POLITECNICO DI MILANO

School of Industrial and Information Engineering

Department of Electronics, Information and Bioengineering

Master of Science in Computer Engineering

The various facets of context-aware systems: A survey

Supervisor: Prof. Letizia Tanca

Co-Supervisor: Eng. Alireza Javadian Sabet

Candidate:

Seyyed Mohsen Kashfi Haghighi

Matr. 859085

Academic Year 2020-2021

Acknowledgement

I would like to thank the university for allowing me to study at this prestigious and well-known university. I had this chance to learn many things during this journey.

I would like to express my special thanks of gratitude to my supervisor Prof. Letizia Tanca and my co-supervisor, Eng. Alireza Javadian Sabet who gave me the opportunity to work on this great project on the topic “The various facets of context-aware systems: A survey” which also helped me in doing much research and I came to know about so many new things, and without their help and guidance this work could not be finished, so I am thankful to them.

Moreover, I would like to thank Prof. Matteo Rossi, Prof. Fabio A. Schreiber, and Eng. Alessandra Scutti for their valuable comments.

Finally, I would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame and supporting me during the whole university career; thank you so much.

Contents

List of Abbreviations

WWW	*World Wide Web*
CDT	*Context Dimension Tree*
CBR	*Case Base Reasoning*
GPS	*Global Positioning System*
HAVS	*Human advanced-vehicle systems*
IoT	*Internet of Things*
XML	*Extensible Markup Language*
JSON	*JavaScript Object Notation*
UML	*Unified Modelling Language*
ORM	*Object Role Modelling*
SQL	*Structured Query Language*
OWL	*Ontology Web Language*
RDF	*Resource Description Framework*
HMM	*Hidden Markov Mode*
ANN	*Artificial Neural Network*
SVM	*Support Vector Machines*
JCAF	*Java context-awareness framework*
CAMUS	*Context-Aware Mobile mashUpS*
CAIS	*Context Awareness Information Service*
Qos	*Quality of Service*
CaaSSET	*Context as a Service Set of Engineering Tools*
IT-CPS	*Intelligent Transportation Cyber-Physical Systems*
B2G	*Business to Government*
CRAFT	*Comprehensive Resilience Assessment Framework For Transportation*
AUM-IoT	*Adaptive Ubiquitous Middleware for context-aware IoT*
CoBrA	*Context Broker Architecture*
ECADL	*Event-Control-Action Domain-specific Language*
CS	*Context Space*
FSI	*Fuzzy Situation Inference*
SDDL	*Security-related Data Description Language*

List of Figures

List of Tables

Abstract

Context-aware systems aim at providing appropriate contextual information to the right user at the right time. Context-aware systems have a life-cycle for context management, starting with acquiring context through sensors, then analyzing and interpreting them using context modeling and reasoning approaches to make relevant decisions by changing the system behavior based on the user's situation.
This survey investigated approximately 100 papers, mainly from 2000 to 2021, regarding context-aware systems. Studying this survey will help application designers to decide what context model and context-aware methods should be used to have an efficient context-aware application. Also, it allows researchers to understand the essential requirements of context-aware systems. We discussed the background of context-aware systems and context information management. Moreover, we explained numerous context-aware methods and different context challenges which need to be taken into account.

Keywords— *context; context-awareness; context-modeling; context-awareness systems; context-awareness computing; context-awareness life-cycle; context-awareness methods; context-aware systems; contextual information*

Sommario

I sistemi di Context-aware mirano a fornire informazioni contestuali appropriate all'utente giusto al momento giusto. I sistemi di Context-aware hanno un ciclo di vita per la gestione del contesto, a partire dall'acquisizione del contesto tramite sensori, quindi analizzarli e interpretarli utilizzando la modellazione del contesto e approcci di ragionamento per prendere decisioni rilevanti modificando il comportamento del sistema in base alla situazione dell'utente.
questo sondaggio ha esaminato circa 100 articoli, soprattutto dal 2000 al 2021, riguardanti sistemi Context-aware. Lo studio di questo sondaggio aiuterà i progettisti di applicazioni a decidere quale modello di contesto e metodi di Context-aware dovrebbero essere utilizzati per avere un'applicazione efficiente di Context-aware. Inoltre, consente ai ricercatori di comprendere i requisiti essenziali dei sistemi Context-aware. Abbiamo discusso il background dei sistemi Context-aware e della gestione delle informazioni contestuali. Inoltre, abbiamo spiegato numerosi metodi di Context-aware e diverse sfide contestuali che devono essere prese in considerazione.

Chapter 1

Introduction

The evolution of Cyber-physical systems (CPS) has led to the spread of the concept of the Internet of Things (IoT), a new era for the information technology, where technology can be embedded at any "objects" to take an active role in the digital world, generating and exchanging messages through a distributed network [20].

Context-awareness is a main feature of ubiquitous computing systems that exist and has been employed since the early 1990s. Context-awareness can be considered a methodology for application development to provide relevant services to the users based on the user's needs at the correct time and location. Context-aware systems are famous because of being flexible, adaptive, and capable of behaving automatically on behalf of users. The term context as one of the context-aware systems' parameters is critical since it provides information about an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application [29].

Moreover, contextual information needs to be managed by a context information management system since the data collected by sensors are raw and potentially considered untrustworthy. Therefore the presence of a system to manage these data and finally delivering appropriate information based on the user's need is unavoidable.

This chapter provides the overview of this document. In section 1.1, we explain our objectives which leads to creation of this document. In sections

1.3 and 1.4 we present main contributions and structure of the document respectively.

1.1 Objective

Although there are plenty of published papers in the last two decades, our main objective is to carry out state-of-the-art context and context-aware systems. Therefore it can be helpful for newcomers to understand the context-aware systems' requirements to design suitable applications for users.

1.2 Motivation

This work is performed within the RIDE2RAIL (R2R)[1] project, in the frame of Innovation Programme 4 (IP4) of the Shift2Rail (S2R) initiative[2]. The aim of the S2R project is to establish a collaborative ecosystems through its Interoperability Framework [72] that offers many components such as data conversion approach [23], automated mapping [41], and ontology management [6]. The ecosystem assists the interoperability among all the IP4 services such as Booking, Issuing, Journey Planning, etc. and travel service providers (TSPs). Among the mentioned services, the so-called Travel Companion (TC) enables travelers to send mobility requests and receive travel solutions and enables drivers to share their rides with other travelers. In order to provide a personalized set of travel offers to the TC's users it is crucial to understand their context and preferences. The recommender core of the TC proposed in [43] takes as input user data, knowledge models, and service-related information and accordingly presents a list of travel offers ranked according to the user's contextual preferences. To enable the TC to capture users' preferences, Javadian et al. [71] designed a model of preferences using the context dimension tree methodology [16].

The aim of this work is to investigate the state-of-the-arts in designing context-aware systems in order to enhance the proposed preference models in [43] and [71].

It should be noted that the outcome of this work can be used by other

[1] ride2rail.eu
[2] shift2rail.org

researchers and developers aiming at building context-aware systems.

1.3 Contribution

Based on the studied papers, we have created two charts as below:

Figure 1.1 indicates the collected papers from numerous resources. By looking at the chart, we can understand the vast majority of papers found in "Sciencedirect " and "Researchgate".

Figure 1.2 illustrates the studied papers divided based on the published year. The graph shows, most of the studied papers are from "2015-2019".

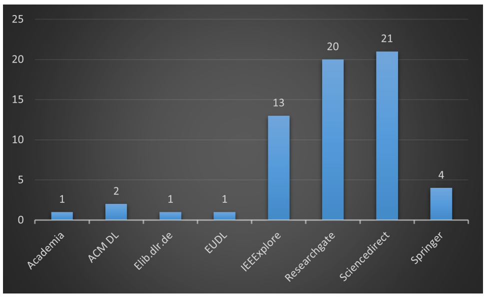

Figure 1.1: Resources of studied papers

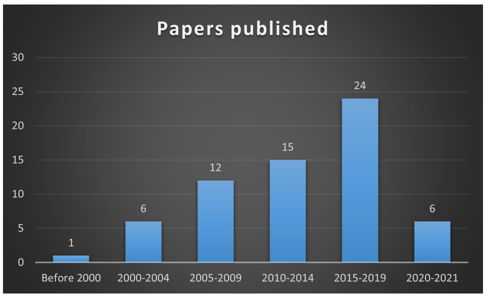

Figure 1.2: Number of studied papers

The main contributions of the document are listed below.

- Exploring the past surveys
- Discussing and analyzing various context and context-aware definitions proposed by papers.
- Analysis of context categorization and their utilization

- Analysis of context life-cycle and its four stages
- Providing an evaluation of the existing context-aware methods
- Presenting context challenges and existing solutions

1.4 Structure of the Document

The rest of the document is organized as follow. Chapter 2 discusses about the published surveys indicating their contributions, Chapter 3 discusses about the concept of context and context-aware systems by presenting different provided definitions. Chapter 4 discusses about different provided context categorizations. In the following, in Chapter 5, we discuss about context information management containing four main stages; context acquisition in section 5.1, context modeling in section 5.2, context reasoning in section 5.3, and context resource management in section 5.4. Chapter 6 discusses about context and context-aware methods. Chapter 7 discusses about context challenges. Finally, Chapter 8 provides the conclusion and future works.

Chapter 2

Surveys on Context-awareness

There are numerous surveys published in terms of context and context-aware systems. Studying past surveys helps us understand to what extent researchers worked and figuring out the research gaps. This chapter will outline and present some of the past surveys concentrated on context and context-aware systems. Following this section, we will explain the *contributions of each survey*.

Hong et al. 2009 [93] is a literature review of context-aware systems from 2000 to 2007. The survey suggested a classification framework based on the architecture of context-aware systems. Studying this survey helps researchers to implement context-aware systems effectively. The main contributions of this review are:

- Classifying and summarizing context-aware systems researches
- The review is done based on the architecture of context-aware systems containing five layers: the concept and research layer, network infrastructure, middleware layer, application layer, and user infrastructure layer

The network layer here means supporting context-aware systems and context information collecting by sensors. The middleware layer manages processes and stores context information, the application layer provides appropriate services to the users, and the user infrastructure layer offers a suitable interface. In this review, the context-aware concept has the highest percentage of context-aware articles. However, the survey studied papers and articles that were published from the year 2000 to 2007 so, many valuable reports and articles have not been considered.

Pradeep and Krishnamoorthy 2019 [64] conducted a survey about context-aware systems from 2000 to 2018. The survey represents the requirements and components for having an effective context-aware system named MOM, an efficient context modeling, context organization, and context middleware. The main contributions of this survey are:

- Analysis of context-aware past surveys
- Presenting several context categories and types
- Ontology development tools and technique presentation
- Comparative analysis of context modeling techniques
- Comparative analysis of context ontological models

One of the advantages of studying this paper is to be a starting point for newcomers to understand how to design and develop an effective context-aware system.

