Cartographic Representation on Small Devices

Safiza Suhana Binti Kamal Baharin

Delft, October 2007

RGI 149 Report Nr 2

Cartographic Representation on Small Displays

Safiza Suhana Binti Kamal Baharin

Delft, October 2007

RGI 149 Report Nr 2

Summary

This report describes the information that may be used in the BSik project RGI-149 "Geo-info to-go", which is applying a push approach to displaying the correct amount of geo-information to users. In this context the correct amount of information means only the information that is relevant to the user profile (identity), the current location and the user's behaviour and that can be displayed on a mobile user device, such as a PDA. In other words, users can use a mobile device offering adaptive cartography to obtain information without interacting with the device. The information will be provided automatically.

This was the inspiration behind the proposal of the RGI-149 Geo-information to-go project to adopt this approach for environmental officials (for example, pollution control officers). This integration of context awareness and mobile technology can be used to identify environmental crimes and raise environmental awareness. By using the adaptive method, environmental crime officers could be provided with the correct amount of geo-information. In this context the correct amount of geo-information means that the environmental officers will be provided only with information related to their profile as investigative officers and to their position. The question of how to implement this approach is the central issue behind this report. The report will explain in theory the limitations on the presentation of cartographic data on small screen displays and the possible solutions to the problem. The report also looks at the context awareness concept and the adaptive method and, in conclusion, the benefits of this approach for environmental awareness.

Project leader	: Section GIS Technology OTB Research Institute for Housing, Urban and Mobility Studies Delft University of Technology
Address	: Jaffalaan 9 2628 BX Delft
Contact	: ir. E. Verbree
Tel.	: +31 (0)15 278 2584
Fax	: +31 (0)15 278 2745
E-mail	: e.verbree@tudelft.nl
Website	:

Contents

1. Introduction 3

2. The Background of Mobile Cartography 4

2.1 Introduction 4

2.2 Introduction to Cartography 4

2.3 Mobile Cartography 5

2.4 The Limitation of Small Screen Size in Mobile Cartography 6

2.4.1 Cartographic Information 6

2.4.2 Auxiliary Information 7

2.4.3 The Legibility of Symbols and Text 8

2.4.4 Limited Interactivity 8

2.4.5 Generalisation 9

3. State-of-the-Art Context-Aware and Adaptive Services **10**

3.1 Introduction 10

3.2 Context-Aware Computing 10

3.3 A Context-Aware Architecture based on Web Services 13

3.4 Adaptive Systems 16

3.5 Adaptive Services 20

3.6 The Adaptation Principle and Adaptation Process 21

3.7 Adaptable Objects in Mobile Map Services 21

4. Conclusion **26**

References & Bibliography **27**

1 Introduction

This report describes the information that may be used in the BSik project RGI-149 "Geo-info to-go", which is applying a push approach to displaying the correct amount of geo-information to users. In this context the correct amount of information means only the information that is relevant to the user profile (identity), the current location and the user's behaviour and that can be displayed on a mobile user device, such as a PDA. In other words, users can use a mobile device offering adaptive cartography to obtain information without interacting with the device. The information will be provided automatically. This can be described using the analogy of a tourist guide who offers an explanation with appropriate information without the tourist needing to ask a question. For example, holidaymakers travelling by boat will have the buildings and landmarks along the river explained by the tourist guide. The guide will only describe what can be seen from the boat along the banks of the river. He will never describe things that the tourists cannot see or provide irrelevant information, such as the result of a soccer match. And all the information is given without the travellers needing to ask questions. However, if one of the holidaymakers has different interests from the others in the same group, he could ask for additional information. This is the scenario in which the conversation plays a major role. If we look at the application of route-guiding, for example, where the user wants to travel from one point to another as quickly as possible, he can use a mobile device that provides the correct amount of geo-information and also a map or verbal instructions. In this case the correct amount of information could relate to the individual user and whether or not he is new to the area and therefore requires additional information.

This was the inspiration behind the proposal of the RGI-149 Geo-information to-go project to adopt this approach for environmental officials (for example, pollution control officers). This integration of context awareness and mobile technology can be used to identify environmental crimes and raise environmental awareness. By using the adaptive method, environmental crime officers could be provided with the correct amount of geo-information. In this context the correct amount of geo-information means that the environmental officers will be provided only with information related to their profile as investigative officers and to their position. The question of how to implement this approach is the central issue behind this report. The report will explain in theory the limitations on the presentation of cartographic data on small screen displays and the possible solutions to the problem. The report also looks at the context awareness concept and the adaptive method and, in conclusion, the benefits of this approach for environmental awareness.

2. The Background of Mobile Cartography

2.1 Introduction

This chapter begins with a brief explanation of cartography and this is followed by a description of mobile cartography. The following section covers the limitations of small screen size in mobile cartography in more detail and highlights possible solutions in cartography design to resolve these limitations.

2.2 Introduction to Cartography

Cartography can be defined as the art, science, technology and craft of making maps. It covers all the steps needed to produce a map. According to Wikipedia, this term comes from the Greek term *chartis* meaning a map and *graphein* meaning to write. Traditionally, maps have been created using pen and paper, but the advent of computers has revolutionised cartography. Nowadays, maps are made using map-making software, such as CAD, GIS etc. Maps function as a visualisation tool for spatial data. In other words they provide a graphic representation of the physical features of all or part of the earth's surface. The representation can use signs and symbols or photographic imagery at a suitable scale, with a specified projection and orientation.

Recent developments in internet technology have indirectly brought a new perspective to cartography, which is referred to as web cartography. As a result of the availability of mobile hardware devices and mobile access to the internet, GIS technology and web cartography have also made major advances and the concept of mobile cartography has developed. Mobile cartography provides maps for mobile devices, such as laptops, mobile phones, smartphones, PDAs and other portable devices connected to a public network. This has resulted in a new approach to cartography.

Mobile devices have a small visual display for user output and a miniature keyboard for user input. In addition, mobile applications are different from their desktop counterparts. Communication methods include the written or spoken language or graphics. However, the most effective way of communicating cartographic concepts is to use maps, which are in graphic form. As the old saying goes "A picture is worth a thousand words". This highlights the importance of cartography as a visualisation technique. Using graphics, people receive information in a visual and synoptic form and the information is provided all at once, instead of in a sequence of events. The process of designing the map is vitally important in this context. According to Nagi (2004), the map design process is guided by the question "How do I say what to whom and is it effective?". As a result the process begins by defining the purpose of the map and collecting the relevant data, including the selection of geographic features and attributes, the map projection, the region of the map, the appropriate scale, the needs of the map user, the appropriate symbols, the generalisation of features and attributes, and also the overall layout of the map.

