The 9th International Conference for Young Computer Scientists

Trustworthiness and Quality of Context Information

Ricardo Neisse*, Maarten Wegdam*,⋆, and Marten van Sinderen*
*CTIT, University of Twente, Enschede, The Netherlands
Email: {r.neisse, m.wegdam, m.j.vansinderen}@utwente.nl
⋆Telematica Instituut, P.O.Box 589, 7500 AN, Enschede, The Netherlands
Email: maarten.wegdam@telin.nl

Abstract

Context-aware service platforms use context information to customize their services to the current users' situation. Due to technical limitations in sensors and context reasoning algorithms, context information does not always represent accurately the reality, and Quality of Context (QoC) models have been proposed to quantify this inaccuracy. The problems we have identified with existing QoC models is that they do not follow a standard terminology and none of them clearly differentiate quality attributes related to instances of context information (e.g. accuracy and precision) from trustworthiness, which is a quality attribute related to the context information provider. In this paper we propose a QoC model and management architecture that supports the management of QoC trustworthiness and also contributes to the terminology alignment of existing QoC models. In our QoC model, trustworthiness is a measurement of the reliability of a context information provider to provide context information about a specific entity according to a certain quality level. This trustworthiness value is used in our QoC management architecture to support context-aware service providers in the selection of trustworthy context providers. As a proof of concept to demonstrate the feasibility of our work we show a prototype implementation of our QoC model and management architecture.

1 Introduction

Context-aware service platforms use context information to adapt their services to the current situation of the service users. An example of a context-aware service targeted to an office environment is a meeting notification service that tracks the location and activity of meeting attendees and detects, based on their context, if someone of them will be late or unavailable for a meeting. In order to maximize its usefulness, context-aware service providers should be provided with high quality context information.

If the context-aware meeting service, for example, receives very old or imprecise location information of a missing attendee, the service may be unable to determine or estimate correctly the delay or probability that the person will or will not be present at the meeting. Low quality context information may be provided due to technical limitations of sensors and reasoning algorithms or also due to privacy policies of the entities to whom the context information refers to (a.k.a. context owners, see Section 2) [15].

More important then to provide context-aware services with high quality context information is to quantify and indicate to the context-aware services the quality of the context they are receiving. This is more important because the services may adapt differently if they are aware of the quality of the context information, for example, the context-aware meeting service may indicate to the meeting attendees that it is impossible to be sure about the presence or absence of a missing attendee if the location information it receives is not precise enough. Quality of Context (QoC) quantification is also important for other reasons such as establishment of QoC level agreements between context-aware service providers and context providers, for efficient selection of context providers according to the service provider requirements, and for privacy protection by decreasing the quality level according to the context owners' preferences [15] [3].

Models for QoC quantification have already being proposed in the literature [15] [3] [6] [7] [9] and they quantify the quality characteristics of the context information using quality attributes (a.k.a. quality indicators), which are attached and communicate together with the context information instances as meta information. Examples of QoC attributes defined by existing QoC models are accuracy, precision, freshness, probability of correctness, trustworthiness, and temporal and spatial resolution. The problems we have identified with existing QoC models is that they do not follow a standard terminology for the QoC attributes and none of them clearly differentiate the quality attributes related to instances of context information (e.g. precision) from trust-

978-0-7695-3398-8/08 $25.00 © 2008 IEEE
DOI 10.1109/ICYCS.2008.368

worthiness, which is a quality attribute related to the context information provider.

In this paper we propose a QoC model that contributes to the alignment of the terminology of existing QoC models using as a reference an existing standard metrology vocabulary [4]. We found out that the concepts defined by this vocabulary fits perfectly to the concepts used by existing QoC models. Despite the terminology alignment, the most important contribution of our QoC model is the explicit differentiation of trustworthiness as a quality attribute of the provider of the context information and not of the context information instances. In our model, trustworthiness is a measurement stating the degree of reliability of a context information provider to provide context information about a specific entity according to a certain QoC level. We have applied our model in a QoC management architecture that manages the QoC attributes and trustworthiness of context providers in order to help context-aware service providers in the selection of trustworthy context providers. We proof the feasibility of our work through a prototype implementation that realizes our QoC model and management architecture.

