Chapter 37
Quality of Context and Mobile Systems:
Past, Present, and Future

Atif Manzoor
Vienna University of Technology, Austria

Hong-Linh Truong
Vienna University of Technology, Austria

Ahmad Kamran Malik
Vienna University of Technology, Austria

Schahram Dustdar
Vienna University of Technology, Austria

ABSTRACT

Mobile and pervasive environments are characterized by a plethora of handheld computation and communication enabled devices working as a smart assistant for users by collecting context from the environment and adapting to dynamically changing situation in the environment. Their capabilities are heavily affected by the quality of context information that has been considered unsatisfactory since the start of research in context-aware systems. Early context-aware systems have also tried to collect extra information that can make the applications aware of the quality of context information. Subsequently term Quality of Context (QoC) is coined to indicate quality of context information. In this chapter, the authors present an overview of research efforts undertaken to realize QoC parameters. They also present a summary of works that have used QoC parameters to optimize the performance of different tasks in context-aware systems. Finally, the chapter discusses future directions of QoC research efforts.

DOI: 10.4018/978-1-61520-655-1.ch037

1. INTRODUCTION

Context-awareness is a key requirement for applications in mobile and pervasive environments. Several middleware solutions have supported the design and development of these applications by performing different tasks, such as, acquiring data from sensors, extracting high level context information, and providing context information to mobile applications (Baldauf et al. (2007)). However, as observed, many conflicting situations can arise during the execution of these tasks (Chantzara et al. (2006), Ranganathan et al. (2004)). These conflicting situations affect the quality of context information that has been considered imperfect since the start of the research in context-aware systems (Dey et al. (2005)). Awareness about the quality of context information can help the applications to use context information effectively.

Research efforts have been undertaken to design context models that consider the imperfection of context information and strive to present and associate it with context information. Few works have also used metadata to indicate the characteristics of context information and discussed the advantages of presenting metadata with context information (Hönle et al. (2005)). Subsequently, the term Quality of Context (QoC) has been coined in (Buchholz et al. (2003)) and is defined as *"any information that describes the quality of information that is used as context information"*. Later on, QoC has also been defined in (Krause et al. (2005)) as "any inherent information that describes context information and can be used to determine the worth of information for a specific application". Important QoC parameters that have been identified are up-to-dateness, trustworthiness, resolution, precision, probability of correctness, and completeness of context information (Buchholz et al. (2003), Gray et al. (2001), Krause et al. (2005)). The characteristics of sensors, situation of a specific measurement, values expressed by context information object itself, and the granularity of representation format have also been recognized as the significant sources to determine QoC (Krause et al. (2005)).

In this article we discuss the state of art of the research that has been undertaken to realize, support, and apply QoC parameters to improve the performance of context-aware systems and to optimize the utilization of the scare resources in mobile and pervasive environments. First we discuss the concept of QoC and analyze QoC parameters that have been presented in literature. Then we examine the context models that have been designed to present QoC along with context information (Section 2). Later, we present the analysis of the approaches, algorithms, and mechanisms that have been used to evaluate various QoC parameters (Section 3). Afterwards, we discuss about how context-aware systems can take advantage of QoC parameters in performing the tasks to acquire and provide context information to context-aware mobile applications (Section 4). Finally, we discuss future research directions to realize and apply QoC in context-aware mobile applications (Section 5) and conclude this study (Section 6).

2. QOC AND CONTEXT MODELS

Context-aware systems aim to adapt their behaviour to the prevailing situation in the environment by sensing and using context information (Buchholz et al. (2003)). But sensing context information is a far more difficult task than explicit input to a system (Gray et al. (2001)). Sensors generate a volume of data that can also differ with each other considering the frequency of updates, capability of sensor to collect a specific type of context information, accuracy of sensor that is used to collect information, processing that has been done on that information, and representation format (Chantzara et al. (2006), Cook et al. (2007)). Continuously emerging situation and dynamic nature of the entities in mobile and pervasive environments also imposes major limitations to the quality of

context information. Even in early years of research in context-aware systems (Schimidt et al. (2000)), it has been recognized that there can be errors in sensing context information in different circumstances. Dey et al. when describing context information model of Context Toolkit (Dey et al. (2001)) also asserted that in contrast to general assumption, context information is usually inaccurate and ambiguous. Judd et al. stated in describing Aura (Judd et al. (2003)) that context information is dynamic and typically has uncertainty associated with it. Apart from above mentioned works (Gray et al. (2001)), (Castro et al. (2000)), and (Rangaanathan et al. (2004)) also recognized the problems of imperfection of context information as shown in Table 1. In these circumstances it is necessary to collect some additional information about the quality of context information and present it along with context information. In the remaining section, we will discuss the works that aimed at presenting that additional information as meta-data and later as QoC. A summary of these works is shown in Table 1.

