KnE Engineering

ICEUBI2019
International Congress on Engineering — Engineering for Evolution
Volume 2020

Knowledge E
enriching | engaging | empowering

Conference Paper

Study of a Context Quality Model for UbiPri Middleware

Diandre de Paula[1], Daniel Saraiva[2], Nuno Garcia[3], and Valderi Leithardt[4]

[1]Universidade da Beira Interior, Departamento de Informática, 6201-001 Covilhã, Portugal
[2]Universidade da Beira Interior, Departamento de Informática e Instituto de Telecomunicações, 6201-001 Covilhã, Portugal
[3]Universidade da Beira Interior, Departamento de Informática, Instituto de Telecomunicações e ALLab Assisted Living
Computing and Telecommunications Laboratory, 6201-001 Covilhã, Portugal
[4]Universidade da Beira Interior, Departamento de Informática, 6201-001 Covilhã, Portugal

Corresponding Author:
Diandre de Paula
diandre.paula.cavalini@ubi.pt

Received: 26 November 2019
Accepted: 13 May 2020
Published: 2 June 2020

Publishing services provided by Knowledge E

Selection and Peer-review under the responsibility of the ICEUBI2019 Conference Committee.

Abstract

With the growth of ubiquitous computing, context-aware computing-based applications are increasingly emerging, and these applications demonstrate the impact that context has on the adaptation process. From the context, it will be possible to adapt the application according to the requirements and needs of its users. Therefore, the quality of the context information must be guaranteed so that the application does not have an incorrect or unexpected adaptation process. But like any given data, there is the possibility of inaccuracy and/or uncertainty and so Quality of Context (QoC) plays a key role in ensuring the quality of context information and optimizing the adaptation process. To guarantee the Quality of Context it is necessary to study a quality model to be created, which will have the important function of evaluating the context information. Thus, it is necessary to ensure that the parameters and quality indicators to be used and evaluated are the most appropriate for a given type of application. This paper aims to study a context quality model for the UbiPri middleware, defining its quality indicators to ensure its proper functioning in the process of adaptation in granting access to ubiquitous environments.

Keywords: QoC, Model, Context-Aware, Data, Privacy

1. Introduction

The ubiquitous computing is increasingly present in our day-to-day, mainly due to the advancement of mobile devices (characterized by a certain processing capacity, wireless communication and data storage) and a smart environment. One of the main factors in a ubiquitous environment is context-aware computing. This area is still on the rise, which has as its goal gathering environment information and user current situation, as well as consider the software and hardware characteristics of the device and communication [1]. This information collected is called context. Although many definitions for context have emerged, the most widely used is formalized in [2] as any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application itself. This definition does not limit the role that the context can play, allowing it to be applicable in the most diverse needs of each application. Also, no source constraint provides the context, so it is possible to receive the most diverse data about the entity. However, as it is a recent area, a lot of work needs to be done to fill certain gaps. As presented in [1], research on sensing, modeling, quality and safety needs to be further, which can be noted that there is a relationship between them. For instance, if an unstable source font is chosen as a provider context, it may compromise the quality of context, may receive information that is out of date or does not match the real world. It is important to emphasize that depending on the current context of an entity, the system will undergo a process of adaptation, and for that the context must be as close as possible to reality, ensuring that this adaptation meets the needs of the entity that is relating to the system. Thus, Quality of Context (QoC) plays a major role and can be used, for instance, to select sensors and/or context providers that achieve the defined minimum quality, improving context-aware reasoning and decision making, and mainly reducing the probability of incorrect context-aware adaptation process [3], so the QoC is any information that describes the quality of the information that is used as contextual information. QoC refers to information and not to the process nor the hardware component that possibly provides the information [4]. Therefore, well defining the system quality parameters and metrics is also a process that cannot be overlooked, as each system is different and may vary in the needs of the parameters and metrics, for example a dating system will not need room temperature information, on the other hand, for an Ambient Assisted Living (AAL) system the temperature may be important to know the state of the person being cared for [5-6].

