. The QoC guarantees of the bus stations are therefore: precision $\geqslant$ 80% and freshness $\leqslant$ 60 s.

② *The buses* are also equipped with a pollution sensor and an embedded software. This sensor placed is faster than the sensor placed on the bus station; it produces a measurement, in the worst case, every 20 seconds, but it is less accurate ($\geqslant$ 50% of precision). The buses still in motion, then a GPS is installed into the buses and computes their location with a medium precision (almost 10

[1] Reporting the daily air quality: `www.airnow.gov/index.cfm?action=pubs.index`

meters). The embedded software supplies the MDCM with the pollution and the location of the bus as context information. The software is also in charge to evaluate and offer to the MDCM the precision of the pollution and the precision of the location as QoC meta-data. The buses are therefore committed to provide their pollution measurements with at least precision $\geqslant 50\%$ and freshness $\leqslant 20$ s. In the same time, they are also committed to provide their location with at least precision $\geqslant 70\%$ and the same freshness.

③ *A general mass-market mobile application* is developed on smart-phones to monitor the location of the users and requests the MDCM to get the pollution level of the streets where the users are. The application notifies them when the pollution is too high and suggests to go in a close and less polluted street if there is one. The application must not indicate wrong informations to users, that is why it needs the pollution level with, at least, a medium correctness. By this way, the notifications of the application do not supply wrong information. This application uses QoC to be more usable and to not display useless information. It also benefits from remote geographical context data that are not available from the ambient space to make relevant recommendations.

④ *A health care application* is provided to asthmatic people. The purpose of this application is to suggest users the best way to avoid the polluted streets. When users walk around the city, the application gets from the MDCM the current and forecast pollution level and suggests the best path to reach their destinations. The application is critical to asthmatic people, that is why the application always requires context information with the maximal correctness. If the MDCM is not able to provide this quality, the application still suggests an alternative paths but warns users about the risks to have an allergy attack and recommends to go in a safe zone until a new relevant path appears. This application operates with geographical and temporal remote context information associated to QoC levels to suggest safe paths and to avoid health troubles.

3 Related and Previous Works

The AWARENESS project [7] identifies three reasons for taking into account the QoC within context managers: (i) "application adaptation" (ii) "middleware efficiency" (iii) "users' privacy enforcement". The authors use five indicators to define QoC and point out the need for a formal methodology for QoC evaluation allowing to share clearly and unequivocally QoC values. They propose that all the parties come to an agreement on QoC before sharing any context information.

The COSMOS project [1] provides mechanisms for the efficient management of QoC. COSMOS is a context manager and handles the QoC provided to applications through contracts. A contract is established between a context-aware application and a context source. The contract defines the QoC level that the context-aware application requires. The COSMOS project does not provide a formal methodology to evaluate QoC as [7] evoked.

Bellavista et al. [3] propose a comparative analysis of context data distribution solutions. The authors propose a generic logical view where they distinguish the

context data delivery layer from the context data management layer. The context data delivery layer focuses on disseminating and routing data whereas the context data management layer focuses on the processing and the representation of data. Then, a runtime adaptation support aims to configure and reconfigure the layers. One of the conclusions of [3] states that providing a common definition of the list of criteria used to qualify the context information within the context managers is still an open problem.

Only a few context managers with a QoC management capacity have already been proposed. But, either they still have to fully integrate QoC management [7] or their definition of QoC is not formally specified [1]. One of the objectives of the INCOME project [2] is to fill these gaps by integrating a formal definition of QoC within MDCMs. The purpose of the project is to control the whole life cycle of context information and its associated QoC, from the production to the consumption including the processing. The project intends to manage context information coming from ambient networks, the IoT or clouds.

4 Implementation of the Pollution Scenario with QoCIM

As we proposed in [5], the QoC Information Model (QoCIM) offers a unified solution to model heterogeneous QoC criteria. Figure 2 contains, as a reminder, an overview of QoCIM. QoICM is a meta-model able to support the design and the representation of information about any QoC criterion.