2.1 Imperfection of Context Information

In (Gray et al. (2001)), they emphasize that sensing context information from the environment is a lot more complex task than explicit input to a system. They proposed the term of *"sensed context"* and defined it as *"properties that characterize a phenomenon are sensed and that are potentially relevant to the tasks supported by an application and/or the means by which those tasks are performed"*. They also proposed a model for sensed context information that aims at handling those complex issues. Along with the information about a particular sensed phenomenon, they also suggested to collect meta-information about properties of sensed context. This meta-information includes forms of representation, information quality, sensory source, transformation, and actuation. They include properties like coverage, resolution, accuracy, repeatability, frequency of sample rate, and timeliness as information quality attributes. They also discussed how the nature of sensed context and meta-data can be used for application design and development and presented global view of their architecture for handling context information.

In (Ebling et al. (2001)), they discussed different design issues faced by a context-aware system. These issues include protecting privacy of users, scalability and extensibility of systems, and synchrony of operations. Apart from those issues, they also discussed quality of context information (QoI) as the extent to which context information corresponds to real world. They identified source of context information as important factor in evaluating quality of context information. They discussed freshness and confidence as important QoI metrics and stressed on the need of work to address the issues related to QoI. In (Castro et al.(2000)), they discussed about QoI and presented it by having a measure of accuracy and a measure of uncertainty in the most likely value of query variable.

Table 1. Different stages in realization of imperfection of context information as QoC

Different stages in realization of imperfection of context information	Works that realized imperfection of context information
Considered imperfection of context information	Schimidt et al. (2000), Castro et al. (2000), Gray et al. (2001), Dey et al. (2001), Rangaanathan et al. (2004).
Modelled imperfection of context information as meta-data	Lei et al. (2002), Henricksen et al. (2004), Hönle et al. (2005).
Modelled imperfection of context information as QoC	Buchholz et al. (2003)), Krause et al. (2005), Razzaque et al. (2005).

In (Dey et al. (2001)), they discussed the issues related to the acquisition and representation of context information and privacy of context information. They asserted that in contrast to general assumption, context information is usually inaccurate and ambiguous. They suggested that there are three approaches to deal with issue: (i) pass ambiguity on to applications (ii) attempt to disambiguate data automatically (iii) attempt to disambiguate data manually. They also suggested that an application should mention its accuracy requirements to context acquisition and provisioning framework and context should be provided to those applications according to their requirements. They suggested fusing data from multiple sources to improve the accuracy of data. But they only mentioned accuracy of sensor data as quality of context metric.

These early works clearly recognized the problem of imperfection of context information but mostly they concentrated to resolve this problem by doing probabilistic reasoning performed by the applications using context information. This undue burden on the context-aware applications affected their capability to concentrate on their main task to adapt to emerging situations in pervasive environments. These works also suggested some parameters to represent the quality of context information. Later works have recognized more parameters that can be used to indicate quality of context information and used those parameters as meta-data with context information. In the next section we will discuss about those works.

2.2 Imperfection Presented as Metadata

In (Lei et al. (2002)), they presented the design of a middleware infrastructure for context collection and dissemination realized as a context service. They emphasized that privacy, quality of information, and extensibility is very critical issues in any context acquisition and dissemination system. They extend the idea of QoI presented in (Castro et al. (2000)) and emphasized those sources of context data should be allowed to express the inaccuracy and uncertainty of data. They recommended using timestamp indicating the freshness of context data and confidence asserted by the data source as QoI metrics.

In (Henricksen et al. (2002)), they presented a scenario to emphasize the importance of context information in particular situations. Their work discussed different characteristics of context information and emphasized that context information can be static and dynamic. They characterized the context information as imperfect and stated that *"information may be incorrect if it fails to reflect the true state of the world it models, inconsistent if it contains contradictory information, or incomplete if some aspects of context are un-known"*. Finally they also tried to associate the quality measures of freshness, accuracy, and certainty of context information in a context model. In (Henricksen et al. (2004)), they stressed the fact that contrary to general assumption about the quality of context information being perfect; context information can be unknown, ambiguous, imprecise, and erroneous. They claim that imperfect context information presents a significant obstacle to the success of context-aware applications that is commonly overlooked. They stressed that conflict in context information should be resolved early using conflict resolution techniques. They also categorized the context information as sensed, static, profiled, and derived and discussed sources that are used to obtain context information, quality issues, and sources of inaccuracy that are associated with them. Finally, this work also presented a model for context information.