This paper is organized as follows. Section 2 presents our context-aware service platform [16] and explains the role of QoC in this service scenario. In Section 3 we detail our QoC model and quality attributes based on a standard vocabulary and in Section 4 we show how trustworthiness relates to our QoC model as a quality attribute of the context information provider. Section 5 presents our QoC model applied in a QoC management architecture and our proof of concept prototype. In Section 6 we discuss the related work on QoC modeling and the limitations of existing approaches. Section 7 finishes this paper with conclusions and future work.

2 Context-Aware Service Platform

We base our QoC model on our target context-aware service platform [16], which is illustrated in Figure 1. We distinguish in our service platform five roles, namely users, context owners, identity providers, context providers, and service providers. In Figure 1 first the user authenticates with an identity provider (1) and receives an identity token. The identity token is used to access a service provider (2), which verifies the user's identity (3) and retrieves context information to adapt the service (4). The context information can be, for instance, the current activity or location of the user; however, it can also include context about other entities (context owners) that are relevant for the service. The context provider produces context information from raw sensor data acquired from sensors in the context-owner's environment (5).

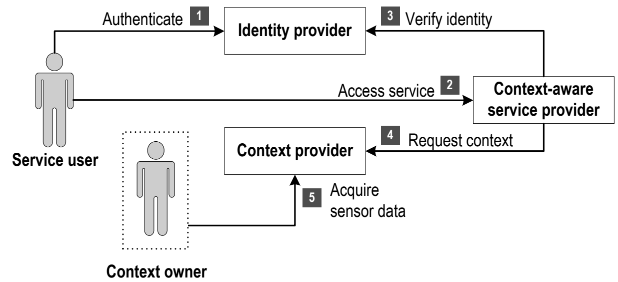

Figure 1. Context-aware service platform

Figure 2 shows in details how the context-aware service provider interacts with a context provider including the discovery process. The context provider acquires raw data from sensors in the environment and produces context information at a certain quality level about an specific set of entities[1]. The context type, supported quality level, and the entities that the context provider can produce context about (a.k.a. context owners) are advertised in a context broker. When the context-aware service provider needs context about a certain entity it queries the context broker and may inform a minimum quality level in the discovery request. The context broker returns to the context-aware service provider a list of context providers together with the supported QoC level and context owners.

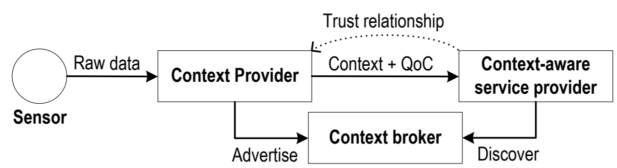

Figure 2. Context discovery and provisioning

After the discovery of the context provider (Figure 2) the context-aware service provider requests context about an specific context owner and receives the context information together with the QoC attributes related to the context information instance. The QoC attributes advertised by the context providers may not correspond to the QoC level advertised by the context provider, because the QoC level may change dynamically according to the environment conditions. For this reason, the QoC attributes always communicate together with the respective instances of the context information.

In the QoC literature, trustworthiness is usually included as a quality attribute of the context information, however, trustworthiness is a property of the context information provider from the context-aware service provider point of view [5] and not of the context information itself. Furthermore, trustworthiness is not related with the quality level

[1]Our context-aware service platform also supports context providers that do not interact directly with sensors

of the information, but with the capability of the context provider on correctly describing the QoC attributes about himself.

3 Quality of Context Attributes

In our QoC research we adopt as a reference the quality concepts and vocabulary proposed by the International Organization for Standardization (ISO) [4] and compare these standardized concepts with the QoC terms and concepts from existing QoC models [3] [9] [6] [15]. This ISO standard is well accepted and used as a reference by engineers in metrology and measurement, and we found that there is a match between the concepts defined by this standard and the concepts of QoC defined by existing QoC models. In this Section we analyze the quality attributes from the ISO standard (accuracy, precision, and resolution) and related them to the concepts from existing QoC attributes (accuracy, precision, probability, spatial resolution, temporal resolution, and freshness) in order to define our own QoC reference model. Due to its nature, trustworthiness is discussed separately in Section 4.