In [7], UbiPri Middleware aims to control and manage privacy in ubiquitous environments. Access levels are assigned to users according to certain factors, such as the type of environment (blocked, private, or public) in which the user is, the day of the week, the period of the day (morning, afternoon or night), working day or not. There are five access levels defined in UbiPri: locked, guest, basic, advanced, and admin. Being blocked the lowest level of the hierarchy, preventing access in the environment, and administrative the highest level, allowing full access to all resources of the environment, and the control and management of the environment. An example of UbiPri's functioning could be its users entering an auditorium for a seminar, an environment whose noise should be avoided, and there may be limitations in the use of mobile data. In this way, the middleware will ensure a low level of access for the devices to be silted, and there may be limitations on the use of mobile data and/or application initiation.

Each module acts independently and with its functionalities, the most relevant to consider in the QoC model will be briefly described below:

• DataBase: module responsible for storing all data and definitions of users, devices and communications in the ubiquitous environment;

• Controller Module: module responsible for receiving access requests and controlling and managing the database directly in the tables;

• Data Module: module responsible for performing the processing of all variables and parameters received from the other modules.

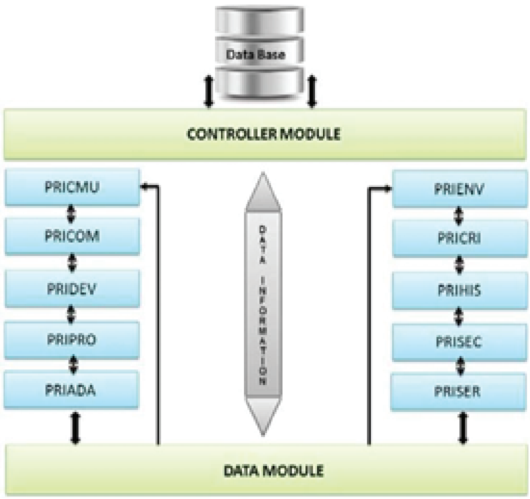

Figure 1: UbiPri privacy modules. (Source: Leithardt et al. [8].)

The UbiPri middleware does not present any module that addresses the quality of context. Since the UbiPri will adapt according to the context, that is, the current situation of the entity (environment, user, etc.), it becomes necessary to give due attention to the quality of context, so that the process of adaptation is not erroneous, for this should be left well clear and defined the parameters and metrics that will be used for the evaluation of the context quality. Therefore, the purpose of this article is to define the parameters and metrics that will be used to evaluate the context quality of the UbiPri middleware in the definition of the environment type, which is of paramount importance in assigning the level of access to the user.

This document is organized as follows: State of the Art discusses related works and the technologies used; Preliminary Aspects of the Implementation describes the preliminary aspects of the implementation process, such as definitions to be made and challenges that should be addressed; Modeling and Measuring Quality of Context in UbiPri Middleware describes the initial architecture of the proposed model. QoC

Quantification describes the indicators of context quality to be used to evaluate context information. Finally, the conclusion discusses the main contributions of this article and the work to be developed in the future.

1.1. State of the Art

Ubiquitous computation is constantly growing and is increasingly aimed at merging the information made up of computers with the physical space. With this, several types of researches focus on the context-aware computation [7, novo], which needs to receive information about the environment so that it can adapt properly to the environment, but few focus on its quality, using directly the data of the environment collected by the sensors which may eventually cause an erroneous adaptation and/or unexpected behavior of the system to the environment. In this way, it is necessary to take due care and treat these data received by the sensors to avoid any ambiguity and/or conflict of information [4], outdated information, wrong information or any type of condition that leads the system to take wrong decisions [9]. The Quality of Context can be used to select the appropriate sensors and/or context providers that contain the minimum level of quality fixed by the global QoC limits, improve decision-making and reasoning, and reduce the likelihood of error in the adaptation process [4].

In the literature, it is possible to find several types of QoC indicators considered [3, 4, 10-12], as a comparison is presented in Table 1.

It should be emphasized that in [3] it is described that to measure each QoC indicator, one or more QoC parameter is required to evaluate it, detailing the relationship between QoC parameters (QoCP) and QoC indicators (QoCI) modeling the quantification of parameters proposal. In [10] the same relationship is defined, but with different terminologies to the work presented in [3], between QoC parameters and QoC sources. It is possible to see in Table 1 that there is no usage pattern in the set of quality parameters, as this will depend on the type of context-aware computing implemented.