In (Hönle et al. (2004)), they presented a context model integrated with meta-data. They emphasized that meta-data, giving additional information about data, improve the operational value of data and can be used for resource finding, enhanced data selection, trust and data quality issues, and sensor fusion. They categorized meta-data as system generated, technically measurable,

technical restrictions, authorship, and cost and asserted that this data can be used to derive quality metrics associated with data such as reliability, precision, consistency, age, and access control. Above mentioned works indicated the factors that affect the quality of context information. These factors include type of context information that can be sensed, static, profiled, or derived, characteristics of source of context information, and the features of environment where context information is gathered. Many parameters indicating the quality of context information also be defined and modelled with context information. Still there was a lack of formal grouping and definition of those metrics. In the next section we discuss the works that recognized the concept of QoC and indicated parameters to present QoC.

2.3 Imperfection Presented as QoC

Since the term QoC was first defined in (Buchholz et al. (2003)), various context models have strived to present QoC along with context information. Different parameters have also been considered to present QoC information. In this section we will discuss about those context models that realize and support the concept of QoC. We will also discuss about the QoC parameters that have been proposed in those context models and why those parameters are important to characterize QoC information in context-aware mobile application.

In (Buchholz et al. (2003)), they defined Quality of Context as *"Quality of Context (QoC) is any information that describes the quality of information that is used as context information. Thus, QoC refers to information and neither to the process nor the hardware component that possibly provide the information"*. They presented precision, probability of correctness, trustworthiness, resolution, and up-to-dateness as important QoC parameters. They also compared QoC with Quality of Service (QoS) that gives the information about the performance of a service and Quality of Device (QoD) that tells us about technical properties and capabilities of a device. They emphasized that although these three quality metrics are different from each other, even then they can influence each other. This work also presented scenarios on how context providers can cooperate with Context-Aware Service (CAS) providers by sensing the context from the environment, refining context information by doing reasoning on it, and finally providing it to CAS provider who used this context information to adopt the behaviour of context-aware services. Their work has also discussed scenarios that illustrate where and why we need QoC. Finally, they compared their work with early works that has also considered quality of context information in context-aware systems.

In (Krause et al. (2005)), they presented the necessity of QoC parameters, analyzed a general context provisioning process, and derived requirements for QoC. They also gave a new definition of QoC as *"QoC is any inherent information that describes context information and can be used to determine the worth of the information for a specific application. This includes information about the provisioning process the information has undergone (history, age), but not estimations about future provisioning steps it might run through."* They identified the sources of QoC parameters as the characteristics of sensor, situation of specific measurement, value expressed by the context information object itself, and granularity of representation format.

In (Razzaque et al. (2005)), they analyzed different existing approaches to model context information. These approaches include set theory, directed graph, first-order logic, and preferences. Later they discussed the dependency relationship which is a special type of relationship that exists between context entities and attributes. They also stressed that Quality of Context Information (QoCI) should also be modelled as part of context information models and user of context information or applications will also be provided with this quality of context information.

Although there are many efforts that tried to model QoC with context information, they lack common terminology for QoC parameters. Different QoC parameters have been used to show same concepts. For example, time resolution of context information has been recognized as up-to-dateness (Buchholz et al. (2003)), timeliness (Gray et al. (2001)), staleness (Henricksen et al. (2004)), refresh rate (Huebscher et al. (2004)), and age (Hönle et al. (2005)). All other QoC parameters also have multiple terms presenting the same concept. Table 2 shows the summary of QoC concepts and different form of representations that have been used for those concepts. There is a need of standard name for each concept. These works also did not provide the QoC parameters in a form that can show the worth of context information for an application and to allow those parameters to be used by an application. They did not make any distinction between the QoC indicators which can be used to calculate high level QoC parameters for application usage.

3. EVALUATION OF QOC PARAMETERS

Though QoC parameters have been indicated and context information models have been designed to accommodate these parameters with context information, few works have tried to evaluate these parameters. In this section we will discuss about the works that have tried to evaluate the QoC parameters according to the needs of applications using context information. We will also discuss about how existing frameworks provide QoC information to context-aware applications.