The ISO standard defines accuracy as how close a measurement is to the real or to an accepted reference value, while precision is defined as how close together or how repeatable are the results from a measurement. Furthermore, according to the ISO definition [4] "*accuracy can not be expressed as a numeric value*", only the inaccuracy, which can be measured as the error or percentage error. For this reason we do not consider accuracy and inaccuracy in our QoC model as relevant concepts because it is impossible for the context provider to determine for every context information request the real known value of the context information, so the accuracy/inaccuracy values can never be determined.

For illustration purposes lets assume that the ambient temperature of a room is known to be 25 degrees Celsius and a sensor in the room indicates 26 degrees, the error is then 1 degree Celsius. The inaccuracy information in this case is useful only for calibration of context providers and for precision calculation when the real context values are known by repetitively comparing the context provider readings with the real know values. Using a graphical analogy the readings of a context provider can be related to shoots in a target, where the center of the target (the bull's eye) is the true value of the context. Figure 3 represents the pattern of shoots and how this would be interpreted as accuracy and precision of the context provider.

For numeric context information, precision can be expressed by means of significant figures, and in this case the average error is understood when not explicitly stated as to be one-half the value of the last significant digit. For example, the ambient temperature measurement of a room that records 25 degrees implies a margin of error of $\pm$ 0.5 degrees or 2% (0.5 divided by 25). In this case the implicit precision is directly determined from the value of the context information and may be used in case there is no explicit precision information defined. The standard measurements of percentage difference and variance can also be used to measure the precision of a context provider.

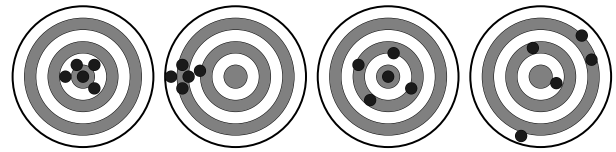

Figure 3. Target accuracy and precision

It is common for a context provider to reply with boolean or discrete sets of values instead of numeric values, for instance, to determine if an entity is in a room or not. In case the context provider replies with boolean values the precision can be measured as the percentage proportion of true positives and true negatives results in relation to the total number of results including the false positives and false negatives. For a presence sensor, for example, a precision of 100% indicates that the context provider identifies the presence/absence of an entity in a room always correctly. Our conclusion is that the ISO concept of precision overlaps with the concepts of precision and probability defined by [15] and [3], therefore we believe that the concept of probability should be considered as a measurement of precision and do not need a different name in our model.

Temporal resolution has been defined by [15] as *the period of time to which a single instance of context information is applicable or the best possible approximation of time at which a context was determined*. In our QoC model we consider temporal resolution an intrinsic quality attribute of the timestamp associated with the context information, which can be measured by the number of significant time units available (e.g. year, month, day, hour, minute, etc.) or by a time period (e.g. 2008/01/01 from 10PM until 12PM). Freshness [15] (a.k.a. up-to-dateness [3]) is also related with the timestamp and measures the age of the context information instance.

Spatial resolution has been defined by [15] as *the precision with which the physical area, to which an instance of context information is applicable, is expressed*. From the examples of spatial resolution presented by [15] we conclude that spatial resolution is always associated with the resolution of the location of a physical entity. Therefore, we also do not include spatial resolution in our QoC model and we adopt a more generic approach where temporal resolution is simply the precision of the location information.

Our QoC and context model (Figure 4) defines an *Entity*, identified by and *identity* attribute, that may be associated with a certain *Context* at a certain moment in time (*Timestamp*). Both *Context* and *Timestamp* have *Quality Attributes* associated with them. In our model the explicitly defined quality attribute is precision, which covers all the definitions from the literature of precision, probability, and spatial resolution. Furthermore, the resolution of the context and timestamp, and the freshness of the timestamp can be calculated from the context information and timestamp, so this attributes are not explicitly included in our model but are defined as methods of our classes. The precision of the timestamp is not addressed by our model because it concerns issues related with clock synchronization and is out of the scope of this paper. Our QoC model is generic, extensible, and supports by default QoC attributes related with the timestamp and numeric context types (more details are presented in Section 5).