Precision, Probability of Correctness, *Resolution*, *Up-to-dateness,* and *Completeness* are parameters that often appear, which represent [3-4]:

- *Precision*: describes how exactly the provided context information mirrors the reality;
- *Probability of Correctness*: denotes the probability that a piece of context information is correct;
- *Resolution*: denotes the granularity of information;

Table 1: Quality parameters considered.

	[3]	[4]	[10]	[11]	[12]
Completeness	X		X	X	
Significance			X		
Trust-worthiness		X	X		
Up-to-dateness	X	X	X		
Resolution	X	X			X
Precision	X	X			X
Access Security	X			X	
Sensitiveness	X				
Probability of Correctness		X			X
Accuracy				X	
Freshness					X
Representation Consistency				X	
Delay Time					
Probability of Consistency					

- *Up-to-dateness:* describes the age of context information;
- *Completeness*: the degree of disponibility with which the context information is provided.

It is noteworthy that in addition to the quality parameters considered, the sensors and context providers influence the quality of context. As presented in [13] there are several types of sensors inserted in a ubiquitous environment, one of which can provide more accurate/updated/correct information than others, in that way the system can receive different information about the same entity, hindering the process of reasoning and adaptation of the system. Regarding the user's privacy policy, 2 works addresses this issue differently. In [3] it is considered that the user's privacy policy influences QoC, on the other hand in [12] it is proposed to use QoC for the protection of user's privacy through the QoC indicators. Remarkably, there is a relationship between QoC and user privacy, but the type of context-aware computing must be taken into consideration to implement the best approach to this issue. This lack of standardization of QoC parameters and the different ways of quantifying the parameters may reflect on the models of representation, making it difficult to understand and share context information and QoC [14]. Defining well the model and type of representation of the QoC is important to enable the correct evaluation of the QoC.

For the representation of the context quality, works were found that use graphic notation [15]; ontology [3, 12, 16]; *Extensible Markup Language* (XML) [10]; *Unified Modelling Language* (UML) [17]. Ontology shows the most popular choice for the representation of context knowledge and quality of context. Based on the studies that use ontology for the representation of QoC, several advantages over the use of ontology are presented, among them:

• Ontologies can be used by inference engines to infer complex contexts from lower-level contexts acquired by diverse sources [18];

• Respects all requirements for ubiquitous computing environments [18];

• Allows computational entities and services to have a common set of concepts and vocabularies to represent knowledge about a domain of interest [19];

• It allows the definition of a specific domain, it is possible to work in a certain area optimizing the process of information extraction and the exchange of knowledge [20].

However, as seen in [21] the use of ontology also presents disadvantages, such as an immature manipulation technology and an impact on the application's performance.

In [22] the main formal languages to construct ontology are described:

- *Resource Description Framework* (RDF): its basis consists of the use of object-attribute-value triples, which denote relationships between the parts of objects;
- RDF-*Schema*: extends the RDF by adding class and hierarchy concepts, allowing the creation of simple ontologies;
- *Ontology Web Language* (OWL): based on XML and RDFS syntax. It provides additional instruments to clarify the meaning of terms and relationships.

In [15] a model based on graphic notation is proposed, which provides a formal basis for representing and reasoning about some of the properties of the context information, such as its persistence and other temporal characteristics, its quality and its interdependencies.

In [10] is proposed an XML-based model, whose main concepts are *SupportWorkers* that perform disaster-site rescue work, *Devices* that are used by *SupportWorkers* for communication and collection of context information, *Activities* of the *SupportWorkers* and the disaster response Site.

In [17] a UML-based model is proposed, which defines an *Entity* identified by an identity attribute that can be associated with a certain *Context* at a certain point in time.

In [3] a semantic approach is made through an OWL-DL-based QoC model, an OWL sublanguage that provides additional constructions that allow more expressiveness

[22], which is built around two main classes, already mentioned previously in this same section, QoCP and QoCI.

In [16] the Proteus model, based on RDF, deals with QoC at two levels: context and policies. Proteus associates each context with a quality attribute, allowing them to disregard context data that does not reach the minimum quality. The selection of applicable policies is made based on the current context and its quality. This helps minimize the risk of granting access to resources based on incorrect or ambiguous context information.