In (Kim et al. (2006)), they proposed accuracy, completeness, representational consistency, access security and up-to-dateness as QoC parameters and presented statistical method to calculate accuracy of sensor data. However, their method to measure accuracy is more appropriate in those cases where sensors get continuous data around some average value, e.g., data from temperature sensors. They also measured the completeness of

Table 2. Different representations used for QoC parameters

Quality Concept	Metric Suggested	Description	Suggested by
temporal	Timeliness	range of measure in time	Gray et al. (2001)
	up-to-dateness	age of context information	Buchholz et al. (2003))
	Staleness	out of time for use	Henricksen et al. (2004)
	refresh rate	how often to receive a new measurement	Huebscher et al. (2004)
	Age	how old is data	Hönle et al. (2004)
	Frequency	sampling rate	Gray et al. (2001)
correctness	Accuracy	information is measured correctly	Gray et al. (2001)
	probability of correctness	probability that information is correct	Buchholz et al. (2003)
observation level	Resolution	Smallest perceivable element	Gray et al. (2001)
	Precision	exactness of measurement	Buchholz et al. (2003)
information amount	Coverage	amount of sensed context	Gray et al. (2001)
	completeness	all aspects of context are known	Kim et al. (2006)
trust on sensor	Reliability	degree of confidence on sensor	Gray et al. (2001)
	trustworthiness	how likely is that sensor provided correct information	Buchholz et al. (2003)
	Authorship	information about sensor	Hönle et al. (2004)

a context object as the ratio of available attributes to the total number of attributes for a specific context object.

In (Sheikh et al. (2008)), they presented five QoC parameters as precision, freshness, spatial resolution, temporal resolution, and probability of correctness and tried to quantify these parameters. They considered different options that can be used to interpret and represent QoC parameters for different type of context information. For example, they discussed that boolean, numeric, complex types with an incremental structure, and unordered complex types can be used to present precision of different type of context information. They also discussed the options that can be used for other types. In (Tang et al. (2007)), they presented a context quality model based on OWL-DL and used a function that is based on a specific application to evaluate QoC value of certain context. They also illustrated with different scenarios that how the value of QoC parameters can be changed with the change in current situation and application.

In (Toivonen et al. (2006)), they used quality attributes to calculate trust in range [0..1]. They used different formulas for evaluation of trust in different situations. Their system evaluates trust in two steps. The first step is the traditional calculation of trust using quality attributes, e.g., using recommendations. The second step adjusts the trust value calculated in first step by using the context attributes. Adjustment function uses a context based predicate and weights. In their work, if context predicate condition is true then the weight for this context attribute is increased otherwise it is decreased. Different weights can be assigned by the user to increase or decrease the context attribute values in different conditions. For example, if a user wants to select a web application which requires less memory then he can assign increasing and decreasing values for relative memory requirement.

To calculate trust Neisse et al. in (Neisse et al. (2007)) used Subjective Logic which expresses trust with a triple belief, disbelief and uncertainty. The results of these functions are mapped on a set {VT, T, U, VU} that describes very trustworthy, trustworthy, untrustworthy and very untrustworthy as follows. If the belief is higher than disbelief, the result is trustworthy, if it has uncertainty not lower than 1/3 and very trustworthy otherwise. But if belief is not higher than disbelief, it is considered untrustworthy if it has uncertainty not lower than 1/3 and very trustworthy otherwise. The used a recommendations manager to establish indirect trust with new entities about which user is not already aware. This trust is based on information received from other entities.

In (Manzoor et al. (2008)), they related QoC to the worth of context information for an application. They classified QoC into QoC sources and QoC parameters. QoC sources are the information about the sources that collect context information, the environment where that context information is collected, and the entities about which the context information is collected. QoC parameters, such as, up-to-dateness, trustworthiness, completeness,

Table 3. Works that evaluated QoC parameters

QoC Parameters	Evaluated in works
Up-to-dateness	Manzoor et al. (2008), Schmidt et al. (2006).
Trustworthiness	Toivonen et al. (2006), Neisse et al. (2007), Manzoor et al. (2008).
Completeness	Kim et al. (2006), Manzoor et al. (2008).
Significance	Manzoor et al. (2008).
Accuracy	Kim et al. (2006).

and significance of context information is evaluated using QoC sources.