The cartographic visualisation process is shown in Figure 2.1 below.

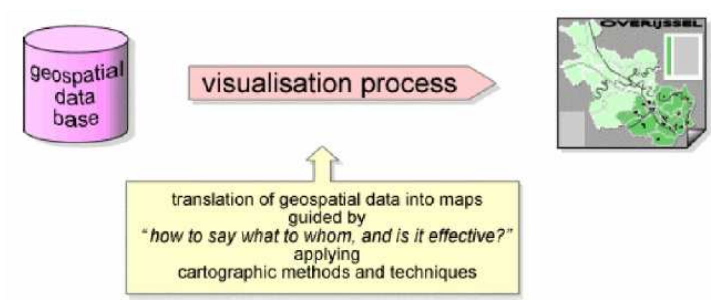

Figure 2.1: The cartographic visualisation process
(Source: Brouwer, P. and Samiei, E. S., 2003).

An understanding of the features of mobile devices and of the differences between them and fixed devices, such as desktop computers, is essential in order to be able to design and develop an effective mobile map application. This report will cover the theory of representing maps on mobile devices with small displays and other limitations.

2.3 Mobile Cartography

Before discussing the issue of the appearance or design of mobile cartographic displays, we need to understand the meaning of the term mobile cartography. According to Reichenbacher (2001a), mobile cartography is primarily a very flexible and dynamic way of presenting spatial data to a mobile user based on his context and his profile. Reichenbacher (2001b) also mentions that mobile cartography deals with the theories and technologies of the dynamic cartographic visualisation of spatial data and its interactive use on portable devices anywhere and at any time, under special consideration of the current context and user characteristics. This means that the user will always be provided with the right information at the right time and in the right place. Whoever the user is, he will always obtain the relevant information based on his position, context, interests, knowledge and skill level. The integration of GIS technology with internet technology and mobile computing, which is referred to as mobile GIS, seems to fulfil all of these requirements. Location-based services (LBS), car navigation systems, location-based weather and traffic information, mobile tourist guides etc. are examples of applications that use this combination of technologies.

The popularity of handheld devices and mobile internet services gives a new dimension to geo-information science and cartography. Developing a suitable cartographic display method for small mobile devices has become a major challenge for researchers, because of limitations such as screen size, colour, resolution, processing power, memory, power supply and bandwidth. Despite the fact that the latest technological developments have removed the majority of these restrictions, the small size of the display remains a problem and requires further investigation in order to identify a solution.

When the position of the device and therefore the position of the user is part of the information, this is referred to as a location-based service (LBS). Terms such as mobile cartography, ubiquitous cartography and telecartography are all used to describe the distribution of map data via wireless data transfer interfaces to mobile devices.

In order to be able to provide an effective service, every map-based LBS requires certain basic elements to handle the main tasks of positioning, data modelling and information presentation. Knowledge of the position of the mobile device is a direct requirement for every LBS system. For some applications, the level of accuracy needed is equivalent to that of the cell-id of a telecommunications network, which gives a positioning accuracy of between 50 and 100 metres in urban areas. This can be reduced to as little as 25 metres for pedestrian navigation, for example, and may be lowered even further for indoor navigation. In addition, within the field of positioning, current research is focused on improving existing methods and finding an integration concept for determining the user's position. In Kopczynski (2003) sketch-based input is suggested as a localisation method. Verbree, Tiberius and Vosselman (2003) use the combination of GPS and Galileo signals and Zlatanova and Verbree (2003) propose user tracking as an alternative approach for establishing the user's position.

However, this report will focus on the basic requirements for presenting a mobile map on a mobile device. The term mobile map refers to the use of maps on portable devices that users carry with them as they move around. As they can present up-to-date spatial and non-spatial information in a very individual, dynamic and flexible way, they offer a number of benefits for personal mobility.

2.4 The Limitation of Small Screen Size in Mobile Cartography

The possibility of transmitting and visualising mobile geospatial information is restricted by the limitations of the mobile device. The representation of a map on a PDA is obviously different from that on a laptop. Therefore it is important to design a flexible cartographic model that can adapt to the user and the tasks. In other words, the restrictions on the size and format of mobile devices may result in the solutions for presenting information within a map-based LBS being broken down into different levels. The first level involves a cartographic presentation without specific adaptations. In the second level the presentation is adapted to the specific requirements of the display. The third level includes multimedia add-ons, replacements and alternative presentation forms. In this report, the main focus is on the question of cartographic representation on a small device and, in general, on the task of presenting a map on a small display designed for graphical modelling and visualisation (Gartner, 2007).

Normally, the small display on mobile devices puts limitations on the cartographic data that can be represented, because it is not possible to display the level of detail generally used in paper or digital maps. The symbols, visualisation and information presentation cannot be discussed without taking into consideration the restrictions on mobile devices mentioned earlier in the report. Research into these limitations is currently underway in a number of areas with the aim of developing new forms of cartographic presentation.

Reichenbacher (2004) mentioned in his thesis that small display cartography for mobile devices is restricted by technical limitations, such as the lack of processing of power and memory and, critically, the battery life.

However, these limitations and problems are gradually being solved by the rapid developments in telecommunication technologies and the arrival of advanced devices on the market. For instance, in the telecom industry GPRS and UMTS have solved the problem of limited bandwidth and the rapid developments in mobile devices, such as smartphones and PDAs, have resolved the limitations caused by colour, power supply, memory and processing power. The only problem that remains is related to the small size of the display. This limitation results in problems for map design, such as the presentation of cartographic and auxiliary information, the legibility of symbols and text, limited interactivity and the generalisation process.