Bettini et al. 2010 [11] is a state-of-the-art discussion about the set of context modeling and context reasoning requirements. The requirements were followed by a description of the most commonly used approaches to context modeling and reasoning. Understanding the requirements will help developers to design relevant context-aware applications. The paper also discussed two particular issues that should be addressed in context reasoning and context-aware methods: managing uncertainty in-context information and high-level context abstractions.

Capurso et al. 2018 [22] studied past surveys of context-awareness for mobile platforms focusing on localization, user's movement tracking, and environmental sensing. Localization means the user's location determination without using GPS in the best case applied in indoor and outdoor environments. Movement tracking focused on identifying common user activities such as driving-related, human body movement. Finally, environmental sensing means to detect the user's environment, including two parts: object detection and environmental condition sensing. The main contributions of this surveys are:

- Classifying different sensing methods
- Classifying different kinds of trade-offs
- Represent movement tracking researched papers
- Comparison classification of object detection and environmental condition works

The survey worked on localization, user movements, and surrounding environment categories within context-awareness mobile platforms. Also, different challenges in these areas have been discussed. However, privacy and the risk of information leakage must be tackled in the future.

Strang and Linnhoff-Popien 2004 [81] is a context modeling survey focusing on describing and reviewing different context modeling approaches and classifying them based on their relevancy and utilization. Moreover, context modeling requirements such as distributed composition, partial validation, richness and quality of information, incompleteness and ambiguity, level of formality, and applicability to existing environments are discussed.

Bolchini et al. 2007 [17] is a survey proposed a framework that comprehensively evaluates different context models helping designers decide which context model best suits their application. The survey considered the data tailoring challenge as a target of applications and reports the systems that use data tailoring application scenarios. The contributions of this survey are as follow:

- Providing a context definition
- Providing a high-level architecture of the context-aware system
- Introducing a framework for context modeling analysis
- Applying the proposed framework on some context models assuming data tailoring as a target application

Gollagi et al. 2020 [35] is a survey providing an inclusive analysis of data characteristics and reviewing the existing techniques in context-aware pervasive computing used for inferring situations from sensor data. The survey can be helpful in different domains such as health care, IoT, and cloud computing. The contributions of the survey are as follow:

- Providing solutions for the challenges like high energy cost, presence of noise, and large-scale datasets in the mentioned domains
- Analysis of existing techniques in context-aware pervasive computing, including their motivations, advantages, limitations, and performance measure

Fernandez-Rojas et al. 2019 [33] is a state-of-the-art discussion on contextual awareness in various human advanced-vehicle systems or HAVSs' technologies. Different tools and techniques used in many applications, such as radio communication and wireless networks, are presented. Moreover, the types and characteristics of context-aware systems have been discussed. The review structure divides into two main components: contextual awareness applicable to vehicles and contextual awareness applicable to humans. The main contributions of this survey are as follows:

- Propose a technical definition of context suitable for HAVSs
- Analysis of different types of context-aware systems such as Local, Centralized, and Distributed
- Analysis of context-awareness in-vehicle systems such as vehicular Adhoc networks, service models, wireless communications, cognitive radio, security, and data management
- Analysis of context-awareness for human-computer interaction such as privacy, recommender systems, human-machine interaction, and activity recognition

Anh et al. 2019 [8] is a survey of context-aware mobility in IoT. The paper represents some mobile data collection challenges, mobile data analysis, energy management, security and privacy, and IoT services delivery. However, this work has just been done in simulation mode by Matlab software and still needs to be more investigated.

Vahdat-Nejad et al. 2016 [86] is a survey that discusses classifying context-aware applications in vehicular networks. The paper proposed a classification framework based on three dimensions: environment, system-and-application, and context-awareness. The proposed framework is used to review and classified the existing context-aware applications in transportation systems from a technical perspective. The other contributions of the paper are:

- Represents context-aware projects in vehicular networks, including their description
- Representing the set of context information used by context-aware vehicular applications

Cabrera et al. 2017 [21] is a state-of-the-art context modeling contain a set of 138 papers. The survey performs a systematic mapping to accurately retrieve and analyze different context models by defining and performing a review protocol to provide the needed background to build or reuse context models. The study mainly focuses on the analysis of the state of the art of ontology-based context modeling due to their reasoning capabilities.

Machado et al. 2019 [55] is a systematic literature review aiming at identifying the state of the art of context-awareness, including the papers that use a hybrid context reasoning. This paper analyzed published papers between 2004 and 2018 and selected ten papers that worked only on hybrid context reasoning out of 3241 papers. The main contribution of this surveys are:

- Representing the comparison between different context reasoning approaches, specifying their advantages, limitations, and applicability
- Analysis of different context reasoning approaches
- Analysis of hybrid reasoning papers

Alegre et al. 2016 [4] is a state-of-the-art for context-aware systems development focusing on context-aware methodologies and their features, demonstrating context-aware engineering challenges during the development stages, and the conceptualization of context-aware systems. The paper is a literature review of 750 researchers discussing first the concept of context and context-aware systems. The paper claims that existing methodologies lack some features and performed a questionnaire to understand the main reasons for adopting these methodologies. The main contributions of this survey are:

- Analysing context information management includes its stages
- Analysis of context challenges
- Classification of design patterns used for context-aware systems development based on their applicability
- Requirement analysis of the context-aware methodologies

Gollagi et al. 2019 [36] is a state-of-the-art aim at formulating the context-aware modeling in two aspects: context data retrieval and context data modeling. The review contains 60 research proposing diverse applications that can offer better improvements. The analysis reviewed the different applications that used the concept of context-aware data modeling and retrieval. Also, research issues have been presented for the researchers to accomplish extent researches.

Perera et al. 2014 [63] is a survey of context-awareness from the IoT perspective view. The survey discusses first the background of IoT and context-aware fundamentals; then context life cycle is analyzed. The work contains a set of 50 projects from 2001 to 2011, providing their taxonomy. The survey addresses various techniques, methods, models, functionalities, systems, applications, and middleware solutions in terms of context-awareness in IoT. The main contributions of this surveys are:

- Analysis of context life cycle including four main stages
- Comparison analysis of context modeling approaches
- Analysis of context reasoning approaches
- Evaluation of context-aware methods
- Context challenges analysis

Villegas and Müller 2010 [90] is a survey about identifying context modeling approaches and management requirements to optimize intelligent interactions and services. The survey discusses context-aware systems challenges by asking questions like How to acquire, combine and distribute context information to several targets efficiently, manage the dynamic nature of context information, and uncertainty the context-aware system's requirements. The main contributions of this paper are:

- Providing an operational definition for context-aware management in the smart internet and its classification to distinguish the context types
- Providing a feature-based framework for specifying context modeling and evaluating existing approaches

Investigating the mentioned surveys provided insights into the relevant state-of-the-art context-aware systems. As a result, we explored the studies provided by each of the surveys and expanded our search to new publications to provide a new categorization for the state-of-the-arts in context-aware systems.

The results of our work are presented in Table 2.1.

The table indicates the contributions of our work divided into four columns. The first column named "A" is related to all the papers that proposed a definition regarding context-aware systems. These definitions will be discussed in detail in Chapter 3. The second column named "B" shows the papers that discussed context information management and proposed a context modeling or reasoning method. Context information management will be discussed in Chapter 5. The third column named "C" illustrates the papers that proposed a specific context and context-aware method. Chapter 6 will discuss in detail about context-aware methods. The last column named "D" indicates the papers proposed a categorization of context-aware systems, which will be discussed in Chapter 4.

	A	B	C	D
Schilit and Theimer [74]	X			
Abowd et al. [1], Dey [31]	X		X	
Zimmermann et al. [94]	X			X
Villegas and Müller [90]	X			
Bolchini et al [17]	X			
Miraoui and Tadj [56]	X			X
Perera et al [63]		X		
Bardram [10]			X	
Cassani et al [24]			X	
Lee and Lien [51]			X	
Perera et al [62]			X	
Schreiber and Panigati [75]			X	
Orsi et al [59]			X	
Grim et al [37]			X	
Otebolaku and Lee [60]			X	
Lijuan et al [53]			X	
Moradi et al [57]			X	
Pan et al [61]			X	
Cianciulli et al [26]			X	
Rosenberger et al [70]			X	
Feng et al [32]			X	
Ranganathan and Campbell [66]			X	
Schreiber and Valcher [76]			X	
Wang et al [91]			X	
Aguilar et al [3]			X	
Van Engelenburg et al [88]	X		X	
Choi and Lee [25]			X	
Anderson et al [7]			X	
Janssens et al [42]			X	
Tang et al [84]			X	
Koc et al [48]			X	
Pradeep et al [65]			X	
Bolchini et al [14]			X	
Abowd and Mynatt [2]				X
Dey and Abowd [30]				X
Van Bunningen et al [87]				X
Recio-García et al [67]			X	
Nalepa and Bobek [58]		X		
TalebiFard and Leung [83]			X	

Table 2.1: The table shows the contribution of the studied paper divided into four columns. The columns description are as follows: **A** indicates Context and Context-aware Definition, **B** Context Information Management, **C** indicates Context and Context-aware Methods, and **D** indicates Context Categorization.

Chapter 3

The Concept of Context and Context-aware Systems

A context involves several pairs of context types and context values. Specifically, a context type describes a characteristic of the context. A context type is associated with a specific context value which may vary over time with the changing environment. For example, the context types for an end-user could include location, identity, and time. The context type "location" may be assigned with a value, such as "New York". A context scenario combines different context types with specific context values to reflect a user's situation. In general, the word context refers to the set of information surrounding an activity and some supplementary information about the environment of it [56]. In computing domain, context is an old word used in particular in operating systems theory to characterize the minimum set of information required about a task in execution (process) [56]. Therefore, from the early years of context-aware computing, researchers recognized the benefits of utilizing context to aid users and systems. A context model is used to specify the relations and storage structure of various context types and values. A context-aware system is designed to adapt systems' operations to an end user's context without direct intervention from the user. A context-aware system generally consists of two parts: (1) sensing and managing contexts; and (2) adapting the system to the changing contexts. Context sensors are used to retrieve context values from different resources. The Global Positioning System (GPS) receiver is an example of a context sensor to detect location. The most common approach for context-aware systems is to manually construct mapping rules between the context scenarios and the corresponding reactions to adapt the behaviors based on different context scenarios. For example, IF-THEN rules are commonly used as mapping rules. The IF sentence describes the context scenario, and the THEN sentence represents the corresponding reactions. For example, if a person is in a meeting and receiving a phone call, then his/her cell phone is automatically switched from the ringing mode to silence or vibrate mode. Moreover, there are various definitions of context and context-aware systems among different domains. What is considered to be a context in a system might not be important in another system. This chapter discusses the concept of context and context-aware systems by mentioning different proposed definitions by authors. Understanding of these concepts is important as they will help application designers to decide what context should be used and which context-awareness behavior should be considered to deliver a relevant application [29]. Following we will discuss about proposed definitions based on the table 2.1 column A indicating the papers that proposed a new definition of context and context-aware systems.