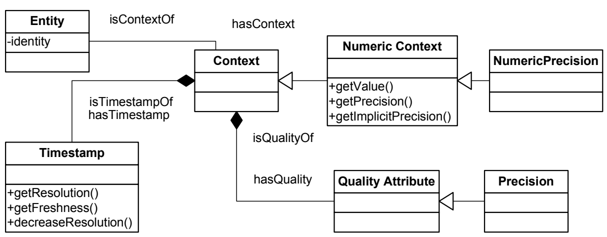

Figure 4. QoC UML model

4 Trustworthiness of Context Providers

The concept of trustworthiness of context information relates with the concept of trust, which has been widely studied in the social [12][11] the informational [5] [1], and the technical research areas[10]. The difference between each of these research areas is the nature of the actors involved in the trust formation process (human or machine), and the class of problems they address. We base our QoC trustworthiness definition on the social and informational trust research because we are interested specifically on the trust beliefs one entity in the context-aware service platform has in another one regarding the provisioning of context information.

In its more general definition, trustworthiness is represented as a *subjective measurement of belief* from one entity regarding the *behavior* of another entity focused on a certain trust *aspect*. The entities part of a trust relationship are widely referred in the trust management literature as Trustor and Trustee. Consider, for example, that Bob (Trustor) trusts Alice (Trustee) at a high degree for what concerns her competence in providing location information about herself with high precision. Trustworthiness in our QoC model is therefore not a quality attribute of the context information instance but a *degree of belief* from the context-aware service provider point of view regarding the context provider.

In our model (Figure 5) trustworthiness is defined as a degree of belief a trustor has in a trustee regarding a certain behavior and aspect. Trustors can perceive or interpret the Trustee's behavior as an isolated or combined measurement of honesty, competency, reputation, usability, credibility and reliability. In this paper we consider trust behavior as "honesty, competence, and reliability". The competence, honesty, and reliability in the specific case of context information provisioning is with regard to the trust aspect of providing context information about a certain context owner, and according to an advertised QoC level. Other trust behaviors are also important and will be considered in our future work, for a list of possible trustee behaviors please consult [14].

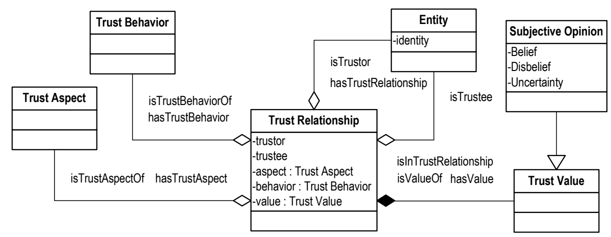

Figure 5. Trust basic UML model

For the quantification of the trust beliefs (a.k.a. trustworthiness [2]) there are different approaches [5], for example, the degree of belief can be measured using a numeric value from 0 to 1 [2]. Based on our previous work on trust for context-aware service platforms [13] we choose to quantify the trustworthiness of context providers using a Subjective Logic (SL) opinion [8]. In a SL opinion the trust belief is not only a measurement of the degree of belief but a composition of belief (b), disbelief (d), and uncertainty (u). We have mapped the subjective opinion triple (b, d, u) to an ordered set {very untrustworthy, untrustworthy, trustworthy, very trustworthy} whose elements model judgment of user perspectives [13]. An opinion *op* whose belief is higher than disbelief is considered trustworthy if it has uncertainty not lower than 1/3 and very trustworthy otherwise. One opinion *op* whose belief is not higher than disbelief, is considered untrustworthy if it has uncertainty not lower than 1/3 and very trustworthy otherwise.