2.4.1 Cartographic Information

When representing cartographic information on a small display, it is inevitable that some information will be lost. In personal navigation applications, for example, where the user uses a mobile device to help him find a certain location, sometimes both a detailed map of the area with the surrounding environment and the current position and an overview map are needed at the same time, so that the user does not lose his focus. However, as a result of the limited size of the screen, it is difficult to display all the necessary information simultaneously and to include the overview. There are four possible solutions for this problem. Firstly, a schematic map can be used. However, this kind of map is most suitable for representing underground railway systems and transportation networks, despite the fact that it also has limited applications for personal navigation. Although it is often claimed that people are only interested in the data relating to their route and ignore the information about their surroundings and the link between them, in fact it is better to reduce the size of some roads, so that well-known landmarks can be shown. The question here is how to choose appropriate roads or landmarks that could be displayed? Is it possible for the system to provide the information automatically or does the user have to select it?

The second approach involves displaying a large-scale map (detailed map area) simultaneously with a small-scale map (overview map) by using an ordinary map and a key map. The large-scale map can be shown in the ordinary map and the small-scale map in the key map. However, using this method the key map hides the data in the ordinary map window. In addition there is still not enough space for both maps at the same time on a small display (see Figure 2.2).

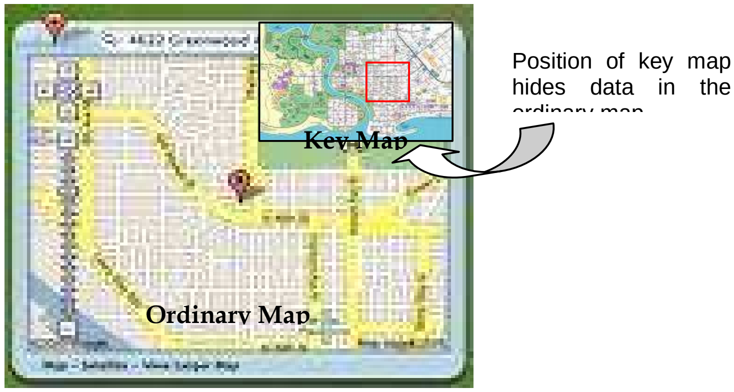

Figure 2.2: Use of ordinary map and key map

In order to resolve the problem described above, Meng, Zipf and Reichenbacher (2005) suggested a method using movable components, in which the area surrounding the user could be shown at the size of the ordinary map, while the small-scale map (key map) is a floating element which allows the user to drag the overview window to a position where it does not conceal any relevant information.

The third approach involves using space contraction. In this case, the relevant area is made up of spatially widely separated locations. The aim of this method is to make the associated content visible at the same time and within the same window. The level of detail (LoD) and topographical relations remain unchanged during the contraction process. This means that the locations which the user is currently interested in have been brought together despite the fact that they are far away from each other. This kind of method may be suitable for viewing all the areas on a small-scale map. However, it can also be used with large-scale maps with a zoom function. It is not appropriate for in-car navigation as it requires the user to use additional tools in order to zoom in on the map. This type of design may eliminate the need for scrolling and panning functions.

The final method uses a variable scale map which has both large-scale and small-scale cartographic data presented on the same map or which integrates a large-scale map and a small-scale map, as mentioned in Harrie, Sarjakoski and Lehto (2002).

2.4.2 Auxiliary Information

As a result of the small displays on mobile devices there is a lack of space to present auxiliary information, such as a legend which describes the meaning of symbols. In mobile situations, the user has little or no time to identify the meaning of each symbol. Designing symbols for mobile cartography without a legend being needed represents a challenge for the cartographer. The question is how to ensure that the symbols can be understood by every user without a legend. Normally, symbols are difficult to understand, in particular for new users who do not use maps frequently. If symbols are misunderstood, this may cause problems for the user. There are three possible approaches to resolving this problem. Firstly, it may be possible to use pictographic symbols. This is because geometric symbols are hard to understand without a legend. Pictographic symbols of an appropriate size which are familiar to all users can be used instead. For example, users can identify the location of an airport from the pictographic symbol shown in Figure 2.3 below.

Figure 2.3: Airport pictography symbol

The second approach is to integrate a pop-up legend which users can turn on and off. In some cases, such as in-car navigation, this approach would not be suitable. Using tool-tips is another possible method. When the user holds the cursor or mouse pointer over an item, a small box appears containing the name or description of the item. Tool-tips and mouse-overs may be a practical solution for personal navigation or tourist applications, but not for in-car navigation.

Thirdly, the auxiliary information could be hidden behind the object. The information is only displayed when the user interacts with the object. In other words the object becomes active when the user holds the mouse pointer over it. Detailed information is displayed when the user clicks on the object.

2.4.3 The Legibility of Symbols and Text

The next problem involved in displaying cartographic information on small screens is the legibility of symbols and text. It is not possible to apply the minimum size of symbols defined for paper or digital maps to mobile maps, because of several factors, including the typical characteristics of the mobile device itself, which has a limited number of pixels. In addition it is important for the symbols to be large enough for their meaning to be understood. Labelling each feature is not an effective solution because it will make the map cluttered and difficult to read. Outdoor viewing conditions, such as sunlight, that will change the use of colour and visual contrast also have to be taken into account in map design. Two other possible approaches are zooming and panning or a bird's-eye view. In the zooming and panning approach, one option is static stepped zooming which uses a multiple representation database (MRDB) to store the same real-world phenomena at different levels of precision, accuracy and resolution. In order to avoid the use of a generalisation technique, a high-speed link with the server is needed. This problem will be resolved by UMTS or advanced technology in the future.

Dynamic zooming is another alternative where there is a direct relation between the map scales and the content. This means that the larger the scale, the more detail is shown on the map. Cartographic symbols may also change as the scale changes. Therefore this method also requires the use of a large MRDB because of the complex algorithm needed for generalisation at each level of detail.

The second approach is the bird's-eye view technique that makes reading maps easier and more intuitive, especially in a mobile environment. The bird's eye point of view (BEPV) is a special kind of viewpoint identification in a virtual environment. It provides intuitive 2D or 3D controls so that the user can create cognitive maps of the environment for use in navigation. Maps will help the user to identify his current position and direction in a virtual environment. However, a BEPV motion is interlinked with the movement of the user and is controlled by intuitive manipulation. The rapid technological developments in the field of games, for example, may result in the creation of a sophisticated display technique or pointing technique using voice recognition, motion recognition or gesture recognition etc.