One of the first definitions introduced about context-aware computing has given by Schilit and Theimer 1994 [74] explaining that *"Context-aware computing is the ability of a mobile user's applications to discover and react to changes in the environment they are situated"*. This definition refers to Context as a location that is too specific and difficult to be applied [29].

The most widely used definitions about Context and context-aware systems provided by Abowd et al. 1999 [1] and Dey 2001 [29] claimed that 'context' could not be identified using Schilit's definition, and stated that *"Context is any information that can be used to characterize the situation of an entity, where an entity can be a person, place, or physical or computational object"* and *"A system is context-aware if it uses context to provide relevant information and services to the user, where relevancy depends on the user's task"*. Dey 2001 [29] discussed the ways that Context can be used in context-aware systems. For example, the entities for an indoor mobile tour guide are the users, the application, and the tour site. By considering two pieces of information like weather and other people's presence, we can determine which one can be remarked as Context based on the provided definition. The weather does not affect the application because it is being used indoors. Therefore, it is not a context. Instead, the presence of other people can be used to characterize the user's situation. If a user is traveling with other people, then the sites they visit may be of particular interest to them. Hence, other people in Context can characterize the user's situation [29].

To further constrain the generality of Context, an operational definition of Context provided by Zimmermann et al. 2007 [94] claiming that Dey's given definition is general, as mentioned "any information" and "characterize the situation" in his definition. Then the definition will be extended by saying that *"Context is any information that can be used to characterize the situation of an entity [29]. Elements for the description of this context information fall into five categories: individuality, activity, location, time, and relations". The activity predominantly determines the relevancy of context elements in specific situations. The location and time primarily drive the creation of relations between entities and enable the exchange of context information among entities.* The operational definition is more actual and implementable than an abstract and comprehensive definition [90].

Another operational definition about Context that is suitable for managing context-awareness in the smart Internet is Villegas and Müller 2010 [90] saying that *"Context is any information useful to characterize the state of individual entities and the relationships among them. An entity is any subject that can affect the system's behavior and its interaction with the user. This context information must be modeled in such a way that it can be preprocessed after its acquisition from the environment, classified according to the corresponding domain, handled to be provisioned based on the system's requirements, and maintained to support its dynamic evolution".*

The three important aspects that make the operational definition of Context are: "The information which characterizes the situation of relevant subjects for the interactions between users and systems, the models required to represent such information, and the management and change of the information along its life cycle" [90].

Following that, another definition of Context is given by Bolchini et al. 2007 [17] saying that *"A context is not just like a profile, but as an active process dealing with the way humans weave their experience within their whole environment, to give it meaning".* Bolchini et al. 2009 [16], mentioned that *"The context is perceived as a set of variables that may be of interest* *for an agent and that influence its actions".*

Miraoui and Tadj 2007 [56] proposed a service-oriented definition of context. The paper claims that previously proposed definitions for Context are almost similar, and most of them referred to the user's location and the environment. Enumerating examples of Context like location, identity, etc. is not practical in all situations because some aspects are important in an application but not in others. Secondly, The definitions introduced in a high-level abstraction [56]. Hence the paper defined Context as *"Any information that trigger a service or change the quality of a service if its value changes"* and introduced context-awareness as *"A system is said to be context-aware if it can change automatically the quality of its services or provide a service as a response to the change in the value of an information or set of information that characterize those services".*

Almazan 2010 [5] introduced a definition of Context by saying that *"Context consists of one or more relationships an information item has to another information item. An information item can be any entity, either physical (such as a person, a computer, an object), virtual (such as a computer service, a group of people, a message), or a concept (location, time, and so on). A relationship describes a predicate connecting two or more information items, which may change at any time for any reason"*. However, this is a very general definition given.

Van Engelenburg et al. 2019 [89] introduced a new definition of Context intended to be applied in various application domains. It is based on the definitions of other concepts such as 'focus', 'situations', 'context elements', and 'context relationships'. The paper provided some new definitions that can be used to develop a criterion for deciding what belongs to Context.

Chapter 4

Context Categorizations

In this chapter, we will represent the various existing classification of the context provided by papers. We will also discuss their motivation and utilization. Pradeep and Krishnamoorthy 2019 [64] surveyed fifteen categorizations provided by papers from 1994 to 2013. A categorization of context types will help application designers uncover the most likely pieces of valuable context in their applications.

Zimmermann et al. 2007 [94] proposed a categorization for context-aware systems. According to the classification of context, information can be classified into five fundamental categories: individuality, time, location, activity, and relational (i.e., social, functional, and compositional relationships among the first four types). The vast portion of contexts falls into personal and environmental contexts. Most of the issues classified as personal context referred to user profiles and usually do not change during the application's operation. Environmental context includes attributes like time, location, and weather. These categories determine the design space of the context models which also by Villegas and Muller [90].

Individuality: Refers to the explanation of properties and attributes of an entity. Any contextual information collected from individual entities like users and items that may share common features is placed in this context. Individuality can be classified into entity types: natural, human, artificial, or groups of entities.

Natural context like weather information is without human intervention. The human context describes user behavior and preferences (e.g., user payment preferences). Artificial context like hardware and software configurations used in e-commerce platforms as human actions needed. Groups of entities refer to the concepts that share common features such as user preferences in the user's social network.

Activity: Refers to all the procedures that entity can be involved such as traveling, the task a user does at a particular time.

Location: Refers to the Spatio-temporal coordinates of the entity. For example, the city where a user lives is placed in a location context. There is also a physical location context information like the location of a movie theater. virtual location context like the IP address of a computer that is located within a network.

Time: Corresponds to information such as time of the day, current time, day of the week, and season of the year. Time context can be specified as definite and indefinite. Definite context demonstrates the beginning and ending period. Indefinite context means that we cannot define the time, for example, recurrent events that occur while another situation occurs, e.g., a user's session in an e-commerce application.

Relations: Refers to any relationship that an entity can have with another entity. It can be defined as social (i.e., interpersonal relations such as associations or affiliations), functional (i.e., the usage than an entity makes of another).

Abowd and Mynatt 2000 [2] proposed a context categorization named five W's of context include Who, What, Where, When, and Why.

Who: Referred to user interaction identity in the systems and will not incorporate with other people in the environment.

What: Referred to the job user is doing otherwise, leave the question. Understanding human activity is difficult, so context-driven devices need to incorporate human activity interpretations to provide useful information.

Where: Referred generally learn from a history of movements in the physical world. The "where" component of context has been explored more than the others.

When: Most context-driven applications are unaware of the passage of time without considering the time of the person being in a location. A particular interest is understanding relative changes in time as an aid for interpreting human activity.

Why: Referred to the reason why the user is doing a specific job. Sensing other forms of contextual information that could provide the user's state, such as body temperature, heart rate, can be useful.

Dey and Abowd 2000 [30] proposed a categorization of context divided into two parts: Primary and Secondary. Location, identity, time, and activity are considered primary context types to determine an entity's situation. These context types will cover all the 5W questions as well as more contextual information. The secondary pieces of context have typical specifications that can be derived by immediate context because they are the entity's attributes with primary context.

Van Buningen et al. 2005 [87] provided an operational and conceptual categorization based on context acquisition. Functional categorization referred to the way context information is acquired, modeled, and treated. Conceptual categorization refers to specifying user's related context from the environmental at a conceptual level.

Bellow is the Figure 4.1 representing one of the context categorization provided

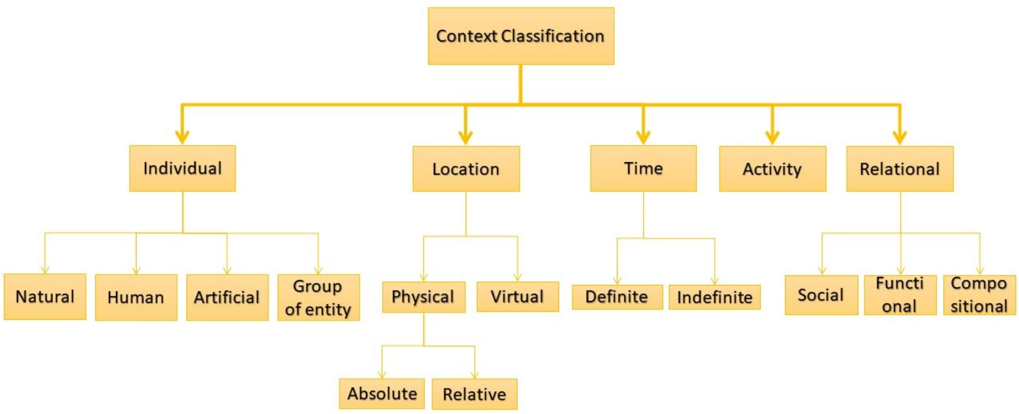

Figure 4.1: Context categorization

Chapter 5

Context Information Management

This chapter discusses the context-aware systems life-cycle. Context life-cycle explains the movement of data from one stage to another in application and middleware. In other words, it describes where the data is generated and how the data will be distributed [63]. Therefore Context management system contains four main stages: Context Acquisition, Context Modeling, Context Reasoning, and Context-aware Resource Management. We will describe Context Acquisition. After that, we will intensely discuss different context modeling and reasoning approaches indicating their characteristics, advantages, and limitations. Finally, Context-aware Resource Managements will be explained. Figure 5.1 indicates context-aware system management stages [3, 4].

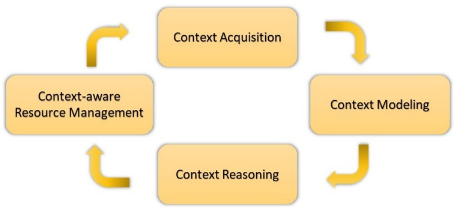

Figure 5.1: Context-aware life-cycle starting from Context Acquisition, Context Modeling, Context Reasoning, and Context-aware Resource Management

5.1 Context Acquisition

Context acquisition refers to sensing the immediate context from the environment. Context management infrastructures must implement mechanisms to generate numeric observables from physical sensors and determine comparable measures by performing fundamental transformations. Context is gathered from heterogeneous and distributed sources using different mechanisms such as pulling and pushing. The raw context must be pre-processed and classified before being manipulated at higher levels of abstraction. As raw data is heterogeneous, mass, and ambiguous, Context acquisition focuses on dealing with uncertainty and gain helpful information. The techniques used to acquire context can be different based on the responsibility, frequency, context source, sensor type, and acquisition process [3]. Therefore, context will be acquired from various sources and performed through sensors in three types: physical sensors, virtual sensors, and logical sensors [63].

Physical sensors as the name implies, these are the hardware sensors built-in to most modern smartphones, for example, the accelerometer, gyroscope, and temperature sensors. Physical sensors are tangible devices, digital or analog outputs.