The research works presented above have evaluated very few QoC parameters as shown by the Table 2. Different data representations have also been used to quantify the QoC parameters. (Sheikh et al. (2008)) have shown different options that can be used to quantify QoC parameters and used the numbers in range [0..3] to use QoC parameters in a scenario to enforce privacy. (Manzoor et al. (2008)) had evaluated the QoC parameters as a real number having value in rang [0..1]. There is a need to standardize the format to representation QoC parameter quantities and design a framework that can provide QoC parameters along with context information to context-aware applications. In next section we discuss about the works that have used QoC parameters to perform different tasks in context-aware systems.

4. CONTEXT MANAGEMENTS SYSTEMS USING QOC

QoC parameters not only indicate about the up-to-dateness, completeness, precision, and significance of context information but also provide the information about the trustworthiness of the source of context information. Context information management systems can take advantage of QoC parameters in performing the tasks, such as, source selection in acquiring context from sensors, reasoning on raw context data to extract high level context information, aggregating high level context information, context query routing, and privacy enforcement while sharing context information. In this section we discuss how different systems have used QoC parameters to enhance their efficiency while providing the context information to high level applications and other peers in mobile and pervasive environment.

An approach is presented in (Chantzara et al. (2006)) that used quality of information for evaluating and selecting the information to be used as context information. They calculated a utility function based on QoC attributes. But in their work QoC attributes are completely provided by the context sources that can bias the decision for source selection. (Huebscher et al. (2004)) used QoC parameters in their adaptive middleware for context-aware applications in smart homes. They have also used QoC parameters to perform different tasks in their middleware such as context provider selection. But their work is based on the assumption that context providers are able to estimate QoC parameters and provide them to their middleware. In (Bu et al. (2006)), they have considered the QoC parameters such as delay time, context correctness probability, context consistency probability and correlation among those parameters. They also calculated another metric relative frequency and used it to resolve inconsistency among various context objects.

(Sheikh et al. (2008)) quantified the QoC parameters, such as, precision, spatial resolution, temporal resolution, freshness, and probability of correctness to have the value in range [0..3] and used those parameters to enforce their privacy policy in a health information system. Owner of context information can specify the quality of context information that can be accessed by caregivers. For example, only the information about the city of a patient is shared in normal situations while in emergencies complete location information including city, street, postcode, and house number are shared. (Breza et al. (2007)) also used QoC for source selection and providing autonomic behaviour in wireless sensor networks. A requestor makes QoC requests and different sensors send their QoC information to the requestor. These values are evaluated at the requestor node to select a sensor that fulfils QoC requirements. As soon as an acceptable sensor has been found, the requestor sends a message to other sensors to stop sending QoC values. As QoC parameters change over time, they also suggested an autonomously managed system for QoC parameters.

Pawar et al. (2007) also proposes a context distribution framework in which context sources are selected on the basis of QoC parameters. When a new context source is registered in the service directory, the registration information also includes the information about the capabilities of context source. Information about the capabilities of context sources is defined by the information about source, information about the entity about which context is collected, context type, and QoC parameters about context information. This capability information is used to make selection among different sources of context information. In (Toninelli et al. (2009)), they have taken QoC parameters into account to make access control decisions. In (Manzoor et al. (2008)) have used QoC parameters to define the policies to resolve the conflicts in context information. They have defined the conflict resolving policies on the bases of QoC parameters such as up-to-dateness, trustworthiness, completeness, and significance. These policies can be used individually or in combination with each other depending upon the context of use of information and the requirements of a specific application.

In the above-mentioned works QoC is mostly used to resolve context conflicts and enforce privacy of context information. Conflicts are resolved in the favour of context object having greater values of QoC parameters. In some critical situations, in spite of the best context object, applications can also request for the context objects having the values of QoC parameters more than a threshold value. These context objects may have the conflicting information. In this case applications may need to reason on the basis of QoC parameters. A question arises that how applications can use the QoC parameters to resolve the conflict in such situations. There can also be the situations where more sophisticated reasoning is required to make decisions on the basis of these parameters. These reasoning can be done on the basis of probability theories such as fuzzy logic(Zadeh (1968)), Bayesian (Braun (2000)) and Dempster-Shafer(Shafer (1965), Dempster(1976)) theories or on the basis of neural networks (Parsons (2003)). Question arises such as how can we use these parameters with those theories. Apart from tasks mentioned in Table 4, QoC parameters can also be used to improve the performance of many other tasks at different layers of a context management system. Future works need to exploit QoC parameters to their full potential.