2.4.4 Limited Interactivity

In a mobile situation, it is impossible for the user to interact with the device frequently. For example, the user can change the mode from the large-scale to the small-scale map to see both overview and detailed information or he can adjust the contrast of the display, for example, while on the move. If a car driver interacts too often with an in-car navigation system, this may result in an accident or even loss of life. Multimedia solutions can be an alternative way of resolving this kind of problem. Where there is little space available on the display and the interaction options for displaying information graphically are limited, multimedia elements, such as audio, video and animation, can be integrated as a new way of delivering the information.

2.4.5 Generalisation

Geo-visualisation was not originally designed for use in mobile situations. In addition, if the user interface is poorly designed and does not take the requirements and input modes of mobile interactivity into account, this could result in usability problems. Devices with small screens present a problem for cartographers when designing and displaying maps. The indirect effect is to require the selection of cartographic data. Information cannot be displayed in as much detail as on paper or digital maps. Therefore, in order to reduce the amount of information and ensure that the information can be displayed on a small device, a generalisation process is needed. However, generalisation may not reduce the content of the information. In other words, cartographers are faced with the challenge of conveying the whole message using less content and of producing results from the generalisation process that can be used by every user.

The context awareness and adaptive map approach is one way of ensuring that users can specify the information that they really need or that they can specify their purpose and allow the system to select the necessary data, leaving out the unimportant details. The basic concept of this approach is that if the system has sufficient information relating to the surroundings of the situation where the map is being used and the user's current activity, then the map can be adapted to the context, so that the user is provided with the correct amount of information which is tailored to the situation and his specific requirements.

Adaptive maps and context awareness are new areas of research within the field of cartography. These are new techniques that can adapt the surrounding context in order to provide users with an appropriate map based on their real needs. In other words, users are unlikely to be frustrated when they try to understand the map because it provides the correct amount of information, no more and no less. Adaptation to the surrounding context can improve the usability of the map and result in satisfied users by preventing unnecessary interruptions. Because context adaptation can reduce the amount of information to only the required level, it can have a major impact on the limitations of cartographic representations on small displays.

3. State-of-the-Art Context-Aware and Adaptive Services

3.1 Introduction

According to Dey (2001), a context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves. If a piece of information can be used to characterise the situation of a participant in an interaction, then that information is context. Take, for example, a museum guide application. In this example, the entities are the user, the application and the museum. Two pieces of information could be the weather and the presence of other people. How can we tell which of these is context? The weather does not affect the application, because it is being used indoors. Therefore the weather is not context. The presence of other people can be categorised as the user's situation. Therefore this is context because it can be used to categorise or characterise the situation.

Context entities can be divided into three categories or domains. These are the user domain, (such as the user's profile, location, people nearby and current social situations), the computing domain, (such as network connectivity, communication bandwidth and nearby resources, for example, printers, displays etc.) and the environment domain, (such as noise level, lighting, temperature, weather, traffic control etc.).

The full context information for one specific entity can be considered to be a set of data objects. Each of the elements in the set has four parameters: the entity name, the feature, the value and the time. Context features fall into two categories: internal and external. Internal features are used to describe those characteristics inside the entity in its domain, whereas external features are used to describe the context information concerning the interaction of the entity with other entities. A feature may have a value of any significant type, such as scalar, vector, symbolic, integer, character, real, string, structure etc.

If the software is adaptable and can use context information, then data relating to nearby people and objects can be collected in order to provide relevant information and services to the user, where the relevancy of the information depends on the user's task. We refer to this technology as context-aware computing. Using the notion of context in software is a paradigm in which information about the situation where the system is being used is gathered and structured in such a way that this information can be used to adapt the software to the situation.

This method, which is a result of the recent revolution in computing technology and the trend for increasingly intelligent systems, involves transferring information from the human world into the computer. The computer takes advantage of the context to adjust its behaviour in order to improve the services provided to users.

3.2 Context-Aware Computing

Context-aware computing relates to the ability of computer systems to obtain contextual knowledge in order to perform relevant tasks. Rather than treating mobility as a problem to be solved, context-aware computing seeks to exploit its nature. As a consequence, it creates a new generation of applications in which the interaction between users and applications is enhanced by the perception of the surrounding environment. It is expected that the dynamic adaptation of devices and applications in a changing physical and social environment will result in an enhancement of the user experience.

Context awareness in a map context lays the foundations for the possible adaptation of the maps to the context at hand. The most important context domain related to mobility is location. Location information is linked to a position and denotes different levels of granularity with distinct ranges of values. For example, location-based services apply the location context in order to filter information about the user's current location. Time is almost as important as location. Time could, for instance, represent the exact system time, the time of day or even the season (the environment domain of the context). Location and time together constitute the situation. The user is also one of the context dimensions. It is a case of identifying which user characteristics need to be modelled and which attributes are important and relevant for mobile information usage and the accompanying adaptations, such as identity, preferences, knowledge, skills etc. Using this information it is also possible to define the appropriate activities for modelling rules and constraints in specific contexts. In the computing domain of the context, information about the device and the network in use, their characteristics and functionality and the available infrastructure influences the way in which information should best be transmitted and visualised.

Individual context dimensions cannot be considered independently. It is important to analyse the different relationships between them. The majority of the relationships are between the user and other context dimensions. The user provides the mobile device and determines the location of the device and the user indirectly (user domain). The user has access to the device and its functionality. The system conditions also depend on the device location, where the location also defines the current temperature, which has a major influence on the battery life (computing domain). Activities are always situated in space and time and differ depending on the user's role. The location and the time will restrict certain activities, influence the possibility or importance of activities and affect their quality. Depending on the time, the significance, importance or accessibility of a location may differ (environment domain).

Take, for example, a touring guide application. A user wants to travel from point A to point B. Instead of providing a map which includes irrelevant information, only sufficient, selected information is offered relating to the user's purpose, current position and situation. This kind of method might offer an indirect solution to the problem of small displays and file size. If only relevant information is displayed, the map will not be cluttered with irrelevant information and unnecessary details which contribute to the space limitations. The result is that the user can find specific objects more easily on the map. In addition the map will also help the user to identify specific buildings.

However, in order to supply the correct amount of information to the user, the map should always relate to the user's current surroundings or situation. For example, if a user is trying to find his way in an unfamiliar environment, the user context information includes important factors, such as the user's identity, the method of transport he is using, his age, his physical constraints and his purpose (entertainment, sport etc.). In addition, the user's physical surroundings are also relevant, for example the weather, the location of rivers, lakes and hills in relation to the user's destination, the user's social situation at the time and the type of mobile device he is using.