The bootstrapping of the trustworthiness values in our QoC model is done through pre-defined trustworthiness values or based on recommendations received from trusted 3^{rd} parties. Pre-defined trustworthiness values are usually defined based on the dispositional trust [12] that is the likeli-

[2] we use the terms trust and trustworthiness interchangeably meaning the measurement of the degree of belief

hood to trust other entities in the absence of concrete trust evidence. A simple strategy in this case would be to consider a context provider trustworthy by default, in case no recommendations are received. If multiple recommendations are received they are combined using a ”fair” combination, which is supported by the consensus operator from the SL [8].

After the bootstrap the trustworthiness values evolve based on the feedback about the perception of users of the context-aware service regarding the reliability of its adaptation. When the users of the context-aware service notice wrong or inappropriate service adaptation they can provide *negative* or positive feedbacks [7]. The positive feedback is mapped to a *trustworthy* opinion and a negative feedback to an *untrustworthy* opinion. In case a positive feedback is received the current trustworthiness value of the context provider for the specific context type, QoC level, and user identity is increased, and for the negative feedback the trustworthiness value is decreased by the same amount. The trustworthiness value decrease/increase is also computed by applying the consensus operator of the SL to the actual trustworthiness value of the context provider and to feedback received [8].

In case the context-aware service adaptation is not satisfactory the service users have the possibility to indicate positive or negative experiences, and indicate exactly which faulty context-aware adaptation behavior they are experiencing. Based on the specific feedback from the users, the service provider is able to detect which context provider is not fulfilling his promises regarding the quality of context and the trustworthiness value is decreased.

In order to be able to collect relevant feedbacks, the context-aware service provider has to be able to map the positive and negative feedbacks regarding the context-aware service adaptation to the context provider that influences that behavior. For example, if there is a negative feedback regarding a context-aware meeting service stating that one meeting attendee arrived after the predicted time for the meeting, while the service indicate that the user would be on time, this may be related with the fact that the location information of this person provided to the service provider was of low quality. The positive and negative feedbacks capture the situation where the context provider provides context at a lower quality then advertised, because he is either dishonest, incompetent, or unreliable.

The assumption we make is that if the context-aware service is not adapting accordingly the context provider is the one to blame, because the context information despite being advertised of being capable of providing context at a certain quality in reality is not. We support with our model also the situation where the context provider advertises a certain quality level and always provides context at a quality level lower then the advertised. In this case the trustworthiness level of the context provider will never increase for the advertised quality level because the trustworthiness value will increase taking into account the QoC level that is attached to the context information instance. In this case the context provider is not being dishonest, because it still provides the correct QoC specification.

5 QoC Management Architecture and Prototype

As a proof of concept of our QoC model we present in this Section our QoC management architecture and prototype implementation. In our architecture (Figure 6), the context-aware service provider interacts with the trust provider to query trustworthiness values about context providers and to provide feedback regarding the QoC level advertised and provided by the context provider. The trust provider manages the trust database and interacts with trusted 3^{rd} parties in case a trustworthiness value for a context provider is not available. The trust database stores a trustworthiness value indexed per context provider, context owner, and quality level.

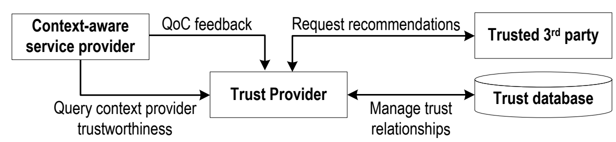

Figure 6. Quality of Context and Trustworthiness Management Architecture

Figure 7 shows our prototype implementation. In the the upper part of the screen we show the discovery of context providers for a certain context type (e.g. ”Ambient temperature”) and context owner (e.g. ”Ricardo Neisse”). The discovery results are shown in a tree and together with each discovered context provider the respective trustworthiness value retrieved from the trust provider for the advertised QoC level and context owner. The result list shows two context providers discovery results for the user ”Ricardo Neisse” where the first one is ranked as ”trustworthy” and the second one is ranked as ”untrustworthy”. For each discovered context provider it is possible to see the advertised context owners (* indicating all users), the precision, and the temporal resolution.