Virtual sensors provide contextual data through a user's software. For example, what apps they might be using. Virtual sensors retrieve data from multiple sources and publish it as sensor data.

Logical sensors are the combination of physical and virtual. For example, a low-energy way of determining a user's location indoors is by performing a location lookup on nearby Wi-Fi access points. Logical sensors Merge physical and virtual sensors in order to produce more information.

5.2 Context Modeling

To provide a mechanism to express, store, and possibly evaluate context, designers and developers of systems create models representing the context itself. Context modeling is an essential component of the context information life cycle. Most importantly, once raw context data is pre-processed in the acquisition stage, it acquires the context model's form. Context modeling is the stage that tries to model the collected or sensed raw data and represent them according to a meaningful manner [63]. Context models are also required to provide higher-level context information, for instance, to represent the quality of context data. As the situation can be dynamic, we cannot predict what piece of information will be relevant at that moment; hence context modeling must be general [64]. Context modeling should be addressed independently from specific objectives [19]. There is no problem for all kinds of context modeling to be used independently in a specific domain. However, there is still a challenge to handle all kinds of context in a unified and general context modeling in today's technology conditions [92]. Based on the actual situation, the unified context modeling can be divided into two levels: Different contexts use the same data structure to express models, such as Key-Value Models, Markup Scheme Models. It also can support semantic unity, such as ontology Based Models, Graphical Models, etc. [91, 92]. The approaches are surveyed in [9, 11, 63, 64, 77, 81]. We will use them to highlight the features, advantages, and limitations of context modeling approaches. Figure 5.2 shows different context modelling approaches.

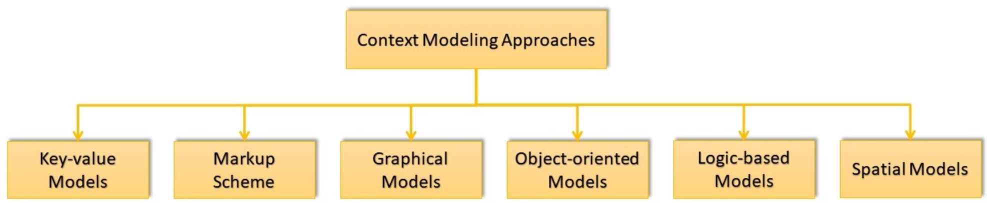

Figure 5.2: Context Models

5.2.1 Key-value Pairs:

Characteristics:

- Used to represent context information as a list of attributes with their corresponding values
- Used in distributed service frameworks (e.g., Capeus [73])
- They are frequently used in various service frameworks, where the key-value pairs are used to describe the capabilities of a service

Advantages:

- Providing the value of context information (e.g., location information) to an application as an environment variable where the 'key' is environmental data and the 'value' in the context state. This can be a simple mapping from sensor data to context
- Key-value pairs are also intuitive when looking for specific patterns within sensed data
- The key-value technique is an application-oriented and application bounded technique that suits the purpose of temporary storage, application preferences, and configuration

Limitations:

- Lack of structuring capabilities for efficient context information retrievals and there is no standard tool for processing
- Hierarchical structures or relationships cannot be modeled using key-value pairs
- Not relevant to store complex data structures
- Capturing a variety of context types
- Support of reasoning on context, on context uncertainty, and on higher context abstractions

5.2.2 Markup Scheme:

Characteristics:

- Is a hierarchical data structure consisting of markup tags with attributes and content
- Profiles represent typical markup-scheme models such as User Agent Profile and Composite Capabilities/Preference Profile
- It models data using tags so any language or mechanism (e.g. JSON, XML) that supports tag based storage allows markup scheme modelling

Advantages:

- Facilitates storage and allows efficient data retrieval

- Validation is supported through schema definitions
- Sophisticated validation tools are available for popular markup techniques such as XML
- Markup schemas such as XML are widely used in almost all application domains to store data temporarily, transfer data among applications, and transfer data among application components

Limitations:

- The limitations of this model are the same as Key-value model

5.2.3 Graphical Models:

Characteristics:

- It models context with relationships.
- Unified Modelling Language (UML) and Object Role Modelling (ORM) are the example of this model
- It could be represented in SQL database, noSQL database, XML, etc
- Can be used as persistent storage of context

Advantages:

- Graphical modelling is better than markup and key-value modelling as it allows relationships to be captured into the context model
- This kind of approach is particularly applicable to derive an ER-model which is very useful as a structuring instrument for a relational database in an information system-based context management architecture

Limitations:

- Limitations on data retrieval mechanisms such as SQL.
- Sophisticated context retrieval requirements may demand very complex SQL queries to be employed
- Adding context information and changing the data structure is also difficult in later stages

5.2.4 Object-oriented Models:

Characteristics:

- Are used to model data using class hierarchies and relationships.
- As most of the high-level programming languages support object-oriented concepts, modelling can be integrated into context-aware systems easily
- The context is modeled as an abstraction level on top of the available cues. Thus, the cues are objects providing contextual information through their interfaces, hiding the details of determining the output values

Advantages:

- Object oriented modelling is suitable to be used as an internal, non-shared, code-based, run-time context modelling, manipulation, and storage mechanism
- The main benefits of any object oriented approach - namely encapsulation, reusability, and inheritance can be used

Limitations:

- Does not provide reasoning capabilities
- Validation of object oriented designs is also difficult due to the lack of standards and specifications

5.2.5 Logic-based Models:

Characteristics:

- Facts, expressions, and rules are used to represent information about the context, and rules are used by other modeling techniques, such as ontologies
- Logic-based modeling allows new high-level context information to be extracted using low-level context.
- It can be used as a supplement for other context modeling approaches
- High degree of formality is a common feature to all logic-based models

- The goal is to form new expressions or facts from previous ones
- Logic-based and ontology-based models define formal specifications of context entities and their relations in a particular domain

Advantages:

- It provides much more expressive richness compared to the other models discussed previously
- Reasoning is possible up to a certain level, and based on existing rules in the systems, reasoning can be used to derive new facts
- Even non-technical users can add rules and logic to the systems during run time

Limitations:

- Lack of standardization reduces the re-usability and applicability

5.2.6 Ontology-based Models:

Characteristics:

- Represents knowledge, concepts, and relationships about a domain and describes specific situations in a domain
- An ontology expresses common concepts (e.g., people, travel, and weather) and the concept's relations
- Ontology is the main component in semantic technology that allows to model data
- Popular semantic web ontology languages are: Ontology Web Language (OWL) and Resource Description Framework (RDF)

Advantages:

- There are several reasons for developing context models based on ontology, such as "Knowledge Sharing", "Logic Inference", "Knowledge Reuse"
- A further benefit of ontologies concerning simpler representation formalisms consists in support of reasoning tasks

- Comparing simpler approaches (e.g., key-value and markup models), ontological models of context provide clear advantages in terms of heterogeneity and interoperability
- A wide range of development tools and reasoning engines are also available
- Ontologies are popular in managing and modeling context despite their weaknesses
- Ontology reasoning techniques can be applied easily

Limitations:

- Context retrieval can be computationally intensive and time-consuming when the amount of data is increased
- Limited in modeling context uncertainty

5.2.7 Spatial context model:

Characteristics:

- Space is a most important context in many context-aware applications
- They are fact-based models that organize their context information by physical location and context is defined as where you are, who you are with, and what resources are nearby
- Places and location can be considered as spatial entities

Advantages:

- Spatial context models are well suited for context-aware applications that are mainly location-based, like many mobile-based applications
- Spatial context modeling is achieved with tiers of spatial ontologies: tier0 (ontology of the physical reality), tier1 (observations of reality), tier2 (observations are defined by uniform properties), tier3 (social reality and relations of all objects), and tier4 (rules are modeled) [34]

Limitations:

- Much effort needs to gather the context information's location data and keep it up to date

5.3 Context Reasoning

On top of the context representation model, appropriate reasoning mechanisms are required to exploit the available context information and add intelligence to the systems. Once the context modeling is deployed, the reasoning is required. It is defined in many papers as a method of deducing higher-level context from the low-level context to better understand the acquired sensed context because of the uncertainty and imperfection of context [63]. The reasoning is able to determine the appropriate system behavior when the context changes to provide relevant data to the user [12]. Context reasoning reduces context-aware application's complexity and improves their maintainability and evolvability. Context-aware applications use context information to evaluate whether there is a change to the user and computing environment context, deciding whether adaptation to that change is necessary, which often requires reasoning capabilities. Using context reasoning improves the quality of context information and inferring new kinds of context information. There are various reasoning approaches that we will explain some of the commonly used, including *Case-based, Rule-based, Fuzzy Logic, Ontological, and Probabilistic reasoning* [3, 4, 11, 63].

5.3.1 Case-based

Case-based reasoning is one of the supervised learning methods dealing with new problems by reusing solutions to past similar problems. Instead of applying classical rules to solve a problem, expertise applies a similar problem that was already solved [68, 82]. Case-based reasoning (CBR) helps to solve a problem with less effort. Also, case-based reasoning can be used in the recommender systems regarding user preferences, using prior user experiences. Context-aware case-based reasoning can develop a personalized recommendation by looking at the user's history or understanding the user's attempt to search and find a product [85]. For example, a user wants to travel to any destination. Using case-based reasoning will allow the system to provide relevant offers based on his/her past travels. However, if the user wants to act on behalf of someone else, then recommending context information based on the experience will not be helpful anymore. Another example is the situation where the case-based reasoning system recognizes the user is hungry and suggests some actions that bring the user into the restaurant to fit his/her preferences [49]. In this example, new cases contain the information collected, such as the user's preferences, the location, time, and any entities accessible in the environment. However, this method has some challenges, like acquiring the initial case and maintaining a high volume of case-based systems running in an online manner during case-based reasoning. Thirdly, the problem of many cases is essential to decide how it will be managed. Moreover, there are still more challenges with this technique that we leave that part to the readers as it will be beyond our work scope [49].

5.3.2 Rule-based

This approach tries to express limitations and preferences in an easily understandable format [64]. Rule-based is the most straightforward reasoning method and usually structured in IF-THEN-ELSE format [63]. This approach is one of the most popular existing methods to deduce high-level context information from the low-level context. This technique is primarily used in IoT systems as it is the simplest way of modeling human thinking and reasoning in machines [63]. Reaction rules represented in the form "Event Condition Action" to encode appropriate knowledge [82]. One of the most popular tools for context-aware applications that implements a rule-based approach is Context Toolkit [31]. Rule-based reasoning allows using more expressive rule languages, such as first order logic, logic programming, and defeasible Logic. However, the rule-based approach suffers from the lack of validation, and quality checking [12]. Below we mentioned more benefits of using the rule-based reasoning approach:

- This approach expresses policies, constraints, and preferences in an easily understandable format
- They are highly error-prone due to manual representation and lacks provision to handle ambiguous and imperfect contextual information
- Integrating Rule-based reasoning with an ontology-based application is useful for applications based on ambient intelligent domain

Mobile platforms require to be lightweight and straightforward, so the model must solve complex and straightforward context tasks. Hence, Nalepa and Bobek 2014 [58] proposed a new rule-based context reasoning platform suitable to the need of mobile computing devices. The model has a learning middleware supporting context acquisition.