Some contextual factors do not have a significant influence on the visualisation of information. Figure 3.1 outlines the relationship between the most important factors and their possible values. Several context dimensions can have a major impact on mobile maps. For example, the season could influence the features shown on the map and the symbols used. The age of the user could influence the style of the symbols and their size. The user's current activity and role could affect the map content, style and graphics.

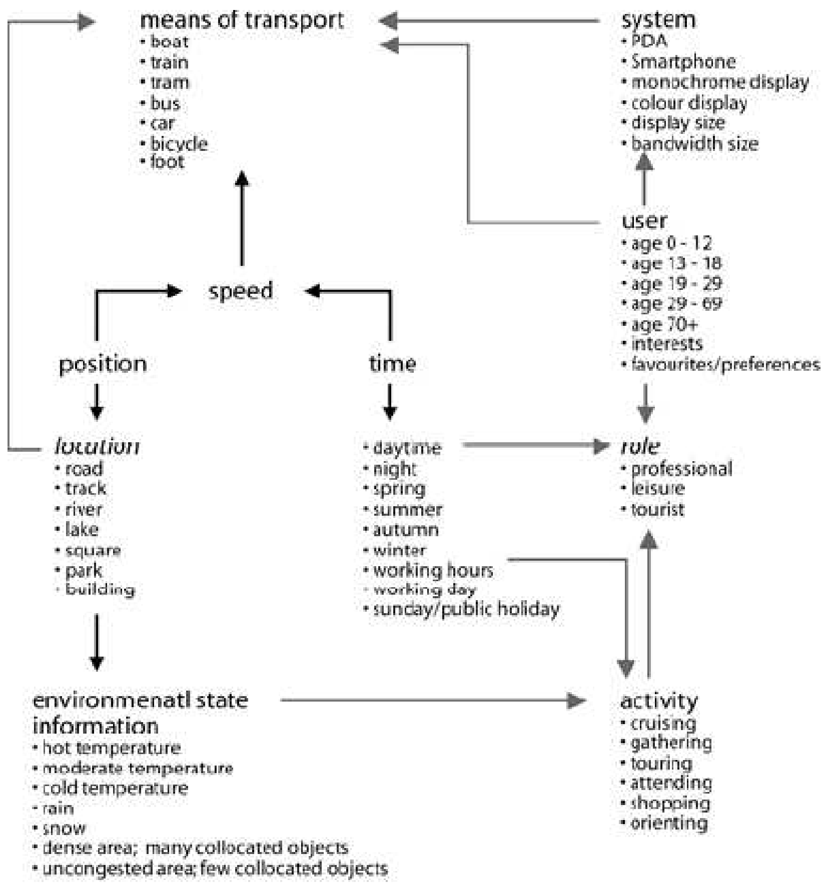

Figure 3.1: Possible values of context factors and their relationships
(Source: Reichenbacher (2007)

The GiMoDig (Geospatial Info-Mobility Service by Real-Time Data-Integration and Generalisation) application, for example, provides a service that delivers maps in real time to mobile devices. This application also takes into consideration user customisation in order to provide a service. This means that after the service has been personalised, the maps are automatically adapted to the different users in different usage situations. However, this application needs a user to push particular information about himself before the system can supply the correct amount of spatial information. In other words, an intelligent map could be created from the user's viewpoint on the basis of embedded context awareness.

Context awareness has emerged as an important and desirable feature in distributed mobile applications. This involves the ability of applications to use information about the user's environment (context) in order to select and provide relevant services dynamically that better match the user's needs. In a general environment, where many are services available at any one time, context information is especially important in order to help determine which services are relevant for the user. Building context-aware systems involves several new challenges which are mainly related to gathering/sensing, modelling, storing, distributing and monitoring contextual information. As a result proper architectural support is needed.

Context-aware applications usually combine context management code with the application code. As a result the application code becomes more complex and more difficult to read and maintain. In order to maintain the application, the boundary between contextual data and application data has to be clearly defined. This depends on the application domain, since data which is on the application level in one domain can be regarded as context data in another domain. For example, the GPS location is context data in a telemedicine application, but is application data in a traffic regulation system. We can define the context as the set of external parameters that can influence the behaviour of the application. Context parameters evolve during application runtime. They are not significant to the user and therefore they must be transparent to him. A new instance of these parameters characterises a new context situation. This does not modify the application data, but may select or process it in a different way. For example, in a home care application, when a doctor moves from one patient residence to another, the patient records do not change. They are part of the application data. However, the current patient *id* changes; it is therefore a context parameter.

In Frank (2005) empirical evidence and analyses of both conventional printed maps and interactive web maps are used. These two types of maps were given to two test groups which were responsible for demonstrating a pattern of map usage. The objective of the study was to determine the effectiveness of spatial information transfer. In addition the results were indirectly relevant to the current debate on function-oriented map design. The results also have an impact on web map design.

In this study the most important methodological aspect is the time required to absorb and understand the information contained in an online map. The transmission of information depends on two factors: effectiveness, in other words, information must be transmitted completely and correctly, and efficiency, which means that the results must be seen in relation to the amount of time required to obtain full and correct information. However, recent research findings have revealed the existence of a third usability factor in addition to effectiveness and efficiency. This is the degree of satisfaction or pleasure experienced by the map user, which could be used as a benchmark for good map design.

Applying context awareness to the mobile map could help with map reading and, in addition, the navigation and the usability of the map may be improved. The topographic data set alone will not satisfy users' needs (the data is difficult to understand and read), especially without a legend and with the other limitations on cartographic presentation discussed earlier in the report. Therefore the solution must include both topographic data sets and more enriched information, (such as points of interest (POI), additional information and linked information sources), related to the user's surroundings.

Using the context information available, the map service must also be able to adapt the visualisation to different usage situations and individual users' needs.

3.3 A Context-Aware Architecture based on Web Services

Users may access different data and exploit different aspects of an application. For example, in one context a doctor may access a health database in order to screen patients for preventive purposes, while in a different context the same doctor may access the same database for the post-treatment analysis of cases. While the data is the same, the way it is returned may vary according to the doctor's objective. Often, in different contexts, users access almost the same data and the same services but receive answers which have a different structure, presentation and, possibly, different content detail. For example, a doctor examines a patient record at the hospital using a desktop computer connected to the hospital database, consults the same record stored on a PDA while visiting the patient at home or receives an audio description of the patient record during a surgical operation.