In [14] is presented a model of knowledge of context quality for ubiquitous environments based on ontology, in his work the development of the model was divided into five cycles.

- Cycle 0 – Purpose and scope: to identify the purpose and scope of ontology, whose identification will represent the starting point for the development of the proposed knowledge model;
- Cycle 1 – Survey of competency issues and definition of terms: elaborate pertinent issues that the proposed model must meet, also define ontology terms, tasks such as defining classes, defining the properties of classes and defining the relationships between classes were performed in this cycle;
- Cycle 2 – Refinement of the terms: perform a technical verification of ontology before the domain, to seek possible inconsistencies about the domain in the sources of knowledge. And from this review, refine the terms defined in Cycle 1;
- Cycle 3 – Prototyping: performing the prototyping of ontology. Tasks such as instantiating instances, valuing instance properties, and valuing the relationship between instances were performed in this cycle;
- Cycle 4 – Model verification: perform a technical check of the ontology before the reference framework. In this cycle will be revisited the purpose, scope and questions of competence of the ontology, to evaluate the consistency of the ontology against the requirements raised.

In the study carried out in [14] some tasks were specified that were performed in each cycle, for example, the consideration of the reuse of ontologies, but QoC is an area that is on the rise and will hardly be found ontologies that can be reused, therefore the distribution of tasks may not be used for all. On the other hand, it is hardly found in the literature how to develop a model of knowledge of context quality and therefore the five cycles can serve as a basis for assisting new work in the area, and each work may require the realization of other tasks or not. However, the development of the context quality knowledge model to be used is of paramount importance for the correct functioning of the QoC assessment. Regarding the practical use of QoC, studies were found in the literature as [23-25], mostly related to the health area, which is understandable because it is an area whose information quality is of paramount importance since an action taken erroneously by the system can cause devastating consequences.

In [23], a QoC approach was made to improve the monitoring functions of Ambient Assisted Living (AAL) applications, according to [5] AAL is the term given to the provision of care to people in their own homes, supported by technology. For the tests performed in the open-source Siafu context simulator, the following QoC parameters will be considered: accuracy, up-to-dateness, trustworthiness and completeness. To quantify the QoC, an arithmetic mean of the parameters is made and from the value obtained from the QoC quantification it will be possible to evaluate whether the information obtained is adequate, otherwise, i.e. if the QoC value is not adequate, it is expected that the set of used parameters allows you to perform an analysis in order to identify the cause of the problem. In the tests performed in the Siafu, 3 scenarios were configured with variable values for the parameters of the sensor to find out which scenario would provide the highest quality of data. After 2 hours of testing in each scenario, and every 30 seconds values of context parameters and QoC were calculated, it was possible to verify the relationship between the QoCP accuracy and completeness since the information does not exist, is incomplete; soon there is no precision whatsoever. For the tests performed in real-world scenarios, was used data from users collected through the e-Health Sensor Platform. The tests were used sensors for pulse and oxygen in the blood, body temperature, blood pressure, body position, and fall, it is that the biometric and medical applications that the e-Health Sensor Shield V 2.0 allows to run are used 10 sensors to carry out the monitoring of the body. Sensor data were collected through the open-source Arduino Software (IDE): ARDUINO 1.0.6 and the environment was written in Java. The proposed QoC parameters were *up-to-dateness* (U), *completeness* (Cm), *coverage* (C), *precision* (P) e *significance* (S). The quantification of QoC was made in the same way as in the simulator (through the arithmetic mean) and took into account U, C, P and Cm, and S is additional information serving only to alert certain situations when necessary. Finally, it was possible to verify that the QoC assessment allows detecting anomalies or inconsistencies in the sensors, activating sensor backups, dropping data with insufficient QoC, choosing appropriate context providers, and generating alerts including potential health problems and emergencies. It would be interesting to present a QoC quantification utilizing a weighted average, differentiating the weights for each

QoCP according to the scenarios, also considering the remaining data of the sensors that were not considered.