5.3.3 Fuzzy Logic

Fuzzy logic suggests approximate reasoning instead of fixed and crisp reasoning. It is suitable for the imprecise notions such as tall, short, dark, trustworthy, and confident in reason values between 0 and 1 [64]. Fuzzy logic has been extended to handle the concept of partial truth [63]. However, this technique cannot be used in most cases as a stand-alone technique, and it should be implemented to complement another technique [63]. Gaia [69] is a context model that uses fuzzy logic to manage uncertainty. It also needs to be manually managed. Below we mentioned the features of the fuzzy logic reasoning approach [82].

- It offers approximate reasoning using a simple and natural representation
- They are highly error-prone due to the manual representation of logic and lacks validation and quality checking of contextual information
- Fuzzy logic provides an inference mechanism that enables reasoning capabilities to be applied to knowledge-based systems. In fuzzy logic, knowledge is the set of fuzzy rules

5.3.4 Ontology-based

Ontological reasoning can be used where a set of description logic rules are primarily used to derive implicit knowledge from the context knowledge base. Although they naturally integrate with the underlying ontology-based representation model, they offer limited expressive, and reasoning capabilities [12]. This approach integrates easily with ontology-based context modeling and has relatively low computational complexity so that they can easily deal with rapidly changing situations. Although they support complex representations and reasoning by providing meaningful information, environmental contexts could be ambiguous. There are different reasoning tools based on ontology reasoning like FaCT [40], Hermit [78], Pellet [80], and RacerPro [38]. Moreover, ontologies can describe knowledge of an application domain in terms of classes, relationships between entities, and instances of a class together with their individual properties [82].

5.3.5 Probabilistic logic

The characteristics of probabilistic logic reasoning is are listed below.

- This reasoning approach handles uncertainty by combining evidence from different sources
- Probabilistic reasoning can be performed only for numerical values with known probabilities
- The most popular techniques are Dempster Schafer Theory of Evidence and Hidden Markov Model (HMM)
- Probabilistic reasoning uses logic and probability to handle uncertain situations [82]
- Probabilistic reasoning is used to integrate ontology-based models with probabilistic models like Bayesian networks, which can explicitly model uncertainty in the context data, make confidence in the available sources and causal relationships between various contexts
- The rich, expressive capabilities constitute the main advantage of such approaches
- High complexity and the requirement to explicitly and precisely define uncertainty are their main limitations concerning their deployment in Ambient Intelligence systems

5.4 Context Resource Management

This section describes the last stage of context information management, where the contextual information will be delivered to the users. Context resource management uses some subscription/ publishing and event triggering mechanisms to distribute context throughout the system [44]. Both high-level and low-level context needs to be distributed to the consumers interested in context [63]. Context resource management is a reasonably simple task and provides methods to deliver context to the users [63]. There are two methods of managing context resources. 1) Query: which is based on the user's request the context. So the context management uses the query to provide a result. 2) Subscription, also called publication, which users need to subscribe to the content management system, would periodically return when an event occurs. This method is used more in real-time processing [3, 63]. TalebiFard and Leung [83] proposed a methodlogy based on publish-subscribe approach discussed in detail in chapter 6.

Chapter 6

Context and Context-aware Methods

To facilitate the inherent complexity of developing context-aware systems, we need to use an appropriate method. This chapter will discuss different existing context and context-aware methods without focusing on a particular domain.

Bardram 2005 [10] introduced Java context-awareness framework (JCAF), which is Java-based context-awareness infrastructure and programming API to create context-aware applications claiming to be simple and robust ability to be used in several applications areas like at the hospital. The core design principles, runtime infrastructure, and its programming model have presented in this paper.

Dey 2001 [29] created an architectural framework called Context Toolkit is a composition of features and abstractions to support context-aware application designers. This method makes context-aware applications resistant to changes in the context-sensing layer. It also discusses a new abstraction called situation abstraction, in which the collection of states can be described as a situation. The toolkit provides three abstractions: widgets, interpreters, and aggregators.

Cassani et al. 2017 [24] represents a methodology called Context-Aware Mobile mashUpS (CAMUS). This method will help developers design flexible mobile applications that can deal with heterogeneous data and filter the content to satisfy user's situational needs. The paper explains how to design and develop CAMUS based on the abstraction of context and mashup modeling. the paper proposes the Context Dimension Model (CDT) To characterize different contextual situations user's behavior,

Lee and Lien 2009 [51] is a paper about the location-based context-aware framework called Context Awareness Information Service (CAIS). The framework works only in a restricted local area, and people must be geographically close to each other. It uses general communication technologies like Bluetooth. It's a service that works on mobile devices in terms of information sharing.

Perera et al. 2012 [62] discusses a method called CA4IOT. In the IoT paradigm, the existing challenge is to select the appropriate sensors to receive relevant data. This architecture will automate selecting suitable sensors by applying reasoning, filtering, and fusion mechanisms. The paper claims that this model will focus on one problem rather than performing weakly on several domains. The paper's main objective is to allow users to register their problems, so the proposed methodology understands them and produces more meaningful information than the raw sensor data streams generated by individual sensors. Although there are middleware solutions in the literature for sensor data management related to IoT and pervasive computing, they are powerful while a week in other aspects. Hence this method only focused on one problem other projects can adopt with this method.

Schreiber and Panigati 2017 [75] propose a hybrid solution method named Perla for designing and managing Context with the help of Context Oriented Programming Java extension JCop used for writing complex procedures where each layer is bound to a specific context. Perla determines how the collected data should be processed and stored and specifies the behavior of the device.

Orsi et al. 2011 [59] proposed a context-aware method named SAFE based on the combination of keywords with natural language queries to reduce the required time to get an answer to the user's queries. The method performs ontology and query patterns for retrieving appropriate data. The method is suitable in application scenarios where query and data retrieval should be performed quickly. The paper also discusses the integration of CDT with SAFE shortly. As far as pure keyword-based solutions have two significant problems: Information transmitted by natural language is not within the structure of the sentences most of the time; thus, imprecise or ambiguous keywords will be retrieved; The user must know available information to input keywords. The situations that speed and precision are essential could be critical issues, for example, in emergencies. The proposed method is a lightweight approach to mobile search in such situations, based on semantic technologies and supported by context-awareness. For keyword semantics, matching ontologies are performed against query patterns already defined, which are prompted to the users for them to identify the most appropriate query. The experiment results show that SAFE improves the time to access the desired data in emergencies by exploiting context-awareness because it suggests a way for tailoring the data, and consequently query templates and searches, to the current Context.

Grim et al. 2012 [37] proposed Grapevine's method for context information sharing in a localized region of a network. The paper aims to represent both individual and groups of context information efficiently to be transmitted even with limited bandwidth connection. Grapevine efficiently shares Context in the immediate network and uses that shared information to create a view of a situation shared among groups of entities.

Otebolaku and Lee 2017 [60] discussed and proposed a framework in IoT, a context recognition model using Artificial Neural Network (ANN) aiming to derive trustworthy and precise high-level Context because sensing data from intelligent objects are unreliable for delivering personalized intelligent services. The proposed method provides essential functionality for context sensing, preprocessing, filtering, and classification.

Lijuan et al. 2013 [53] proposed two network selection algorithms on network decision making: fuzzy mathematics evaluation method and multiple attribute decision making with TOPSIS evaluation method. The network selection is performed based on contextual information gathering from context-aware services, including network information, user information, and local information. Quality of Service (QoS) and user preferences are considered, and other network metrics such as speed, coverage percentage, location, etc.

Moradi et al. 2020 [57] discuss a framework named Context as a Service Set of Engineering Tools (CaaSSET) to ease context services development. In order to reach it's goal, the CaaS domain should be specified as a metamodel. The framework consists of a CaaS reference model, a context service metamodel, a graphical modeling tool, a code generation tool, and a context service management tool. "The CaaS platform is a context management middlewaremiddleware that lies between the providers and consumers of context services and facilitates publishing, discovering, monitoring, and maintaining contextual data and information." [57]. The framework enables developers to design valid context service models graphically and transform models into executable context services.

Pan et al. 2019 [61] discuss a proposed method for context-aware intrusion detection, which is usually deployed in Building Automation and Control network. It's a runtime model for service interactions and functionality patterns by modeling the collected heterogeneous information from various resources into a context-aware data structure by utilizing a UML class diagram. This framework analyses anomaly behavior to detect anomalous events triggered by cyber-attacks or any functional failure. This method can accurately detect and classify different types of BACnet attacks and asset malfunctions with high detection rates.

Cianciulli et al. 2017 [26] proposed a distributed software method to improve public transport efficiency by many mobile devices incorporating beacon technology and clients for getting information such as bus position and estimated arrival times. The method supports users by providing buses position as well as relevant travel information based on statistics. This method will significantly affect transportation in a long time in terms of cost and urban mobility quality. The method can capture helpful information such as users who take and get off buses at given stations, the time series history of bus positions, and enable interesting extensive data analysis. The main advantages of using this method are no infrastructure costs for implementation and maintenance; the System has a self-learning mechanism when new routes are introduced, requiring less human interaction.

Rosenberger et al. 2020 [70] proposed a new method called CASAD matrix. CASAD is a method that facilitates the development of context-aware systems and reduces their complexity to a manageable level. This method is used for documentation, analyzing, and optimizing the contextual services and Context related systems.

Feng et al. 2017 [32] proposed a new context-aware method for the sake of intelligent transportation cyber-physical (IT-CPS). The paper also discusses the principle of IT-CPS. The method can significantly improve the efficiency of IT-CPS and can be divided into four parts: Data collection from sensors, Collected data will be organized as they are isolated and disorder, Third part is the fusion and process of the organized information, which is the core of the structure, the most intelligent part which uses storage, calculation, recognition. The last part is responsible for information output and plays the role of service; in other words, resource management to the user will be performed in this part. This analysis proves that the intelligent transportation CPS learns through context-aware technology, which integrates the environment with people, cars, and other terminals. Hence the user experience will be improved as well as system enhancement.

Ranganathan and Campbell 2003 [66] proposed a method based on first-order predicate calculus. The first-order model enables us to write complex rules involving Context, perform automating inductive and deductive reasoning on contextual information, Boolean operators and existential and universal quantifiers description of Context can be implemented. The paper claimed that using first-order logic is a very effective and powerful way of dealing with context change behavior.