Figure **3.2** below describes the scenario.

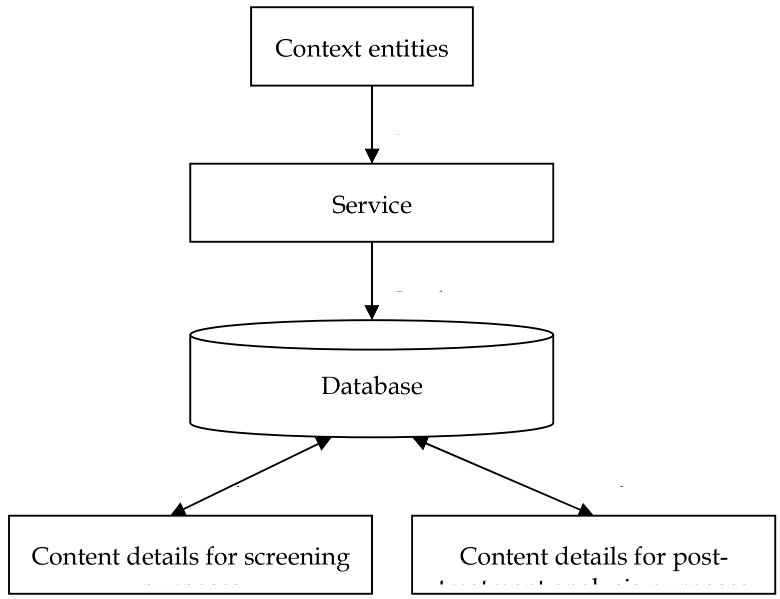

Figure 3.2: Scenario of a context-aware system

Context awareness and adaptation are closely related, and the two terms are often used to mean the same thing. However, they refer to different capabilities. **Adaptation** is the ability to provide different versions of a service or different presentations of a document to suit the needs of the user, the environment, the equipment etc. **Context awareness** is the capability of perceiving all the aspects of the user's situation and of adapting the behaviour of the system as a consequence, in other words the services, the data and the interface. Adaptation is therefore the goal of context awareness, which can drive context awareness without explicit intervention by the user. A context-aware application must manage the context as one of its inputs, processing any user requests on the basis of the different contexts. However, it is simplistic to consider the context as one of the items of application input data, because of the difficulty of classifying in advance all the combinations of user situation, equipment and other context-dependent parameters. Context, therefore, must be managed separately and its influence on the behaviour of the application must be described orthogonally with respect to the application data.

Context-aware applications must perform three tasks. Firstly they must capture low-level contextual information from different sensors, (for example, GPS coordinates). Secondly they must identify what needs to be captured in order to create high-level contextual information which is more relevant to the application. For example, GPS coordinates can be transformed into a complete address and physical, temporal and semantic relationships can be deduced from the initial low-level context values. Finally, this interpreted information must be supplied to the application. The Context Toolkit (Dey, Salber and Abowd (2001)) is one of the first context-aware architectures to include these three main steps. Sensors capture low-level context signals and present them to the context widget. Widgets use the interpreter to carry high-level context data to the application through a context server.

In some approaches the context is stored before being disseminated to build a contextual history. This step is very important for the production of high-level representations of the context. For example, in order to analyse the user's movements, information about the user's previous location may be needed. This extra step has revealed another potential requirement in context research, which is context modelling. A practical approach involves creating a reliable representation of all the aspects of the context before storing it. The different approaches to context modelling are shown below:

- The first approach stores context as a simple set of attribute/value pairs, for example, {Name="context1", User="x", Location="y", Time="t"). The Context Toolkit uses this approach.

- The second approach presents the context using RDF. The most consistent method is an extension of the CC/PP W3C profile called Comprehensive Structured Context Profiles (CSCP), proposed by Held.
- The third approach models the context using ontologies. The most consistent method is CoOL, which presents a context parameter as a set of entities having certain aspects representing its characteristics.

After modelling and storing the context, it must be carried to the application, with information about how the application can adapt to context changes. In this area, Dockhorn Costa (2003) distinguishes between four research approaches:

- Conceptual frameworks focus on the architectural aspect of context-aware systems and provide the means to facilitate capturing, interpreting and carrying context data to the interested parties, for example the Context Toolkit (Dey, Salber and Abowd (2001)) and the Cooltown (Kindberg and Barton (2001)).
- Service platforms aim to provide the pertinent services to the user depending on the context. This includes dynamic service discovery, dynamic deployment of adaptive services, addressing issues of scalability, security and privacy, for example, M3 (Indulska, Loke, Rakotonirainy, Witana and Zaslavski (2001)) and the Platform for Adaptive Applications (Efstratiou, Cheverst, Davies and Friday (2001)).
- Appliance environments aim to resolve the problem of heterogeneity by providing interoperability techniques and frameworks, for example, Ektara (DeVaul, Pentland (2000)) and the Universal Information Appliance (Eustice, Lehman, Morales, Munson, Edlund and Guillen (1999)).
- Computing environments for pervasive applications focus on designing the physical and logical infrastructure to hold ubiquitous systems, for example, the PIMA project and Portolano (Esler, Hightower, Anderson and Borriello (1999)).

The development of context-aware, adaptable applications is based on two objectives: designing an architecture which supports context awareness at run-time and designing the application itself to be context-aware. Different technologies are used to build applications. The adoption of web services, however, is widespread and is considered today to be a viable architecture for evolving applications, mainly due to its "loose coupling" approach to the integration of application functions (Austin, Barbir, Ferris and Garg (2002)).

Context capturing concerns physical sensors and the raw data they generate. This aspect is highly device-dependent and a generic model is difficult to build. In one architecture a context provider is defined that represents a context capturing system.

Context interpretation of the low-level representations which are initially captured may not be meaningful to the application, while high-level representations are easier to interpret and to use (for example, an address is more significant than GPS coordinates). In this architecture, the context interpreter module is responsible for the context interpretation.

Context modelling represents a context repository where context values are stored. In order to model context parameters, XML documents are used to store and exchange context values. They define a set of elements for each context facet (user, network, device, metadata), containing the current values of parameters relevant to that facet. Defining the set of context parameters is the job of the application designer, who also specifies their syntactic and semantic structure, (for example, scalar values, sets, references to other parameters).