5. FUTURE DIRECTIONS

Collecting context from environment is just a step to become context-aware. Adaptability and robustness of context-aware systems is very much dependent on the quality of context information. Currently context-aware systems have not been putting enough effort to realize, evaluate and use QoC information to improve their performance. In this section we will discuss the future direction that can be set for research in QoC. Apart from many works to model QoC parameters with context information, there is still lack of standard terminology to represent those QoC parameters. Different metrics have been proposed to repre-

Table 4. Works that have used QoC parameters

Functionality performed	Used in works
Context Selection	Huebscher et al. (2004), Chantzara et al. (2006), Pawar et al. (2007), (Breza et al. (2007))
Inconsistency Resolution	(Bu et al. (2006)), Manzoor et al. (2009)
Privacy Enforcement	Neisse et al. (2007), Skeikh et al. (2008)
Context Aggregation	Manzoor et al. (2009)

sent similar quality concepts as we have shown in Table 2. Future steps to make use of QoC to its full potential will include having a standard terminology to represent the concepts that indicate the quality of context information. This standardization of terminology will prevent the researchers to repeatedly put efforts in defining the same concepts with different representations. Most works that have tried to model QoC suggested QoC parameters by considering only a few types of context information. Significance of those parameters also changes with different type of context information. For example, up-to-dateness of information may not be very useful in case of profile information of person that participate in a collaborative task in pervasive environments. We need to distinguish these generic QoC parameters with the level of their significance for a specific application and domain.

Classifications of context information such as sensed, profiled, and derived can also be useful. QoC parameters for each type of context information can be selected on the basis of class to which that type of context information belongs. There is also a need to devise a mechanism in which user of the context information can decide about the useful QoC parameters according to the context of use of information and request to provide those parameters. These QoC parameters are not equally useful in different domains and applications. QoC parameters considering different domains and groups of applications should also be defined.

There has also not been much work to evaluate QoC parameters. As a first step in this regard, we need to clearly define the sources that affect those QoC parameters and distinguish those sources on the bases whether they change dynamically or they can be statically profiled. We also need to look how the sources that affect QoC parameters can be collected from the environment and used to evaluate QoC parameters. We also need to decrease the role of user in the evaluation of QoC parameters and devise an autonomic system for the management of QoC parameters. We can enhance the QoC parameters values by combining two different context objects having same information. So far QoC parameters are mostly used in resolving conflicts in context information, privacy enforcement, and context source selection. QoC parameters can further be used to perform the functions of context query routing, context dissemination model, context fusion, and decision making to improve the performance of context-aware systems.

6. CONCLUSION

In this article we have presented the state of the art of the research to support and realize QoC in mobile and pervasive environments. We have discussed how QoC can be useful in the design of a context-aware system to improve its efficiency and optimally use the scare resources in mobile and pervasive environments. We have observed that although a lot of QoC parameters have been discussed, there is a lack of standard terminology for QoC parameters and different representations have been used for similar concepts. There have been a few works that have tried to evaluate and utilize these parameters. There is a need of system that evaluates the selected QoC parameters and employs those parameters to perform various tasks in context management systems. Future research in QoC should minimize the role of user profiled static information to evaluate QoC parameters. Research efforts are also needed to utilize QoC parameters not only in the middleware tasks but also in context-aware mobile applications that can use these parameters for reasoning on context information to make decisions.

REFERENCES

Baldauf, M., Dustdar, S., and Rosenberg, F. (2007) A survey on context-aware systems. *International Journal of ad Hoc and Ubiquitous Computing, 2*(4), 263–277.

Braun, J. (2000). Dempster–Shafer theory and Bayesian reasoning in multisensor data fusion. *Sensor Fusion: Architectures, Algorithms and Applications IV. Proceedings of the Society for Photo-Instrumentation Engineers, 4051*, 255–266.

Breza, M., Anthony, R., & McCann, J. (2007) *Quality of context driven autonomicity*. In 2nd International Workshop on Engineering Emergence in Decentralised Autonomic Systems, 2007.

Bu, Y., Gu, T., Tao, X., Li, J., Chen, S., & Lu, J. (2006) Managing quality of context in pervasive computing. In *QSIC '06: Proceedings of the Sixth International Conference on Quality Software,* (pp. 193–200). Washington, DC: IEEE Computer Society.