Our approach consists in providing a graphical editor dedicated to graphically produce, at design time, new UML class diagrams of QoC criteria. With the editor, it is possible to edit the criteria by adding new definitions, new descriptions or new values of QoC. The editor also enables to define composite criterion depending on other already defined criteria. For that purpose, we modelled QoCIM as an Ecore model based on the EMF technology[2] and we then developed a graphical editor with the Sirius technology[3].

At programming time, developers of QoC-aware applications choose the required QoC criteria and use the editor to generate the corresponding code. The generated code is able to manage the QoC within context producers or consumers. The link between the value of the QoC and the context information is isolated into an empty method. So, developers just have to complete this method to manage the QoC within the QoC-aware application. In addition, sharing the QoC meta-data with the MDCM is much easier because the QoCIM framework is able to serialize any QoC criterion into XML documents. We developed the module to generate source code with the Acceleo technology[4].

As described in Figure 2, the QoCIM framework operates in four steps.
① *context acquisition* is about QoC guarantees and context acquisition.

[2] EMF technology, provided by the Eclipse foundation: `www.eclipse.org/modeling/emf`
[3] Sirius project, available on the Eclipse platform: `www.eclipse.org/sirius`
[4] Acceleo project, available on the Eclipse platform: `www.eclipse.org/acceleo`

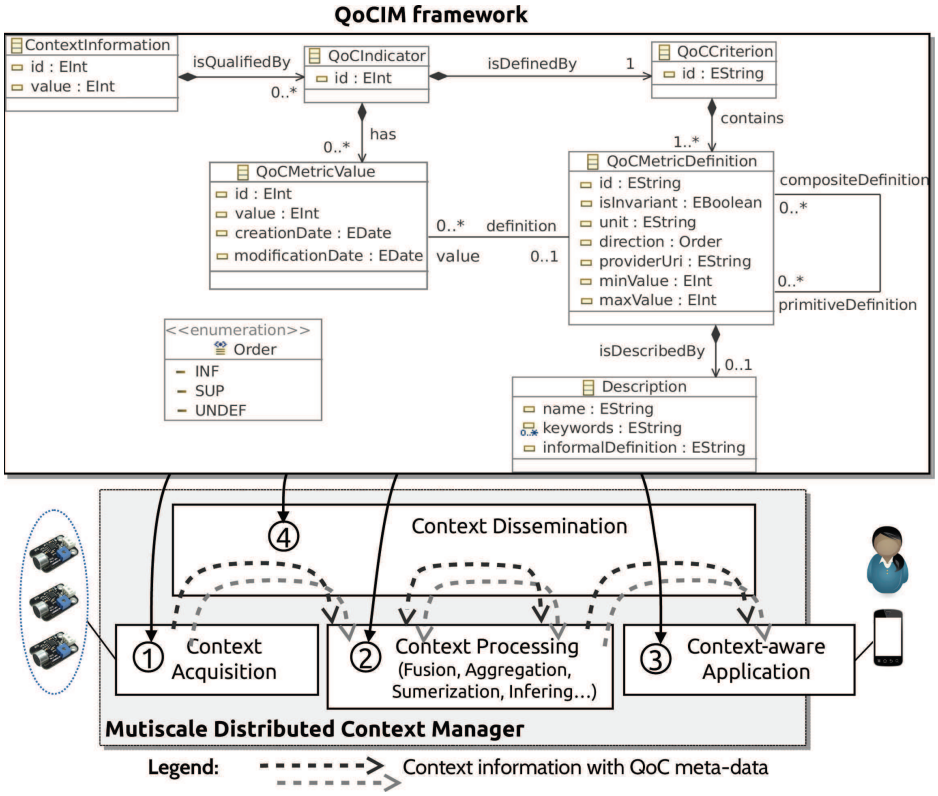

Fig. 2. The four points where QoCIM [5] operates

Communications between context providers and the MDCM are divided in two steps. During the first one, context providers inform the MDCM about their QoC guarantees. In the pollution scenario, the bus stations guarantee their information with a precision of at least 80%. In the second step, context providers publish information with QoC meta-data according to their guarantees.