Schreiber and Valcher 2020 [76] proposed a modeling method based on Boolean Control Network for evaluating properties of context-aware systems focusing on safety-critical applications. The method is trying to formally assess the reliability and safety properties of self-adapting safety-critical systems converted into computer algorithms. Boolean Control Network provides powerful tools typical of system theory that can be applied to linear analog systems and digital systems, whose properties are usually expressed by logical rules.

Wang et al. 2004 [91] proposed an ontology-based context model named CONON to deal with issues like formal context representation, knowledge sharing, and logic-based context reasoning. The design of the context model and logic-based context reasoning scheme are discussed. The proposed method's primary goals are capturing the general concept of essential Context and providing flexible extensibility to add specific concepts in different application domains in a hierarchical manner.

Aguilar et al. 2018 [3] proposed Meta ontology for the context modeling, called CAMeOnto, to capture primary Context to a higher level in a hierarchical manner. This method performs based on the principle of 5Ws context categorization: who, when, what, where, and why. The services of CARMiCLOC must use the provided ontology method, a reflective middleware for context-aware applications, to represent the Context, to reason about it, to be shared, among other things, without sophisticated tools and processes. The method has been tested in kinds of contexts like a smart classroom or a Surgery Room.

Van Engelenburg et al. 2019 [89] proposed a method to specify what elements of the environment should be counted as relevant Context and then base the design in a structured manner. The proposed method consists of three steps: A detailed understanding of the Context, determining what components are needed to sense and adapt to Context, and determining the rules for how the System should adapt in different situations. The method was applied in the container shipping domain to develop a context-aware system for business-to-government (B2G) information sharing. A context-aware tour guide and a context-aware B2G in-formation sharing system used as an example scenario.

Choi and Lee 2012 [25] proposed a method for requirements gathering of context-aware systems. The paper argued that the analysts have to figure out Context related information, document, and need to agree on Context and context-aware services. Hence, for the information collecting, the paper proposed a use-case technique, and it shows an example of a tour guide system. There are essential features for system developments such as the System's scope, system platforms, target users, and system intelligence level.

Anderson et al. 2015 [7] as building context-aware applications for smartphones can be challenging and time-consuming; the paper proposed a new ontology-based reasoning method to create context-aware applications. It uses micro-services, which is a mechanism to provide multiple services for collecting, processing, and representing context information.

Janssens et al. 2005 [42] introduce a method for implementing an adapted Markov Chain modeling heuristic in the Context of transportation. This method extracts knowledge from data to understand the travel pattern of individuals' decision-making for transport mode. The method uses two different ways to store sequential information. (sequence of activities) in 'codebooks,' referred to a place where the sequential information for activities and travel is stored and is introduced to reflect that the information kept here represents low- and high-order combinations of activities that typically sequentially occur in one particular diary.

Tang et al. 2018 [84] provided a method for developing a tensor-based context-aware to model the travel time estimation on different drivers, road networks, and different time slots, using sparse and large-scale GPS paths. The method has four major components: map matching, travel time tensor construction, context-aware feature extraction, and travel time tensor factorization. The method has been applied in a real case on the urban road network of Beijing, China. One of this method's features is to extract several categories of contextual information such as historical, geographical, and spatial-temporal Context.

Koc et al. 2020 [48] introduced a framework named Comprehensive Resilience Assessment Framework For Transportation systems (CRAFT) in urban areas, which is used to analyze transportation disruptions. The methodology focused on capturing and generating resilience insight in 3 levels.

Pradeep et al. 2021 [65] proposed a new method named Adaptive Ubiquitous Middleware for context-aware IoT (AUM-IoT) that users, data sources, smart objects, and services can interact with each other. The method considers the situational Context of the applications, devices, or people and the contexts of the network formed and accordingly adapts the ecosystem's behavior. This method's main contributions are: Able to consider different contexts to be adapted to the situation; a multi-agent acts as an integration point for applications, optimal service delivering by achieving a minimum average response time.

Bolchini et al. 2007 [14] discuss a proposed method allowing application designers to compare different context model features depends on their target. Data sources require not only to be integrated but also to be filtered and provide the relevant data portion. The user receives only the appropriate tailored set of data. Hence, this method has been proposed because of the lack of a uniform approach for context modeling. The method first analyzes and compares available context models built on a well set of features. After derivation of these features, the application designer needs to define which features are more relevant to the target application. In this case, the paper assumed data tailoring as a target application; then, the System will provide the most appropriate context information to the developer. Then respecting each feature, a classification of context models will be performed. The result is a structured view of the state of the art, enabling the designer to compare the various models.

Khabou et al. 2018 [46] aims at facilitating the development of context-awareness applications using context prediction and context change detection. This paper analyzed the implementation phase deeply, prediction, and context change detection approaches. The paper also shows a case study of the smart building as an example. Laine and Brahimi 2016 [39] discuss different types of context-aware frameworks such as Context Toolkit, Context Broker Architecture (CoBrA), Service-Oriented Context-Aware Middleware (SOCOM), etc. After that, some applications for context-awareness frameworks have been indicated. Daniele et al. 2007 [27] discuss a rule engine component being able to process context-aware applications behaviors expressed in Event-Control-Action Domain-specific Language (ECA-DL). This component is deployed by using Jess, which is a tool for rule-based systems development. The computation is accomplished by using the rule-based context modeling technique. Klimek 2017 [47] talks about human behavior in the environment. Although the paper discusses a different case study called mountain rescuers, it provides valuable information for any domain regarding its environmental behavior. It provides a pro-active decision supporting actor operating in an intelligent space by representing a unified approach comprising informal (use cases) and more formal (algorithm) to obtain a common framework through context-based reasoning. Another contribution is an innovative method of mapping and filtering the pervasive streams of data-sets into collecting individual and anonymized tourist activities located in a particular tourist destination. Behavior analysis in intelligent environments has been studied in this paper. Bolchini et al. 2007 [15] discuss context-ADDICT project aims to discover relevant data and tailor them based on the user's need to run on mobile devices. Bolchini et al. 2011 [18] discuss the Context-ADDICT project aiming at defining a complete framework where Context is modeled hierarchically in terms of observable parameters and concentrates on data-oriented perspective meaning to provide user correct information at the right moment. The paper profoundly discusses different aspects of context-ADDICT such as Data-Access Manager, Design-time Context Manager, etc.

Recio-García et al. 2014 [67] provided an open-source platform called COLIBRI used for developing systems based on case-based reasoning. In this paper, they provided a functional description of the platform, its capabilities, and tools. Several applications are using COLIBRI to build specialized systems. One of the COLIBRI platform's critical features is its support for different families of specialized CBR systems, such as textual, knowledge-intensive, data-intensive, Distributed. The COLIBRI platform comprises a framework JCOLIBRI that provides the components required to build a CBR system. More information regarding case-based reasoning can be found in Section 5.3.1.

TalebiFard and Leung 2013 [83] proposed a fuzzy MADM context-aware method based on publish-subscribe approach for the purpose of context-aware resource management . The method retrieves the value of a network node based on the contextual information derived from the node and contextually similar nodes. As the context of the user could be dynamic, this method is suitable where there are not enough knowledge about the user's history of preferences.

Chapter 7

Context Challenges

This chapter discusses different challenges in context modeling, reasoning, and context-aware systems that need to be considered for improving these systems. We will present the papers which introduced and discuss challenges and their solutions.

Contextual information is collected from different resources, such as a single sensor. Data in the environment is uncertain and dynamic; thus, managing uncertainty is one of the situation-awareness needs. These uncertain situations can be of high importance to specific applications, such as a health monitoring application that monitors changes in a patient's health situation.

Below is the table indicates the papers discussed about context challenges.

	Context Challenges	**Resolution**
Bolchini et al [16]	Data Tailoring	CDT Context Model
Haghighi et al [28]	Data Uncertainty	Fuzzy Situation inference (FSI)
Silva and Dantas [79]	Data Imperfection	QoC filtering
Jung et al [45]	Uncertain and untrustworth Data	Context model security-relevant
Lin et al [54]	Untrustworthy Data in heterogeneous networks	SDDL
Lachner et al [50]	Privacy Preservation	A method based on context-aware
Li et al [52]	Privacy Preservation	hierarchical game framework to model
Bobek and Nalepa [13]	Data Uncertainty	unknown

Table 7.1: Context challenges

Haghighi et al. 2008 [28] proposed and validated a technique, namely Fuzzy Situation Inference (FSI), to deal with uncertainty. The method can represent uncertain situations and apply appropriate delta changes of context in the situation inference results. The FSI model uses fuzzy logic principles into the Context Space (CS) model, a reasoning and context modeling approach in the environment.

As the quality of data is highly influenced by how it is gathered; and most sensors feature an inherent inaccuracy. Hence, insufficient context data may prevent the proper functioning of context-sensitive applications. Silva and Dantas 2013 [79] proposed a QoC filtering approach that aims at sending only relevant data to the context server, Also indicating QoC policies based on the QoC parameters and application requirements. The result of using this method shows a significant reduction in sending data.

Jung et al. 2015 [45] proposed a context modeling method to deal with the uncertainty and untrustworthy problem of raw data collected by sensors. The model helps to improve context detection by explicitly reflecting on security and context information relevancy. The method can identify security-relevant context information and reliably detect these contexts by quantifying the overall trustworthiness of the context description during runtime, i.e., its vulnerability to forgery.

Detecting network intrusions to collect security-related data is the first step. However, due to the heterogeneity of networks and the ever-growing amount of data in today's world, there are some challenges in which traditional data collection methods cannot be applied, especially for security-related data. Lin et al. 2018 [54] presents the design and implementation of an adaptive security-related data collector based on network context in heterogeneous networks. The method can solve network heterogeneity by Security-related Data Description Language (SDDL) based on XML. It also uses an adaptive pattern to reduce the amount of collected data.

Privacy is one of the challenges with systems dealing with sensitive data. The lack of presence of a system involving privacy preservation prevents the acceptance of such systems. Edge computing copes with this concern by recognizing the key enabler for privacy-enhanced applications. Thus Launcher et al. 2019 [50] propose a method based on context-aware edge computing. The method aims at achieving privacy by deciding on how to process and manage context data. The paper applied this method in the eHealth domain as an example scenario. The method comprises three parts: first is how to define context-aware privacy policies for a particular system and how to describe them using the proposed model. Second, define a mechanism to gain more policies, and the third part describes the mechanism that determines how to perform on identified policies.

Li et al. 2018 [52] provided a privacy-preserving data collection scheme by incorporating complex interaction among users, attackers, and service providers into a three-antithetic-party game. The system model takes into account these three players. This method aims to maximize the service provider's utility while preserving the user's data privacy. This experiment shows that it allows the service provider to set a service policy and decide on realizing data.