Buchholz, T., Kpper, A., & Schiffers, M. (2003). Quality of context: What it is and why we need it. In *Proceedings of the 10th International Workshop of the HP OpenView University Association* (HP-OVUA). Hewlet-Packard OpenView University Association.

Castro, P., & Muntz, R. (2000). Managing context data for smart spaces. *Personal Communications, 7*(5), 44–46. doi:10.1109/98.878537

Chantzara, M., Anagnostou, M., & Sykas, E. (2006) Designing a quality-aware discovery mechanism for acquiring context information. In *AINA '06: Proceedings of the 20th International Conference on Advanced Information Networking and Applications - Volume 1 (AINA'06)*, (pp. 211–216). IEEE Computer Society.

Cook, D. J. (2007). Making sense of sensor data. *IEEE Pervasive Computing / IEEE Computer Society and IEEE Communications Society, 6*(2), 105–108. doi:10.1109/MPRV.2007.37

da Rocha, R. C. A., Endler, M., & de Siqueira, T. S. (2008) Middleware for ubiquitous context-awareness. In *MPAC '08: Proceedings of the 6th International Workhop on Middleware for Pervasive and Ad-Hoc Computing*, (pp. 43–48). New York, NY: ACM.

Dempster, A. P. (1976). *A mathematical theory of evidence*. Princeton University Press.

Dey, A. K., Abowd, G., & Salber, D. (2001). A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction, 16*, 97–166. doi:10.1207/S15327051HCI16234_02

Dey, A. K., & Mankoff, J. (2005). Designing mediation for context-aware applications. *ACM Transactions on Computer-Human Interaction, 12*(1), 53–80. doi:10.1145/1057237.1057241

Ebling, M. R., Hunt, G., & Lei, H. (2001). Issues for context services for pervasive computing. In *Proceedings of the Advanced Workshop on Middleware for Mobile Computing*, Heidelberg, Germany, 2001. Springer.

Gray, P. D., & Salber, D. (2001) Modelling and using sensed context information in the design of interactive applications. In *EHCI '01: Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction*, (pp. 317–336). London, UK, 2001. Springer-Verlag.

Henricksen, K., & Indulska, J. (2004) Modelling and using imperfect context information. In *PERCOMW '04: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, (pp. 33–37). March 2004.

Henricksen, K., Indulska, J., & Rakotonirainy, A. (2002) Modelling context information in pervasive computing systems. In *Pervasive '02: Proceedings of the First International Conference on Pervasive Computing, 2414*, (pp. 79–117). Berlin, Germany: Springer.

Hönle, N., Kappeler, U. P., Nicklas, D., Schwarz, T., & Grossmann, M. (2005). Benefits of integrating meta data into a context model. In *PERCOMW '05: Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops*, (pp. 25–29). Washington, DC: IEEE Computer Society.

Huebscher, M. C., & McCann, J. A. (2004). Adaptive middleware for context-aware applications in smart-homes. In *MPAC '04: Proceedings of the 2nd Workshop on Middleware for Pervasive and Ad-Hoc Computing*, (pp. 111–116). New York, NY: ACM.

Judd, G., & Steenkiste, P. (2003). Providing contextual information to pervasive computing applications. In *PERCOM '03: Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, (p. 133). Washington, DC: IEEE Computer Society.

Kim, Y., & Lee, K. (2006) A quality measurement method of context information in ubiquitous environments. In *ICHIT '06: Proceedings of the 2006 International Conference on Hybrid Information Technology*, (pp. 576–581). Washington, DC: IEEE Computer Society.

Krause, M., & Hochstatter, I. (2005). Challenges in modeling and using quality of context (QOC). In *Proceedings of Mobility Aware Technologies and Applications (Vol. 3744*, pp. 324–333). Berlin, Germany: Springer. doi:10.1007/11569510_31

Lei, H., Sow, D. M., Davis, J. S. II, Banavar, G., & Ebling, M. R. (2002). The design and applications of a context service. *SIGMOBILE Mobile Computing and Communication Review, 6*(4), 45–55. doi:10.1145/643550.643554

Manzoor, A., Truong, H.-L., & Dustdar, S. (2008). On the evaluation of quality of context. In *EuroSSC*. In *Lecture Notes in Computer Science (Vol. 5279*, pp. 140–153). Springer.