Context dissemination in a context-aware application requires the use of part of the context. In a service-oriented architecture, it must subscribe to the context broker that carries the pertinent data to each service in the application. During the process of subscribing, the service tells the broker which part of the context is relevant to it. As a result, the broker can provide a context view for each service. This view can evolve dynamically during execution, requiring some intelligence in the broker. Services may pull context values each time they require context or the context broker may push context to subscribers every time it is updated.

The adaptation process is where context consumers have to adapt the context. In order to do so, they are first registered with the context broker. Context adaptation can take place on three levels: data flow (content adaptation), visualisation (user interface adaptation) and the behaviour of the application (service adaptation). All these adaptations can be static (before runtime) or dynamic (at runtime). Both methods should be used to ensure the best adaptation to the context.

An architecture of this kind is shown in Figure 3.3 below:

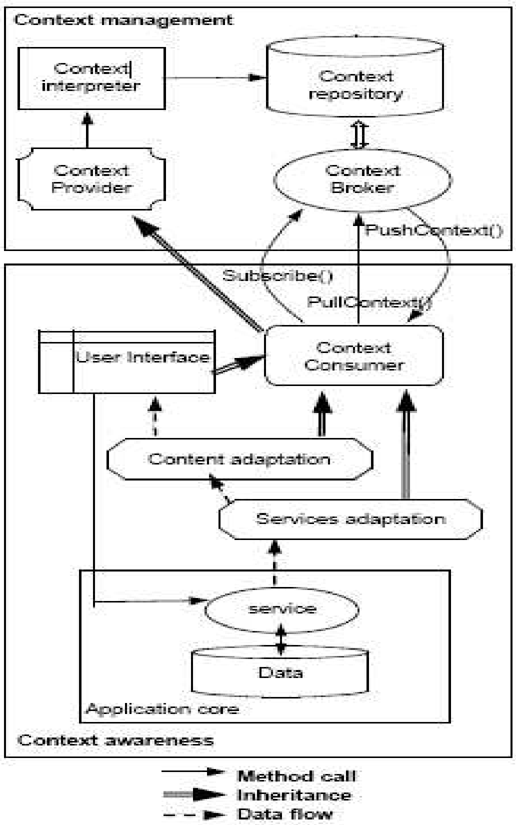

Figure 3.3: A service-oriented architecture

3.4 Adaptive Systems

An adaptive system is a system that builds an internal model of an individual user and applies it for adaptation to the user. Wu (2002) defines an adaptive hypermedia application as a system incorporating a domain model, a user model, an adaptation model and an adaptation engine. Wikipedia defines the adaptive system as a system that is able to adapt its behaviour according to changes in its environment or in parts of the system itself. A human being, for instance, is certainly an adaptive system; so are organisations and families. Some man-made systems can be made adaptive as well; for instance, control systems use feedback loops in order to sense conditions in their environment and adapt accordingly.

The entire scope of a context-aware adaptive system is shown in Figure 3.5. The computing entity can be any hardware component or software module, including:

- At application level: the user interface, application, content provision and consumption, media streaming and transcoding etc.

- At platform level: the middleware, file system, OS, database system, network transport system, resource management, Quality of Service (QoS) etc.
- At infrastructure level: the processor, memory, communication channel, transmitter and receiver, disk, I/O device, adapter, power supply etc.

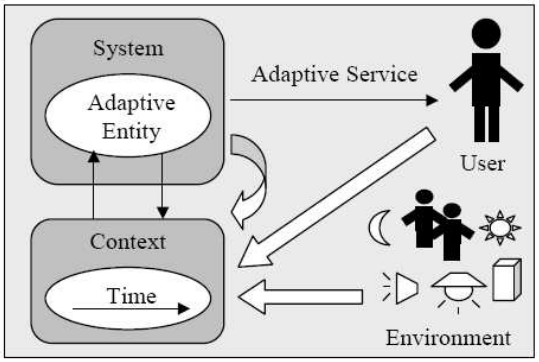

Figure 3.4: Context-aware adaptive system
(Source: Sun and Sauvola (2003))

Services provided by an entity are totally dependent on the functions and targets of the entity. Moreover, several entities can be integrated with one another in order to provide adaptive services jointly. The input to an entity can be:

- Explicit input, leading to inherent adaptation where the adaptation cannot be separated from the original functions, for example an adaptive wireless receiver.
- Implicit input, leading to additional adaptation where the adaptation is added to enhance the original functions, for example an adaptive file system.

Most of the adaptation mechanisms at infrastructure level involve inherent adaptation based on explicit input and have been integrated into the hardware design. We are more interested in the additional adaptation mechanisms adopted by entities at platform and application level, in other words adaptive software. Context information forms the implicit input set for an entity. So adaptation based on context awareness is an additional adaptation, such as the active context awareness introduced in the previous section. Figure 3.5 illustrates the basic adaptation principles and Figure 3.6 shows the relation between adaptation and context awareness in a system. Note that an entity can provide both context and adaptation. For example, in a network-aware adaptive file system, the network is the context entity and the file system is the adaptation entity, while in an energy-aware adaptive network, the energy is the context entity and the network is the adaptation entity.

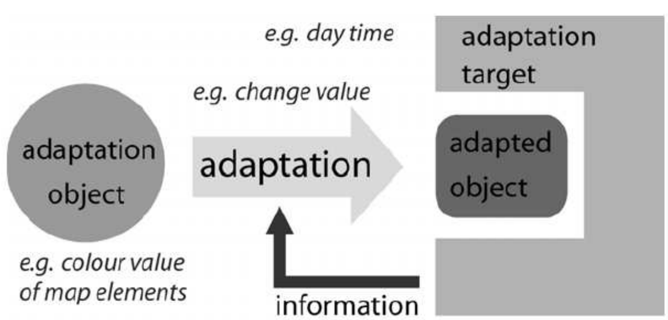

Figure 3.5: Basic adaptation principle
(Source: Reichenbacher (2007))