② *QoC meta-data processing* infers new high informations. Because mobile phones could not process all the measurements made by the sensors, the MDCM provides aggregated information that summarize several raw data. Consequently, the abstraction level of the QoC increases and in our scenario the applications just receive the correctness composite criterion. Figure 3 is the representation of the correctness criterion made with our graphical editor. Combined with the Listing 1.1, it illustrates how the high value of the correctness can be computed within the MDCM. The QoC can also act for the context processing. For example, weighted arithmetic mean aggregates pollution measurements where high coefficient is associated to information with high precision.

③ *context aware applications* express their QoC requirements and consume context information. As for the context acquisition, communications between context consumers and the MDCM are also divided in two steps. During the first one, context consumers negotiate with the MDCM their QoC requirements. In our scenario, the health care application requires context with high correctness. If the MDCM is not able to fulfil the requirements, the application may change them. In the second step, the MDCM delivers to the application the context information with the QoC level it requires.

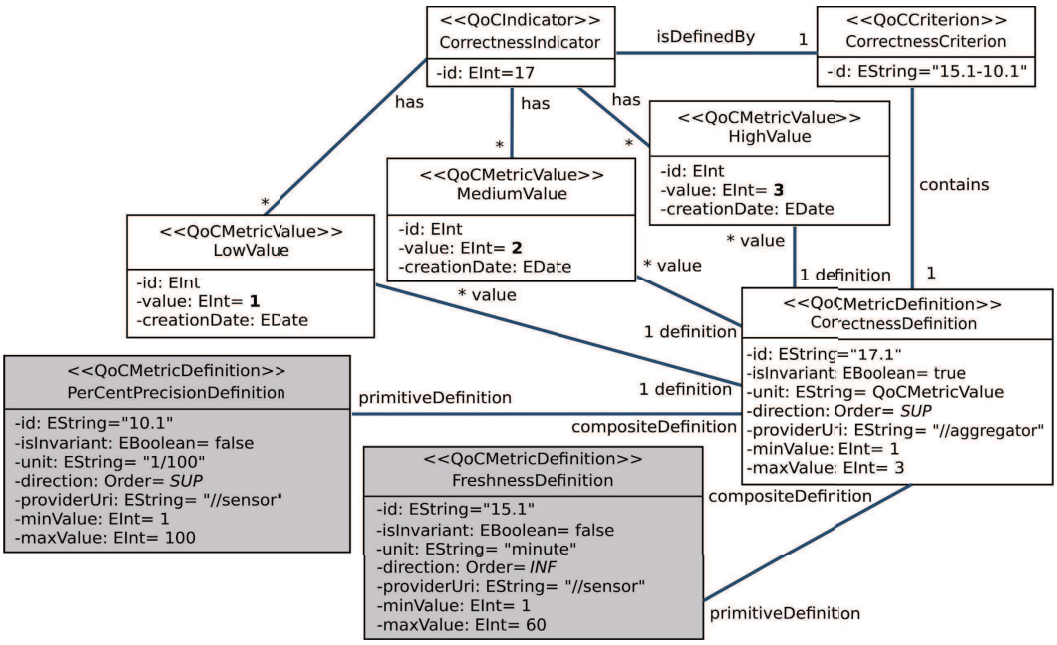

Fig. 3. QoCIM based definition of the composite criterion

```
context CorrectnessDefinition::value(): HighValue
    pre: self.PerCentPrecisionDefinition.QoCMetricValue.value >=
        85 * 100 / self.PrecisionDefinition.maxValue
    pre: self.FreshnessDefinition.QoCMetricValue.value <=
        15 * 100 /  self.FreshnessDefinition.maxValue
```