Uncertainty of acquired data by physical sensors like GPS or virtual sensors like activity recognition sensors provided by the Google API is inevitable. Therefore it is necessary to model and process these data. For dealing with uncertainty problems, one of the challenges of mobile context-aware systems is Bobek and Nalepa 2017 [13] proposed three complementary methods by combining modified certainty factors algebra, probabilistic interpretation of the rule-based model, and time-parametrized operators into a toolkit for modeling and building robust mobile context-aware systems. The paper provided a hybrid approach that combines rule-based solutions and machine learning methods that provide a robust and efficient modeling method for handling uncertainties in mobile context-aware systems.

Reducing the size of data using contextual preferences is one of the challenges with context-aware systems. The aim of using context-aware systems is to integrate available data and filter the relevant data to provide the user an appropriately tailored set of data and manage the amount of data to improve query processing efficiency. Bolchini et al. 2009 [16] proposed a context model named Context Dimension Tree (CDT) that is suitable for tailoring the target data based on the user needs.

Chapter 8

Conclusions

These days the presence of Context-aware systems in pervasive computing is unavoidable. Context-aware systems concentrate on acting on behalf of the user and react to the dynamicity of an entity's situation to reduce the human-computer interaction.

In this survey, we presented the research directions within context-aware systems, and the main contributions are:

1. Reviewing related past surveys to understand the research gaps
2. Analyzing the background of context and context-aware systems by demonstrating various context-aware definitions provided
3. Representing different context categorization introduced
4. Comparative analysis of context-aware life-cycle by explaining how data will be collected and how it will be managed and delivered
5. Analysis of different context-aware methods
6. Representing different context challenges that need to be solved by specific method

This survey will help researchers understand requirements of context-aware systems and application designers to figure out which context method or which context model they need to use to deliver an appropriate and efficient application. This work can be a good starting point for those who want to research in this area and to be able to work on context-aware recommender systems (CARS), however many papers and articles remained to be investigated.

Bibliography

[1] Abowd, Gregory Abowd, Anind Dey, Anind K, Brown, Peter J, Davies, Nigel, Smith, Mark Sandler, Steggles, and Pete. Towards a better understanding of context and context-awareness. 01 1999.

[2] Gregory D. Abowd and Elizabeth D. Mynatt. Charting past, present, and future research in ubiquitous computing. *ACM Trans. Comput.-Hum. Interact.*, 7(1):29–58, March 2000.

[3] Jose Aguilar, Marxjhony Jerez, and Taniana Rodríguez. Cameonto: Context awareness meta ontology modeling. *Applied Computing and Informatics*, 14(2):202 – 213, 2018.

[4] Unai Alegre, Juan Carlos Augusto, and Tony Clark. Engineering context-aware systems and applications: A survey. *Journal of Systems and Software*, 117:55 – 83, 2016.

[5] Christian Almazan. Rover: Architectural support for exposing and using context. 01 2010.

[6] Ahmad Alobaid, Daniel Garijo, María Poveda-Villalón, Idafen Santana-Perez, Alba Fernández-Izquierdo, and Oscar Corcho. Automating ontology engineering support activities with ontoology. *Journal of Web Semantics*, 57:100472, 2019.

[7] Christoph Anderson, Isabel Suarez, Yaqian Xu, and Klaus David. An ontology-based reasoning framework for context-aware applications. In Henning Christiansen, Isidora Stojanovic, and George A. Papadopoulos, editors, *Modeling and Using Context*, pages 471–476, Cham, 2015. Springer International Publishing.

[8] Vu Tuan Anh, Pham Quoc Cuong, and Phan Cong Vinh. Context-aware mobility in internet of thing: A survey. *EAI Endorsed Transactions on Context-aware Systems and Applications*, 6(16), 3 2019.

[9] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. A survey on context-aware systems. *Information Systems*, 2, 01 2007.

[10] Jakob Bardram. The java context awareness framework (jcaf) – a service infrastructure and programming framework for context-aware applications. volume 3468, pages 98–115, 05 2005.

[11] Claudio Bettini, Oliver Brdiczka, Karen Henricksen, Jadwiga Indulska, Daniela Nicklas, Anand Ranganathan, and Daniele Riboni. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, 6(2):161 – 180, 2010. Context Modelling, Reasoning and Management.

[12] Antonis Bikakis and Grigoris Antoniou. Rule-based contextual reasoning in ambient intelligence. In Mike Dean, John Hall, Antonino Rotolo, and Said Tabet, editors, *Semantic Web Rules*, pages 74–88, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[13] Szymon Bobek and Grzegorz J. Nalepa. Uncertainty handling in rule-based mobile context-aware systems. *Pervasive and Mobile Computing*, 39:159–179, 2017.

[14] Cristiana Bolchini, Carlo Curino, Giorgio Orsi, E. Quintarelli, Fabio Schreiber, and Letizia Tanca. Cadd: A tool for context modeling and data tailoring. pages 221 – 223, 06 2007.

[15] Cristiana Bolchini, Carlo Curino, Giorgio Orsi, Elisa Quintarelli, Rosalba Rossato, Fabio Schreiber, and Letizia Tanca. Context-aware views for mobile users. pages 221–223, 06 2007.

[16] Cristiana Bolchini, Carlo Curino, Giorgio Orsi, Elisa Quintarelli, Rosalba Rossato, Fabio Schreiber, and Letizia Tanca. And what can context do for data? *Communications of the ACM*, 52:136–140, 11 2009.

[17] Cristiana Bolchini, Carlo A Curino, Elisa Quintarelli, Fabio A Schreiber, and Letizia Tanca. A data-oriented survey of context models. *ACM Sigmod Record*, 36(4):19–26, 2007.

[18] Cristiana Bolchini, Giorgio Orsi, Elisa Quintarelli, Fabio Schreiber, and Letizia Tanca. Context modeling and context awareness: steps forward in the context-addict project. *IEEE Data Eng. Bull.*, 34:47–54, 01 2011.

[19] Cristiana Bolchini, Giorgio Orsi, Elisa Quintarelli, Fabio A. Schreiber, and Letizia Tanca. Context modeling and context awareness: steps forward in the contextaddict project. *IEEE Data Engineering Bulletin*, 34(2):47–54, 2011.

[20] M Brambilla, A Javadian Sabet, and A Masciadri. Data-driven user profiling for smart ecosystems. In *Smart Living between Cultures and Practices. A design oriented perspective - ISBN: 978-88-7461-496-7*, pages 84–98. Mandragora, 2019.

[21] Oscar Cabrera, Xavier Franch, and Jordi Marco. Ontology-based context modeling in service-oriented computing: A systematic mapping. *Data Knowledge Engineering*, 110:24–53, 2017.

[22] Nicholas Capurso, Bo Mei, Tianyi Song, Xiuzhen Cheng, and Jiguo Yu. A survey on key fields of context awareness for mobile devices. *Journal of Network and Computer Applications*, 118:44 – 60, 2018.

[23] Alessio Carenini, Ugo Dell'Arciprete, Stefanos Gogos, Mohammad Mehdi Pourhashem Kallehbasti, Matteo Rossi, and Riccardo Santoro. ST4RT – semantic transformations for rail transportation. In *TRA 2018*, pages 1–10, 2018.

[24] Valerio Cassani, Stefano Gianelli, Maristella Matera, Riccardo Medana, Elisa Quintarelli, Letizia Tanca, and Vittorio Zaccaria. On the role of context in the design of mobile mashups. volume 696, pages 108–128, 01 2017.

[25] Jongmyung Choi and Youngho Lee. Use-case driven requirements analysis for context-aware systems. In Tai-hoon Kim, Jeong-Jin Kang, William I. Grosky, Tughrul Arslan, and Niki Pissinou, editors, *Computer Applications for Bio-technology, Multimedia, and Ubiquitous City*, pages 202–209, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[26] Danilo Cianciulli, Gerardo Canfora, and Eugenio Zimeo. Beacon-based context-aware architecture for crowd sensing public transportation

scheduling and user habits. *Procedia Computer Science*, 109:1110–1115, 12 2017.

[27] Laura Daniele, Patrícia Costa, and Luis Ferreira Pires. Towards a rule-based approach for context-aware applications. pages 33–43, 08 2007.

[28] Pari Delir Haghighi, Shonali Krishnaswamy, Arkady Zaslavsky, and Mohamed Medhat Gaber. Reasoning about context in uncertain pervasive computing environments. In Daniel Roggen, Clemens Lombriser, Gerhard Tröster, Gerd Kortuem, and Paul Havinga, editors, *Smart Sensing and Context*, pages 112–125, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[29] Anind Dey. Understanding and using context. *Personal and Ubiquitous Computing*, 5:4–7, 02 2001.

[30] Anind Dey and Gregory Abowd. Towards a better understanding of context and context-awareness. pages 304–307, 01 2000.

[31] Anind K. Dey. Providing architectural suppor t for building context-aware applications. 2000.

[32] Y. Feng, X. An, and S. Li. Application of context-aware in intelligent transportation cps. In *2017 36th Chinese Control Conference (CCC)*, pages 7577–7581, 2017.

[33] R. Fernandez-Rojas, A. Perry, H. Singh, B. Campbell, S. Elsayed, R. Hunjet, and H. A. Abbass. Contextual awareness in human-advanced-vehicle systems: A survey. *IEEE Access*, 7:33304–33328, 2019.

[34] Andrew Frank. Tiers of ontology and consistency constraints in geographical information systems. *International Journal of Geographical Information Science*, 157667-678, 01 1998.

[35] Shantappa Gollagi, M. Math, and A. Daptardar. A survey on pervasive computing over context-aware system. *CCF Transactions on Pervasive Computing and Interaction*, 2, 04 2020.

[36] Shantappa Gollagi, M. Math, and U.P. Kulkarni. Research outlook and state-of-the-art methods in context awareness data modeling and retrieval. *Evolutionary Intelligence*, 08 2019.

[37] E. Grim, C. Fok, and C. Julien. Grapevine: Efficient situational awareness in pervasive computing environments. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 475–478, 2012.

[38] Volker Haarslev, Kay Hidde, Ralf Möller, and Michael Wessel. The racerpro knowledge representation and reasoning system. *Semantic Web*, 3, 07 2012.

[39] Amina Hameurlaine and Samiha Brahimi. *Background on Context-Aware Computing Systems*, page 31. 10 2016.

[40] Ian Horrocks. The fact system. In Harrie de Swart, editor, *Automated Reasoning with Analytic Tableaux and Related Methods*, pages 307–312, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[41] Marjan Hosseini, Safia Kalwar, Matteo Giovanni Rossi, and Mersedeh Sadeghi. Automated mapping for semantic-based conversion of transportation data formats. In *1st International Workshop On Semantics For Transport*, volume 2447, pages 1–6, 2019.

[42] Davy Janssens, Geert Wets, Tom Brijs, and Koen Vanhoof. The development of an adapted markov chain modelling heuristic and simulation framework in the context of transportation research. *Expert Systems with Applications*, 28(1):105–117, 2005.

[43] A. Javadian Sabet, M. Rossi, F. A. Schreiber, and L. Tanca. Towards learning travelers' preferences in a context-aware fashion. In *Ambient Intelligence – Software and Applications*, pages 203–212, Cham, 2021. Springer International Publishing.