In [24], a model is proposed that uses QoC indicators to improve access control, grant/deny access to resources, in pervasive environments. It is interesting the approach made in its work, because it was proposed a method that would measure QoCI at any level of semantic representation and from the proposed components and equations, an algorithm will perform the measurement of the QoC indicator for each tuple (CI, QoI), where CI corresponds to the context information and QoCI the relevant QoC indicator for the application/user, its result is a real number in the range [0, 1], in which 0 corresponds to a minimum degree of QoCI quality and 1 the maximum degree. The QoC degree (QoCD) is also defined, which is the degree of global quality of relevant information. The QoCD is measured to reflect the QoC requirements of the application/user taking into account the weight of the tuples (CI, QoCI) relevant to the adaptation process in the context-aware service. Thus, its quantification is done through a weighted average of the tuples. The case study was done using the PhotoMap application. This is a location-based mobile and Web application for photographic annotation. The model aims to achieve 3 main objectives: i) offer a mobile application that allows users to take photos and group them into space-temporal collections; II) propose a Web system that organizes the user's photos using the context information acquired; III) improve the recall of the user of their photos showing spatial, temporal and social information inferred. According to the case study, it was possible to notice that the relevance of tuples (CI, QoCI) changes according to the situation and perception of the user. It would be interesting to present performance results of the execution, in this way it would be possible to carry out an approach considering the characteristics of the devices, could also be mentioned as the client-server communication PhotoMap is performed, again being able to take into account the characteristics of the device for considerations in communication.

2. Preliminary Aspects of the Implementation

After studying a Context Quality Model for Middleware UbiPri, and before implementing the solution for ontology-based context quality, it is important to formulate the taxonomy to be used. Taxonomy serves to classify rules and parameters, resulting in a better understanding of the functionality, also allows a greater degree of accuracy in the classification process [26]. Thus, understanding the classes, properties of objects, data properties and competency issues that will constitute the ontology to be used is an important step, as well as define other components of taxonomy such as communication, user device, criteria to consider that influence the final decision of UbiPri's adaptation process.

It is also important to identify the challenges and how they will be treated, such as the use of the Boolean decision or fuzzy logic, while the quantification of Context Quality can be separated into 3 classes, which were presented in the QoC quantification section. This problem can be solved with tests and evaluation, between the use of boolean logic and fuzzy logic and evaluating which results are most effective in quantifying and accurately context quality. Another challenge to be faced will be the criteria of different parameters, so the use of taxonomy becomes important to allow the domain to be robust and well defined.

2.1. Modeling and Measuring Quality of Context in UbiPri Middleware

Based on the study presented and based on the literature research, this article aims to present a study for an initial architecture and define the indicators of QoC and its evaluation methods to be applied in the UbiPri middleware. To prevent incorrect context information from compromising the middleware's operation, the information must be treated before it reaches the Controller Module, which will perform the processing of all variables and parameters received from other modules. In this way, the proposed model will be initially implemented in the Data Module, which would allow dealing with the raw information and access to the information stored in the database, without the need for a restructuring of the UbiPri middleware. To demonstrate the need for QoC, a practical example that could lead to poor evaluation by the middleware and grant a certain level of access erroneously, it would be the user to be in a certain location, which there can be no noise disturbances, however if the context information is outdated and shows that the user is in a public place, the device would be able to have full access to the resources, i.e. it could disrupt a meeting. It is also necessary to deal with the inconsistency so that if two different context information about the same entity is collected from different sensors, it is considered the most reliable information. Based on the UbiPri middleware objective, the following context quality indicators will be considered: i) up-to-dateness and II) trustworthiness, which will initially be enough to cope with these QoC problems. A third indicator will be proposed to use QoC to select security mechanisms with a minimum level of quality so that it does not exceed the limits of the user's devices.

2.2. QoC Quantification

The context quality indicators to be considered by the UbiPri middleware will be presented to deal with outdated data, inconsistencies and the security to be considered for the realization of the communication. The values obtained will be found in the range [0..1], and 1 corresponds that the information has maximum compliance with certain requirements and 0 the minimum value. However, it is possible to separate the range in 3 classes: low, medium and high [3]. Being low corresponding to the interval [0, 0.33]; medium to the interval [0.34, 0.66] and high to the interval [0.67, 1].

According to UbiPri's needs, initially, the parameters chosen are enough to deal with QoC and eliminate possible ambiguities that may affect the UbiPri middleware adaptation process. Your choices will be justified in each subsection of the parameters to be used. However, no future tests will be discarded to verify the efficacy and efficiency of QoCI.