Manzoor, A., Truong, H.-L., & Dustdar, S. (2009) Quality aware context information aggregation system for pervasive environments. In *The 5th International Symposium on Web and Mobile Information Services, The IEEE 23rd International Conference on Advanced Information Networking and Applications (AINA-09)*, Bradford, UK, May 26-29, 2009. IEEE Computer Society.

Neisse, R., Wegdam, M., van Sinderen, M., & Lenzini, G. (2007) Trust management model and architecture for context-aware service platforms. In R. Meersman & Z. Tari (Eds.), *OTM Conferences (2), volume 4804 of Lecture Notes in Computer Science*, (pp. 1803–1820). Springer.

Park, I., Lee, D., & Hyun, S. J. (2005). A dynamic context-conflict management scheme for group-aware ubiquitous computing environments. In *COMPSAC '05: Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)* Vol. 1, (pp. 359–364). IEEE Computer Society.

Parsons, O., & Carpenter, G. A. (2003). ARTMAP neural networks for information fusion and data mining: Map production and target recognition methodologies. *Neural Networks, 16*(7), 1075–1089. doi:10.1016/S0893-6080(03)00007-8

Pawar, P., van Beijnum, B. J. F., Peddemors, A., & van Halteren, A. T. (2007). Context-aware middleware support for the nomadic mobile services on multi-homed handheld mobile devices. In *IEEE Symposium on Computers and Communications, ISCC 2007*, 1-4 July 2007, Aveiro, Portugal, (pp. 341-348). IEEE Computer Society. ISBN 1-4244-1521-7

Ranganathan, A., Al-Muhtadi, J., & Campbell, R. H. (2004). Reasoning about uncertain contexts in pervasive computing environments. *IEEE Pervasive Computing / IEEE Computer Society and IEEE Communications Society*, *3*(2), 62–70. doi:10.1109/MPRV.2004.1316821

Razzaque, M. A., Dobson, S., & Nixon, P. (2005) Categorization and modelling of quality in context information. In *Proceedings of the IJCAI 2005 Workshop on AI and Autonomic Communications*. Edinburgh. Scotland.

Schmidt, A. (2000). Implicit human computer interaction through context. *Personal and Ubiquitous Computing*, *4*(2/3).

Schmidt, A. (2006). A layered model for user context management with controlled aging and imperfection handling. In *Modeling and Retrieval of Context* (*Vol. 3946*, pp. 86–100). Berlin, Germany: Springer. doi:10.1007/11740674_6

Shafer, G. (1968). A generalization of Bayesian inference. *Journal of the Royal Statistical Society. Series A (General)*, *30*, 205–247.

Sheikh, K., Wegdam, M., & van Suinderen, M. (2008). Quality-of-context and its use for protecting privacy in context aware systems. *Journal of Software*, *3*, 83–93. doi:10.4304/jsw.3.3.83-93

Tang, S., Yang, J., & Wu, Z. (2007) A context quality model for ubiquitous applications. *IFIP International Conference on Network and Parallel Computing Workshops, NPC Workshops*, (pp. 282–287).

Toivonen, S., Lenzini, G., & Uusitalo, I. (2006) Context-aware trust evaluation functions for dynamic reconfigurable systems. In T. Finin, L. Kagal, & D. Olmedilla (Eds.), *MTW, volume 190 of CEUR Workshop Proceedings*. CEUR-WS.org.

Toninelli, A., Corradi, A., & Montanari, R. (2009). *A quality of context aware approach to access control in pervasive environments* (pp. 236–251). Springer.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*, 338–353. doi:10.1016/S0019-9958(65)90241-X

KEY TERMS AND DEFINITIONS

Context Consumer: Any entity, such as a context-aware application, that uses the context information to adapt its functionality are called context consumers.

Context Provider: Any entity, such as sensors, that collect or drive context information and provide it to other entities, such as a context-aware application is called context-provider.

Context: The information about any entity, such as a person, that is relevant to perform the functionality of an application is called context.

Context-Awareness: The ability to adapt the functionality of an application according the context of a specific entity or prevailing situation is called context-awareness.

Pervasive Systems: The systems that facilitate the human users by pervasively providing the computing power, information, and other services specifically tailored to their needs, such as easy living environments for physically and cognitively impaired persons, remotely providing health care services to chronic patients, and adaptive disaster response systems, are pervasive systems.

QoC Parameters: The metrics that are used to indicate the quality of context information are called QoC parameters.

Quality of Context (QoC): The information about the quality of context is called Quality of Context.