Also in Figure 7 we show the query of the selected context provider from the discovery results. The context provider’s query result shows the context value, the timestamp information, the context precision, the timestamp resolution, and the respective trustworthiness for the ”Ambient temperature” context provider. After receiving the result it is possible to provide a negative or positive feedback, which

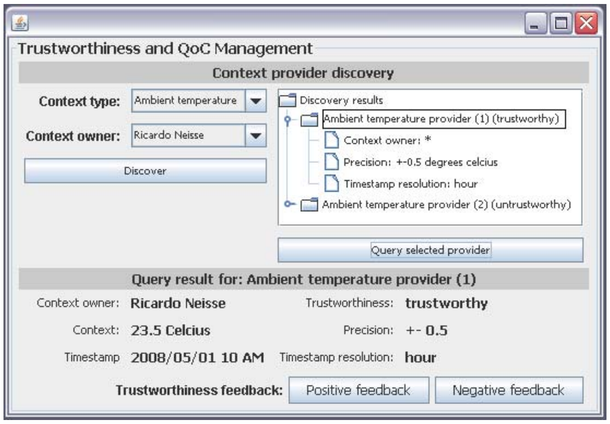

Figure 7. Prototype screen

triggers respectively an increase/decrease of the trustworthiness value. For this trustworthiness calculation we use the consensus operator from the Subjective Logic API [8], for more details on the usage of this operator please refer to our previous work on trust calculus for context-aware service platforms [13] were we have followed a similar approach.

It is important to notice that the trustworthiness value shown is regarding the specific context owner "Ricardo Neisse" and not the advertised context owner "*", which indicates *any* user. As a proof-of-concept we have implemented our QoC model only for numeric context types, and the trustworthiness management according to the description provided in the previous Section, however, our model can be extended with other context types as required. We also do not support in this version of our prototype the exchange of recommendations.

6 Related Work

One of the first papers about Quality of Context has been written by Buchholz et al. [3]. They define QoC as "*any information that describes the quality information that is used as context information*" and complement their definition saying that "*QoC refers to information and not to the process nor the hardware component that possibly provides the information*". Buchholz et al. introduce a contraction when they define trustworthiness because according to their definition *trustworthiness is used by the context provider to rate the quality of the actor from which the context provider originally received the context information*, which is clearly opposed to the first definition of QoC that states that QoC is about the information and not the process nor hardware component that provides the information.

Sheikh et al. [15] have extended the QoC definition of Buchholz et al. by splitting the resolution quality attribute into two different types: temporal and spatial resolution. According to Sheikh et al. these quality attributes do not apply to all types of context and are only relevant for context information about physical entities. Sheikh et al. go one step further then the other QoC models because they not only propose a QoC model but also quantification strategies for the quality attributes and show how QoC can be used for privacy protection. Sheikh et al. do not develop further in the concept of trustworthiness and consider this as future work.

Huebscher et al. [6] also propose a QoC model and an algorithm to rank context providers according to their QoC capabilities. They represent the QoC attributes as a vector and use the Euclidean distance as a ranking metric for context providers. Their QoC model does not address quantification of the QoC attributes and they do not specifically mention the quality attributes that should be included in their model, they only mention precision and refresh rate as examples and do not develop this further. In another paper [7], Huebscher et al. propose a learning model for QoC trustworthiness which is similar to our approach in the sense that feedbacks are also used for trustworthiness calculations. Despite this similarity, their model does not index the trustworthiness values for specific context owners and they also do not provide a QoC management architecture nor prototype implementation to proof the feasibility of their work.

7 Conclusions and Future Work

We define in this paper a QoC model based on the existing literature that uses as a reference an existing ISO standard metrology vocabulary. Our QoC model clearly distinguishes the important quality concepts and aligns the terminology of existing QoC models providing a more accurate vocabulary. This model directly benefits developers of context-aware service platforms because it shows the difference between the QoC quality concepts and how they can be applied in practice, including the support for the trustworthiness quality aspect that has not been concretely supported by existing QoC models.

Another major contribution of our work is to show hour to apply our QoC model in a QoC management architecture that supports context-aware service providers in the management of QoC and trustworthiness values and helps in the selections of trustworthy context providers. We prove the feasibility of our QoC model and management architecture through our proof-of-concept prototype implementation that shows how our model and architecture can be used to define QoC attributes and manage the trustworthiness of context providers that returns numeric values of ambient temperature.