In this way, the general plans to test the solution will consist of generating two types of ambiguous data:

- with a difference in time;
- same time.

And from the tests, it will be possible to check how effective are the QoCIs chosen, and if they need complementary QoCI to deal with unpredictable situations, in which elected QoCIs cannot deal with, and thus check strengths and weaknesses in the QoCI used and whether there are complementary QoCI needs or even use other QoCIs.

2.2.1. Up-to-dateness

One of the UbiPri middleware's adaptations to access levels is taking into account the type of environment, verifying the validity of information is essential in the adaptation process. One way to verify the validity of information is through its age. Thus, this factor can be decisive for the behaviour and adaptation of the UbiPri middleware. For example, if an employee who was in a public environment during lunch (having access to all resources) returns to work and joins a meeting, the middleware must ensure a low level of access for devices to be muted, and there may even be limitations on mobile data usage and/or application initiation. But suppose the sensors have not detected the change of environment (a physical failure, for example), measuring the age of the information will be essential to decide whether the information is valid or not. Thus, the upto-dateness parameter was one of the chosen ones to be applied in the UbiPri QoC evaluation because describes the age of context information. It should be considered the age of the context information (CI) and the lifetime that corresponds to the time limit until the context information becomes obsolete. To get the age of the context information (CI), it will be necessary to calculate the difference between the time when the information was measured and the current time, as shown in (1):

$$\text{age(CI)} = \text{time}_{\text{current}} - \text{time}_{\text{measured}}\text{(CI)} \tag{1}$$

If the age of context information CI is less than the lifetime, equation (2) shows how to obtain the value of the up-to-dateness indicator relative to CI, U(CI):

$$\text{U(CI)} = 1 - \frac{\text{age(CI)}}{\text{lifetime(CI)}} \tag{2}$$

Otherwise, the value of U(CI) will be 0, that is, the information should not be used because it is not possible to be sure about the validity of this information. The higher the age of the context information CI, the lower its validity. Thus, information that is near or within the range considered low, can be ignored or measured again. For static information, such as user profiles, it is possible to define the lifetime of this information with a maximum global value, thus the CI age would not affect the value of QoCI up-to-dateness [10]. This indicator is useful for verifying the validity of the context information, so that there is no outdated data stored, but also that it is not necessary to always update its value due to the definition of its lifetime.

2.2.2. Trustworthiness

The QoCI trustworthiness will deal with the ambiguity of new data, i.e. different values received at the same time by different sensors. For example, let's imagine that a student is entering a classroom and two types of environment is captured by UbiPri, the up-to-dateness parameter could not cope with this ambiguity since by age the data would be considered valid, to deal with such ambiguity the best solution would be to consider the distance of the sensors in relation to the entity. The QoC trustworthiness indicator describes the probability that the information provided is correct. The trustworthiness of context information is highly affected by the space, that is, by the distance between the sensor and the entity. The farther the sensor is from the entity, the more doubtful it will be the accuracy of the context information provided by this sensor. For this, it is necessary to have defined the maximum distance in which we can rely on information provided, but only the distance will not be sufficient to define the reliability of the information, it will be necessary to consider the accuracy of the sensor, i.e. the probability of the context information being correct.

If the distance between the sensor and the entity (d (S, E), where S corresponds to the Sensor and E the entity) is less than the maximum distance for that sensor type, then the QoCI trustworthiness, T (CI), will be obtained through equation (3) [10]:

$$T(CI) = \left(1 - \frac{d(S,E)}{d_{max}(S)}\right) \times \alpha \quad (3)$$

The accuracy of the sensor, α, is defined based on the estimate made in [11]. The maximum distance for the sensor cannot be the same for all sensors, for example, d_{max}(Satelite) cannot be the same as d_{max}(RFID), even if both provide location information. The QoCI trustworthiness is useful for dealing with inconsistency issues, that is when different sensors provide different information about the same entity. In the practical example presented earlier in this same section, it would be considered the sensor that has a high-reliability value, that is, the sensor is closest to the entity.