Listing 1.1. Example of OCL consraints used to produce the correctness

④ *context dissemination* is the key feature for providing, in our scenario, the health care application with the suitable context information that it needs. Therefore, the efficiency of the MDCM strongly depends on the QoC-aware routing system used to forward information. Based on the QoC requirements and guarantees expressed by all the entities connected to the MDCM, the routing system dynamically handles, at run time, advertisement and subscription filters. Those filters are composed of many constraints relative to the value of context information or the associated QoC metadata. Following the constraints contained within the filters, the routing system is able to decide whether a context information must be forwarded or not. With the uniformity offered by QoCIM, writing and interpret constraints concerning the QoC metadata is simplified. Indeed, because all the entities share the same QoC model, QoCIM, and our implementation of QoCIM can be serialize as XML documents. Developers of entities just have to write, with a programming language, their expectations about any attributes of QoCIM and then, the routing system is able to analyse the filters and correctly rout the context information. For example, in our scenario, the subscription filter relative to the health care application has constraints to only receive context information with QoC metadata composed of *CorrectnessDefinition*, identified with the attribute *id* equals to 17.1 and *HighValue* identified with the attribute *value* equals to 3.

5 Conclusion and Perspectives

Although several works on QoC modelling and management have been achieved over the past decade, no consensual proposition has emerged. This article illustrates QoCIM, the QoC Information Model we proposed as a generic, expressive and computable QoC information model to be used at any time during the QoC life-cycle management. QoCIM is dedicated to handle any QoC criterion within context managers and context-aware applications. It is able to qualify a piece of context information with different QoC criteria. A same QoC criterion can be reused to qualify different pieces of context information. Sharing the definition of QoC based on the same core concepts, QoCIM offers a common language to express requirements and guarantees for producers and consumers of context. By this way, MDCMs are able to match the needs of these producers and consumers and to evaluate QoC all along the life cycle of context information. The software tool chain we provide facilitates the development of QoC-aware applications. It offers a solution to easily create and use a collection of definitions of QoC criteria derived from QoCIM. Our future works will consist in improving our graphical editor and add the possibility to directly edit QoC guarantees, QoC requirements, and filters used for the dissemination. This will be possible with a Domain Specific Language editor based on QoCIM.

Acknowledgment. This work is part of the French National Research Agency (ANR) project INCOME (ANR-11-INFR-009, 2012-2015).

References

1. Abid, Z., Chabridon, S., Conan, D.: A framework for quality of context management. In: Rothermel, K., Fritsch, D., Blochinger, W., Dürr, F. (eds.) QuaCon 2009. LNCS, vol. 5786, pp. 120–131. Springer, Heidelberg (2009)
2. Arcangeli, J.-P., et al.: INCOME – multi-scale context management for the internet of things. In: Paternò, F., de Ruyter, B., Markopoulos, P., Santoro, C., van Loenen, E., Luyten, K. (eds.) AmI 2012. LNCS, vol. 7683, pp. 338–347. Springer, Heidelberg (2012)
3. Bellavista, P., Corradi, A., Fanelli, M., Foschini, L.: A Survey of Context Data Distribution for Mobile Ubiquitous Systems. ACM Computing Surveys 44(4), 24:1–24:45 (2012)
4. Buchholz, T., Schiffers, M.: Quality of context: What it is and why we need it. In: Proceedings of the 10th Workshop of the OpenView University Association (2003)
5. Marie, P., Desprats, T., Chabridon, S., Sibilla, M.: QoCIM: A meta-model for quality of context. In: Brézillon, P., Blackburn, P., Dapoigny, R. (eds.) CONTEXT 2013. LNCS, vol. 8175, pp. 302–315. Springer, Heidelberg (2013)
6. Neisse, R.: Trust and Privacy Management Support for Context-Aware Service Platforms. Ph.D. thesis, University of Twente, Enschede (2012)
7. Sheikh, K., Wegdam, M., Van Sinderen, M.: Middleware support for quality of context in pervasive context-aware systems. In: Pervasive Computing and Communications Workshops (2007)