[44] Sun Jie and Wu ZhaoHui. Context reasoning technologies in ubiquitous computing environment. In Edwin Sha, Sung-Kook Han, Cheng-Zhong Xu, Moon-Hae Kim, Laurence T. Yang, and Bin Xiao, editors, *Embedded and Ubiquitous Computing*, pages 1027–1036, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[45] Christian Jung, Andreas Eitel, Denis Feth, and Manuel Rudolph. Dealing with uncertainty in context-aware mobile applications. 06 2015.

[46] Nesrine Khabou, Ismael Bouassida Rodriguez, and Mohamed Jmaiel. An overview of a novel analysis approach for enhancing context awareness

in smart environments. *Information and Software Technology*, 111, 11 2018.

[47] Radoslaw Klimek. Understanding human behavior in intelligent environments: A context-aware system supporting mountain rescuers. pages 267–279, 05 2017.

[48] Eyuphan Koc, Barbaros Cetiner, Adam Rose, Lucio Soibelman, Ertugrul Taciroglu, and Dan Wei. Craft: Comprehensive resilience assessment framework for transportation systems in urban areas. *Advanced Engineering Informatics*, 46:101159, 2020.

[49] Antonio Krüger and Rainer Malaka. Artificial intelligence goes mobile. *Applied Artificial Intelligence*, 18:469–476, 07 2004.

[50] Clemens Lachner, Thomas Rausch, and Schahram Dustdar. Context-aware enforcement of privacy policies in edge computing. pages 1–6, 07 2019.

[51] Y. Lee and K. Lien. Location based enabled context awareness information service. In *2009 International Conference on New Trends in Information and Service Science*, pages 944–947, 2009.

[52] Wei Li, Chunqiang Hu, Tianyi Song, Jiguo Yu, Xiaoshuang Xing, and Zhipeng Cai. Privacy-preserving data collection in context-aware applications. pages 75–85, 09 2018.

[53] L. Lijuan, L. Tingjie, and C. Xia. Network selection based on context-awareness services. *China Communications*, 10(12):50–56, 2013.

[54] Huaqing Lin and Yulong Fu. Adaptive security-related data collection with context awareness. *Journal of Network and Computer Applications*, 126, 11 2018.

[55] Roger S. Machado, Ricardo B. Almeida, Ana Marilza Pernas, and Adenauer C. Yamin. State of the art in hybrid strategies for context reasoning: A systematic literature review. *Information and Software Technology*, 111:122–130, 2019.

[56] Moeiz Miraoui and Chakib Tadj. A service oriented definition of context for pervasive computing. 01 2007.

[57] Hossein Moradi, Bahman Zamani, and Kamran Zamanifar. Caasset: A framework for model-driven development of context as a service. *Future Generation Computer Systems*, 105:61–95, 2020.

[58] Grzegorz Nalepa and Szymon Bobek. Rule-based solution for context-aware reasoning on mobile devices. *Computer Science and Information Systems*, 11:171–193, 01 2014.

[59] G. Orsi, L. Tanca, and E. Zimeo. Keyword-based, context-aware selection of natural language query patterns. In *EDBT/ICDT '11*, 2011.

[60] Abayomi Otebolaku and G. Lee. Towards context classification and reasoning in iot. *2017 14th International Conference on Telecommunications (ConTEL)*, pages 147–154, 2017.

[61] Zhiwen Pan, Salim Hariri, and Jesus Pacheco. Context aware intrusion detection for building automation systems. *Computers Security*, 85:181–201, 2019.

[62] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Ca4iot: Context awareness for internet of things. In *2012 IEEE International Conference on Green Computing and Communications*, pages 775–782, 2012.

[63] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Context aware computing for the internet of things: A survey. *IEEE Communications Surveys Tutorials*, 16(1):414–454, 2014.

[64] Preeja Pradeep and Shivsubramani Krishnamoorthy. The mom of context-aware systems: A survey. *Computer Communications*, 137:44 – 69, 2019.

[65] Preeja Pradeep, Shivsubramani Krishnamoorthy, and Athanasios V. Vasilakos. A holistic approach to a context-aware iot ecosystem with adaptive ubiquitous middleware. *Pervasive and Mobile Computing*, 72:101342, 2021.

[66] Anand Ranganathan and RoyH Campbell. An infrastructure for context-awareness based on first order logic. *Personal and Ubiquitous Computing*, 7:353–364, 12 2003.

[67] Juan Recio-García, Belén Díaz-Agudo, and Pedro Gonzalez-Calero. The colibri platform: Tools, features and working examples. *Studies in Computational Intelligence*, 494:55–85, 09 2014.

[68] Eduardo Roldán, Stéphane Negny, Jean Marc Le Lann, and Guillermo Cortés. Modified case based reasoning cycle for expert knowledge acquisition during process design. In E.N. Pistikopoulos, M.C. Georgiadis, and A.C. Kokossis, editors, *21st European Symposium on Computer Aided Process Engineering*, volume 29 of *Computer Aided Chemical Engineering*, pages 296 – 300. Elsevier, 2011.

[69] M. Roman, Hess CK, Renato Cerqueira, Anand Ranganathan, Roy Campbell, and Klara Nahrstedt. A middleware infrastructure for active spaces. *Pervasive Computing, IEEE*, 1:74– 83, 11 2002.

[70] Patrick Rosenberger, Manfred Grafinger, Detlef Gerhard, Martin Hennig, and Stefan Dumss. The casad matrix method: Introduction of a technique for the documentation, analysis, and optimization of context-aware systems. *Procedia CIRP*, 91:427 – 432, 2020. Enhancing design through the 4th Industrial Revolution Thinking.

[71] Alireza Javadian Sabet, Matteo Rossi, Fabio Alberto Schreiber, and Letizia Tanca. Context awareness in the travel companion of the shift2rail initiative. In *Proceedings of the 28th Italian Symposium on Advanced Database Systems*, volume 2646 of *CEUR Workshop Proceedings*, pages 202–209. CEUR-WS.org, 2020.

[72] Mersedeh Sadeghi, Petr Buchníček, Alessio Carenini, Oscar Corcho, Stefanos Gogos, Matteo Rossi, and Riccardo Santoro. SPRINT: Semantics for PerfoRmant and scalable INteroperability of multimodal Transport. In *TRA*, pages 1–10, 2020.

[73] Michael Samulowitz, Florian Michahelles, and Claudia Linnhoff-Popien. Capeus: An architecture for context-aware selection and execution of services. In Krzysztof Zieliński, Kurt Geihs, and Aleksander Laurentowski, editors, *New Developments in Distributed Applications and Interoperable Systems*, pages 23–39, Boston, MA, 2001. Springer US.

[74] Bill Schilit and Marvin Theimer. Theimer, m.m.: Disseminating active map information to mobile hosts. ieee netw. 8(5), 22-32. *Network, IEEE*, 8:22 – 32, 10 1994.

[75] Fabio Schreiber and Emanuele Panigati. Context-aware self adapting systems: a ground for the cooperation of data, software, and services. *International Journal of Next-Generation Computing*, 8:32–61, 03 2017.

[76] Fabio Schreiber and Maria Elena Valcher. Formal verification of context aware systems *. 07 2020.

[77] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu. Context-aware computing, learning, and big data in internet of things: A survey. *IEEE Internet of Things Journal*, 5(1):1–27, 2018.

[78] Rob Shearer, Boris Motik, and Ian Horrocks. Hermit: A highly-efficient owl reasoner. volume 432, 01 2008.

[79] C. Silva and M. A. R. Dantas. Quality-aware context provider: A filtering approach to context-aware systems on ubiquitous environment. In *2013 IEEE 9th In. Conf. on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 422–429, 2013.

[80] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical owl-dl reasoner. *Jou. of Web Semantics*, 5(2):51–53, 2007. Software Eng. and the Semantic Web.

[81] Thomas Strang and Claudia Linnhoff-Popien. A context modeling survey. In *First International Workshop on Advanced Context Modelling, Reasoning And Management at UbiComp 2004, Nottingham, England, September 7, 2004*, September 2004. LIDO-Berichtsjahr=2004, Potential plagiarism at http://dl.comsoc.org/cocoon/comsoc/servlets/GetPublication?id=9014634.

[82] Rajarajeswari Subbaraj and Neelanarayanan Venkataraman. Reasoning in context aware computing – a review. *International Journal of Pharmacy and Technology*, 8:5021–5032, 11 2016.

[83] Peyman Talebifard. Context-aware dissemination of information and services in heterogeneous network environments. *Journal of Ambient Intelligence and Humanized Computing*, 5, 12 2013.

[84] Kun Tang, Shuyan Chen, and Aemal J. Khattak. Personalized travel time estimation for urban road networks: A tensor-based context-aware approach. *Expert Systems with Applications*, 103:118–132, 2018.

[85] C. Tsatsoulis, Q. Cheng, and H. . Wei. Integrating case-based reasoning and decision theory. *IEEE Expert*, 12(4):46–55, 1997.

[86] Hamed Vahdat-Nejad, Azam Ramazani, Tahereh Mohammadi, and Wathiq Mansoor. A survey on context-aware vehicular network applications. *Vehicular Communications*, 3:43–57, 2016.

[87] A. H. van Bunningen, L. Feng, and P. M. G. Apers. Context for ubiquitous data management. In *International Workshop on Ubiquitous Data Management*, pages 17–24, 2005.

[88] Sélinde van Engelenburg, Marijn Janssen, and Bram Klievink. Designing context-aware systems: A method for understanding and analysing context in practice. *Journal of Logical and Algebraic Methods in Programming*, 103:79–104, 2019.

[89] Sélinde van Engelenburg, Marijn Janssen, and Bram Klievink. Designing context-aware systems: A method for understanding and analysing context in practice. *Journal of Logical and Algebraic Methods in Programming*, 103:79–104, 2019.

[90] Norha M. Villegas and Hausi A. Müller. *Managing Dynamic Context to Optimize Smart Interactions and Services*, pages 289–318. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[91] Xiaohang Wang, Daqing Zhang, Tao Gu, and Hung Pung. Ontology based context modeling and reasoning using owl. pages 18–22, 01 2004.

[92] Wei Liu, Xue Li, and Daoli Huang. A survey on context awareness. In *2011 International Conference on Computer Science and Service System (CSSS)*, pages 144–147, 2011.

[93] Jong yi Hong, Eui ho Suh, and Sung-Jin Kim. Context-aware systems: A literature review and classification. *Expert Systems with Applications*, 36(4):8509 – 8522, 2009.

[94] Andreas Zimmermann, Andreas Lorenz, and Reinhard Oppermann. An operational definition of context. In Boicho Kokinov, Daniel C. Richardson, Thomas R. Roth-Berghofer, and Laure Vieu, editors, *Modeling and Using Context*, pages 558–571, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.