2.2.3. Security

One of the main objectives of UbiPri middleware and maintaining user privacy, and this includes communication. However, there may be situations where the user's device is limited (for example companies that provide devices to employees) or situations where demand for communication security is high, but the current level of security is low. To do this, we must calculate the level of security of the communication to be carried out. Aiming at this objective, we define security as the probability that the current security of communication is adequate, and can be adjusted according to the characteristics of the device used by the user. The user hierarchy level and the current security level that is being used must be considered. If the current security level ($sec_{current}$) is less than the hierarchical level of the user (level (user)), the communication QoCI Security, S(Com), can be obtained through equation (4):

$$S(Com) = \frac{sec_{current}}{level(user)} \quad (4)$$

For example, if the user is of a maximum hierarchical level, but your device does not support high-level security, you can consider another approach that does not require so many device resources and that it is within the range considered safe for this hierarchical level. This QoCI will be useful when the user's device resources are limited, allowing it to select a communication security mechanism that is within the limits of the device.

3. Conclusions

Although the quality of context did not have an in-depth study, it was possible to notice the importance of addressing the quality received from sensors and other context providers to avoid the incorrect behavior of the system. Many problems may arise due to the lack of quality of context information depending on the application for application, but it is possible to say that inconsistency and outdated values of the context data are the main problems due to this lack of attention in quality that leads the system to a process of erroneous and unexpected adaptation. Despite this, few context-aware systems demonstrate this concern in addressing the quality of context.

This article has as its contribution the proposal to implement QoC in the UbiPri middleware since its previous versions do not have mechanisms that deal with this, proposing a new indicator of context quality that aims to minimize the cost in user's device and maintain the quality provided by the UbiPri middleware, also propose QoC indicators and their measurement methods, so that it can evaluate the validity (up-to-dateness) and reliability (trustworthiness) of the context information that the UbiPri middleware possesses, allowing for proper functioning in the adaptation process. To do this, the QoC model will be implemented in the Data Module, so that the middleware can address the context quality of its information without the need for its restructuring. For future work, we plan to implement the representation of the knowledge of the context quality, to deepen the evaluation of the proposed QoC indicators verifying its efficacy, to carry out an evaluation for the other QoCI and whether they have a significant influence to UbiPri and verify the dependency between the context quality indicators. Also, propose the implementation of a module dedicated to the quality of context.

References

[1] Loureiro, A.; Oliveira, R.; Silva, T.; Júnior, W.; Oliveira, L.; Moreira, R.; Siqueira, R.; Rocha, B.; Ruiz, L.: "Computação Ubíqua Ciente de Contexto: Desafios e Tendências" 27° Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuído, Recife, Brasil, Mai 2009.

[2] Dey, A.: "Providing Architectural Support for Building Context-Aware Applications", Georgia Institute of Technology, 2000.

[3] Filho, J.; Miron, A.; Satoh, I.; Gensel, J.; Martin, H.: "Modeling and Measuring Quality of Context Information in Pervasive Environments" 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, Austrália, Abr 2010,

p. 690-697.

[4] Buchholz, T.; Schiffers, M. Quality of Context: What It Is And Why We Need It. Proceedings of the 10th Workshop of the OpenView University Association: OVUA'03, 2003, p. 1-14.

[5] Mcnaull, J.; Augusto, J.; Mulvenna, M.; Mccullagh, P.: "Data and Information Quality Issues in Ambient Assisted Living Systems" ACM Journal of Data and Information Quality, Vol. 4 nº 1 (2012), p. 1-15.

[6] Zdravevski, E.; Lameski, P.; Trajkovik, V.; Kulakov, A.; Chorbev, I.; Goleva, R.; Pombo, N.; Garcia, N. Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering. IEEE Access 5 (2017), p. 5262–5280.

[7] Leithardt, V. UbiPri: middleware para controle e gerenciamento de privacidade em ambientes ubíquos. Dissertação Ph.D. Universidade Federal do Rio Grande do Sul, Instituto de Informática, 2015. https://hdl.handle.net/10183/147774 (22/08/2019).

[8] Leithardt, V.; Borges, G.; Rossetto, A.; Rolim, C.; Geyer, C.; Correia, L.; Nunes, D.; Sá Silva, J.: "A Privacy Taxonomy for the Management of Ubiquitous Environments" Journal of Communication and Computer 10, 2013, p. 1529–1553.