As future work we plan to evaluate if users of context-aware services are likely to provide feedback about the reliability of the adaptation of their services and verify the validity of our trustworthiness calculus using Subjective Logic in a real scenario. Furthermore we want to include in our trustworthiness calculus the fact that a context provider might have different QoC levels for different circumstances, in other words, the context may influence the quality level of the context provider. We also want to verify if users are more likely to report negative experiences then positive ones, which would mean that our trustworthiness value for a context provider should be increased automatically with the number of context information requests in case negative feedbacks are not received. At last we plan to use our QoC management solution to automatically manage and adapt the QoC levels of context providers in a scenario where the context-aware service providers and context providers are part of the same administrative domain.

Acknowledgment

This work is part of the Freeband AWARENESS project. Freeband is sponsored by the Dutch government under contract BSIK 03025. Ricardo Neisse is supported by a CNPq Scholarship, Brazil.

References

[1] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6*, page 6007, Washington, DC, USA, 2000. IEEE Computer Society.

[2] F. Almenárez, A. Marín, C. Campo, and C. R. García. A pervasive trust management model for dynamic open environments. In *First Workshop on Pervasive Security and Trust in MobiQuitous 2004, Boston, USA*, 2004.

[3] T. Buchholz, A. Küpper, and M. Schiffers. Quality of context information: What it is and why we need it. In *Proceedings of the 10th HPOVUA Workshop, Geneva, Switzerland*, Jul. 2003.

[4] CEI and ISO. *International vocabulary of basic and general terms in metrology (VIM)*. International Organisation for Standardization, 2004.

[5] T. Grandison and M. Sloman. A survey of trust in internet application, ieee communications surveys, fourth quarter, 2000.

[6] C. Huebscher and A. Mccann. An adaptive middleware framework for context-aware applications. *Personal Ubiquitous Comput.*, 10(1):12–20, December 2005.

[7] M. C. Huebscher and J. A. McCann. A learning model for trustworthiness of context-awareness services. In *PERCOMW '05: Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 120–124, Washington, DC, USA, 2005. IEEE Computer Society.

[8] A. Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(3):279–311, 2001.

[9] M. Krause and I. Hochstatter. Challenges in modelling and using quality of context (qoc). In *Mobility Aware Technologies and Applications (MATA), Montreal, Canada*, volume 3744 of *Lecture Notes in Computer Science*, pages 324–333. Springer, October, 2005.

[10] A. Levi, M. U. Caglayan, and C. K. Koc. Use of nested certificates for efficient, dynamic, and trust preserving public key infrastructure. *ACM Trans. Inf. Syst. Secur.*, 7(1):21–59, 2004.

[11] R. C. Mayer, J. H. Davis, and D. F. Schoorman. An integrative model of organizational trust, 1995. The Academy of Management Review, Vol. 20, No. 3. (1995), pp. 709-734.

[12] D. H. Mcknight and N. L. Chervany. The meanings of trust, 1990. http://www.

[13] R. Neisse, M. Wegdam, M. van Sinderen, and G. Lenzini. Trust management model and architecture for context-aware service platforms. In *In: Proceedings of the 2nd International Symposium on Information Security (IS07), November 26-27, Vilamoura, Portugal*, pages 1803–1820, 2007.

[14] K. Quinn, D. Lewis, D. O'Sullivan, and V. P. Wade. Trust meta-policies for flexible and dynamic policy based trust management. In *POLICY '06: Proceedings of the Seventh IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'06)*, pages 145–148, Washington, DC, USA, 2006. IEEE Computer Society.

[15] K. Sheikh, M. Wegdam, and M. J. van Sinderen. Quality-of-context and its use for protecting privacy in context aware systems. *Journal of Software (JSW)*, 3(3):83–93, Mar. 2008.

[16] M. J. van Sinderen, A. T. van Halteren, M. W. H. B. Meeuwissen, and E. H. Eertink. Supporting context-aware mobile applications: an infrastructure approach. *Communications Magazine, IEEE*, 44(9):96–104, Sept. 2006.