

# Chapter XI

## Context Models and Context Awareness

**Melanie Hartmann**

*Technische Universität Darmstadt, Germany*

**Gerhard Austaller**

*Technische Universität Darmstadt, Germany*

### ABSTRACT

*To support users in performing their tasks, applications need a better understanding of the current situation they are being used in. This chapter gives an overview of how knowledge of the current context, that is, information characterizing the situation, can be represented and how this knowledge can be used for enhancing applications. We discuss what is actually meant by “context” and “context-aware” applications. Further, we describe what has to be considered when building a context-aware application. We thereby focus on the representation of context information and how to deal with its unreliable nature. This chapter should sensitize the reader to the difficulties of using context information and give guidelines on how to build an application that benefits from knowing its current context.*

### INTRODUCTION

Humans use all kinds of information characterizing their current situation, like time, location and identity of persons nearby, to adapt their behavior to the situation and to make decisions. For example, when we speak to a person, we adapt what we say and how we say it to the social rank of the person (e.g., most people would not say “that’s nonsense” to their boss, but would to a friend). All this information is not easily captured, represented and processed by a computer. However, this information can help to build more user-friendly applications that adapt and respond to the user’s

current situation. If the computer were aware of the user’s context and its interpretation, it would be able to make decisions on behalf of the user, anticipating user needs like another human would. For example, it would be possible to provide the user only with information relevant to the current situation and thus reduce the cognitive load. This is especially necessary in the area of ubiquitous computing (UC), where the user has to deal with a multitude of different computers, and thus with a multitude of possible distractions. To enable all these devices disappear into background, they have to anticipate the user’s future demands and adapt to the user’s context to reduce the amount of interaction needed.

The word “context” has its origins in the Latin word “contextus” meaning: “to weave together”, originally denoting the construction of a text. Nowadays, the term is loaded with a variety of different meanings. According to Merriam-Webster’s Collegiate Dictionary, context is defined as “the interrelated conditions in which something exists or occurs”. Even in computer science itself, context is used with a number of different meanings. For example, context in context-free or context-sensitive grammars refers to the symbols that surround a placeholder and determine which strings can replace it. In contrast, “context” in the area of context-aware computing refers to any information that can be used to enhance an application, especially the interaction with the user.

The term “context-aware computing” became popular in the middle of the 90s, when researchers started to develop applications that incorporated the current location of users. By then, location-awareness was regarded as the most important subset of context-awareness. These applications used the location as auxiliary information to improve interaction with the user by adapting it to the user’s needs; thus context-awareness was more or less regarded as synonymous with adaptivity.

*Adaptivity* thereby comprises principally:

- Restricting the user interface to the relevant input possibilities and relevant data;
- Adapting dynamically to the user’s context how the information is presented and how it can be accessed, for example, use of audio output instead of visual output if the user is currently driving;
- Automating actions for the user, for example, prefilling data.

However, there were also a growing number of applications, such as navigation systems, that did not regard location as auxiliary but rather as mandatory information. These applications require location information in order to provide their normal functionality and cannot operate

without this information. As location-awareness was still regarded as a subset of context-awareness, the latter lost its connotation of using auxiliary information for enhancing the interaction. Thus, the term context became even more difficult to define as the discussion in the section “*What is Context?*” illustrates.

Nowadays, context-aware applications go beyond using context for adapting the interaction with the user. For example, they use it as an additional information source for facilitating the later retrieval of data. In the section “*What are context-aware applications?*”, we list the various features making an application context-aware. Further, we discuss the difficulties that have to be faced when using context. In the section “*How to build a context-aware application*”, we describe the design process for building a context-aware application and show how context information can be acquired, represented, accessed and managed. A middleware can be used to support the developer in building the application. We describe the different layers of such a middleware in the section “*Middleware Architectures*”. As context information is highly dynamic and error-prone in comparison to traditional information sources, we have a closer look how to handle uncertainty in context-aware applications in the section “*Dealing with uncertainty*”.

## WHAT IS CONTEXT?

Which constituents of all this information surrounding us can be used to improve an application and thus should be regarded as context for this application? Many researchers have given a definition for the concept of context, but none of them are really widely accepted. In this section, we give an overview of the most commonly used ones.

We illustrate the problem of defining context with an example application of a booking process for train tickets. In Figure 1, we list some information that is available when using the booking application. Some of this information is mandatory

20 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: [www.igi-global.com/chapter/context-models-context-awareness/21771](http://www.igi-global.com/chapter/context-models-context-awareness/21771)

## Related Content

---

### New Hybrid Proposed Solution for Video Steganography Based on Clustering Algorithm

Ahlem Fatnassi, Hamza Gharsellaoui and Sadok Bouamama (2020). *International Journal of Security and Privacy in Pervasive Computing* (pp. 30-43).

[www.irma-international.org/article/new-hybrid-proposed-solution-for-video-steganography-based-on-clustering-algorithm/259341](http://www.irma-international.org/article/new-hybrid-proposed-solution-for-video-steganography-based-on-clustering-algorithm/259341)

### Proposed Abelian ACM Method Optimizing: The Risk on a Real-Time Unix Operating System

Abhishek Asthana and Padma Lochan Pradhan (2021). *International Journal of Security and Privacy in Pervasive Computing* (pp. 33-52).

[www.irma-international.org/article/proposed-abelian-acm-method-optimizing/306596](http://www.irma-international.org/article/proposed-abelian-acm-method-optimizing/306596)

### Contactless Payment with RFID and NFC

Marc Pasquet, Delphine Vacquez, Joan Reynaud and Félix Cuozzo (2010). *Ubiquitous and Pervasive Computing: Concepts, Methodologies, Tools, and Applications* (pp. 112-120).

[www.irma-international.org/chapter/contactless-payment-rfid-nfc/37780](http://www.irma-international.org/chapter/contactless-payment-rfid-nfc/37780)

### From Fixed to Mobile Convergence

John Ayoade (2009). *Auto-Identification and Ubiquitous Computing Applications* (pp. 167-179).

[www.irma-international.org/chapter/fixed-mobile-convergence/5462](http://www.irma-international.org/chapter/fixed-mobile-convergence/5462)

### Learning Methodologies to Support E-Business in the Automated Negotiation Process

Paolo Renna (2013). *Intelligent Technologies and Techniques for Pervasive Computing* (pp. 37-58).

[www.irma-international.org/chapter/learning-methodologies-support-business-automated/76781](http://www.irma-international.org/chapter/learning-methodologies-support-business-automated/76781)