[9] Manzoor, A.; Truong, H.; Dustdar, S.: "Using Quality of Context to Resolve Conflicts in Context-Aware Systems", QuaCon 2009, Springer, 2009.

[10] Manzoor, A.; Truong, H.; Dustdar, S.: "On the Evaluation of Quality of Context". 3^{rd} European Conference on Smart Sensing and Context, 2008, p. 140-153.

[11] Kim, Y.; Lee, K.: "A Quality Measurement Method of Context Information in Ubiquitous Environments". International Conference on Hybrid Information Technology, 2006, p. 576-581.

[12] Sheikh, K.; Wegdam, M.; van Sinderen, M.: "Quality-of-context and its use for protecting privacy in context aware systems". JSW, vol. 3, no. 3, p. 83–93, 2008.

[13] Beigl, M.; Krohn, A.; Zimmer, T.; Decker, C.: "Typical sensors needed in ubiquitous and pervasive computing". First International Workshop on Networked Sensing Systems, ser. Society of Instrument and Control Engineers, vol. 4, June 2004, p. 153–158.

[14] Nazário, D; Dantas, M.; Todesco, J.: "Modelo de Conhecimento de Qualidade de Contexto para Ambientes Ubíquos baseado em Ontologia". International Journal of Knowledge Engineering and Management - IJKEM, 2015, 4(9), p. 51–86.

[15] Henricksen, K.; Indulska, J. Modeling context information in pervasive computing systems". Pervasive Computing: Lecture Notes in Computer Science, vol. 2414, 2002, p. 79-117.

[16] Toninelli, A.; Corradi, A.: "A Quality of Context-Aware Approach to Access Control in Pervasive Environments", MobileWireless Middleware, Operating Systems, and Applications: LNCS, vol 7, p. 236-251, 2009.

[17] Neisse, R.; Wegdam, M.; Sinderen, M.V.: "Trustworthiness and Quality of Context Information". 9^{th} International Conference for Young Computer Scientists, 2008, p. 1925-1931.

[18] Strang, T.; Linnhoff-Popien, C.; Frank, K.: "Applications of a Context Ontology Language", in Workshop on Advanced Context Modelling Reasoning and Management, in 6^{th} International Conference on Ubiquitous Computing, 2004, p.1-5.

[19] Pawar, P.; van Halteren, A.; Sheikh, K.: "Enabling Context-Aware Computing for the Nomadic Mobile User: A Service Oriented and Quality Driven Approach". IEEE Wireless Communications and Networking Conference, 2007, p. 2529-2534.

[20] Lopes, L.: "Um modelo de engenharia do conhecimento baseado em ontologia e cálculo probabilístico para o apoio ao diagnostico", Universidade Federal de Santa Catarina, 2011.

[21] Santos, V.: "Gerenciamento de Contexto em Sistemas Colaborativos", Universidade Federal de Pernambuco, 2006.

[22] Heinzle, R.: "Um modelo de engenharia do conhecimento para sistemas de apoio a decisão com recursos para raciocínio abdutivo", Universidade Federal de Santa Catarina, 2011.

[23] Nazário, D.; Campos, P.; Inacio, E.; Dantas, M.: "Quality of Context Evaluating Approach in AAL Environment Using IoT Technology". In 2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS). 2017. 22-24 Junho. p. 558-563.

[24] Filho, J.; Martin, H.: "Using context quality indicators for improving context-based access control in pervasive environments". In EUC '08: Proceedings of the 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing. IEEE Computer Society (2008). p. 285–290.

[25] Valéria, O.; Ribeiro, A.; Leal, L.; Lemos, M.; Carvalho, C.; Filho, J.; Filho, R.; Agoulmine, N.: "Using QoC for Improving Energy-Efficient Context Management in U-Health System". In 2014 IEEE 16th International Conference on e-Health Networking, Applications and Services (Healthcom), Natal, Brazil, 15-18 Out 2014, p. 317-322.

[26] Silva, L.; Leithardt, V.; Rolim, C.; Villarrubia-González, G.; Geyer, C.; Silva, J. PRISER: managing notification in multiples devices with data privacy support. Sensors,19(14):3098,2019.