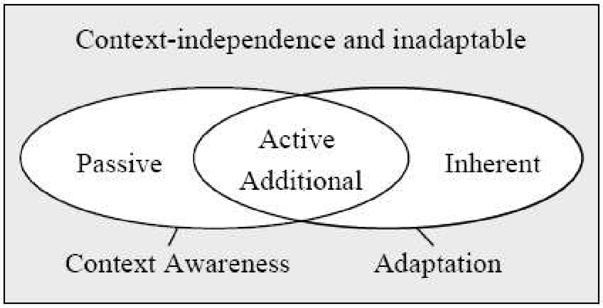

Figure 3.6: Relation between adaptation and context awareness
(Source: Sun and Sauvola (2003)

With the increase in the use of several mobile technologies, such as mobile data networks (GPRS and UMTS), the Global Positioning System (GPS), mobile phones and PDAs, it is possible to offer online map services to people wherever they are and at any time of day. Online map services could be particularly beneficial to people who are visiting places where they have never been before. This could include, for example, sales people, truck drivers and tourists. For instance, tourists often do not know which route to take in a new place, nor which restaurants, museums, shops, public services etc. are available to them. In this case an adaptive system can help tourists to find places which they are interested in and which correspond to their current situation.

Mark, Stanislar and Johan (2004) divide adaptive systems into two groups: recommender systems and context-aware systems. Recommender systems use the opinions of users in the community to help individuals to identify more effectively the content which interests them amongst a potentially overwhelming set of choices. In other words, a recommender system is defined as being capable of helping people to find their way quickly and easily through large amounts of information by determining what is of interest to a user.

A context-aware system, on the other hand, can be used to provide relevant information and/or services to the user, where the relevancy depends on the user's tasks. On the basis of these definitions, both recommender systems and context-aware systems are, in fact, used to provide users with relevant information and/or services. In the first case this is based on the user's interests and in the second case it is based on the user's context. Therefore, there is the potential for combining these two systems.

The combination of the systems has been tested in the form of the Context-Aware Mobile Personal Assistant application (COMPASS) (Mark, Stanislar and Johan (2004)) which provides tourists with information and services relating to their specific context and which are of interest to them. For example, if a tourist is interested in history and architecture, he will be offered information about a nearby monument built before 1890 or if a user expresses the wish to find somewhere to stay, he will be given a list of hotels and campsites in and around the town that correspond with his preferences for accommodation.

According to Mark, Stanislar and Johan (2004), context and interest can also be used as hard or soft criteria in the selection of relevant services. Hard criteria limit the set of available services. Those services which do not match a hard criterion are discarded from the set. Soft criteria are used to order the set of selected services or to present a relevance score to the user for each selected service. For example, location as a context element can be used to select only the services within a certain distance from the user. This is a hard criterion because it only selects a service if it is available to the user. Location can also be used to reduce the predicted relevance of a service the further away from the user the service is located. This is a soft criterion which allows the user to see a set of selected services.

DAIDALOS uses an end-to-end approach for service positioning from user to service provider and to the internet. DAIDALOS aims to design and develop the necessary infrastructure and components that support the composition and deployment of pervasive services. The Active Badge System (Want, Hopper, Falcoa, Gibbons (1992)), developed at Olivetti Research Lab, is one of the early works in the field of context awareness which redirects phone calls based on the location of the person being called. Subsequently the ParcTab (Want, Schilit, Adams, Gold, Petersen, Goldberg, Ellis and Weiser (1993)) system was developed at the Xerox Palo Alto Research Centre in the mid 1990s to support a variety of context-aware office applications. Cyberdesk (Dey (1998)) was designed with an architecture which automatically integrates web-based services on the basis of virtual context or context derived from the online world. The virtual context was the personal information that the user was interacting with on the screen, including e-mail addresses, postal addresses, dates, names, URLs etc. The Cyberguide application enhanced the current service of a guide book by adding location awareness and a simple form of orientation information.

The Ektara architecture (DeVaul, Pentland (2000)) was a distributed computing architecture for building context-aware, ubiquitous and wearable computing applications. Ektara reviewed a wide range of context-aware, wearable and ubiquitous computing systems, identified their critical features and finally proposed a common functional architecture for the development of real-world applications in this domain.

Mediacup (Gellersen, Beigl and Krull (1999)) and the TEA project (Gellersen, Schmidt and Beigl (2002)) explored the possibility of concealing context sensors in everyday objects. The Mediacup project studied the process of capturing and communicating context in human environments using infrared communication and multiple sensors based on a coffee cup. On the basis of this study, various new applications were developed, which used the context information that had been collected. The TEA project investigated technologies for enabling context awareness and their application in mobile telephony.

Other interesting examples of context management include the Owl Context Service (Ebling, Hunt and Lei (2001)), the Kimura system (Hansen, MacIntyre, Mynatt, Tullio and Voida (2001)) and Solar (Chen and Kotz (2002)). The Owl Context Service is a context system which aims to gather, maintain and supply context information to clients, while protecting people's privacy through the use of a role-based access control mechanism. It also tackles issues such as historical context, access rights, quality, extensibility and scalability. The Kimura system attempts to integrate both physical and virtual context information in order to enrich the activities of knowledge workers. Solar is a middleware system designed by Dartmouth College that consists of various information sources, such as sensors, which gather physical and virtual context information, together with filters, transformers and aggregator modifying centres, to offer the application usable context information.

The Aura project investigated how applications could proactively adapt to the environment in which they operated. A set of basic contextual services was developed within Aura in order to provide adaptive applications with environment information. Aura also focused on developing a standard interface to access services and forced all the services and clients to use the same wire protocol.

The Context Toolkit (Salber, Dey and Abowd (1999)) isolated the application from the context sensing process. The proposed architecture was based on abstract components called context widgets, interpreters and aggregators that interact in order to gather context data and disseminate it to the applications. The Context Toolkit also focused on developing an object-oriented framework and allowed the use of multiple wire protocols.

The Hot Town project developed an open and scalable service architecture for context-aware personal communication. Users and other entities were represented by mobile agents that carried a representation of the context knowledge.

The Service-Oriented Context Aware Middleware (SOCAM) project focused on providing context information to different services. A service can send and request specific context information from the locating service. The service and the locating service match the context information provided with the context information required and then a reference to the context provider is returned to the requesting service.

3.5 Adaptive Services

Adaptation services are provided by adaptation mechanisms at both application level and platform level. Figure 3.7 shows the classification for service adaptation. Three methods of providing adaptive services based on context awareness can be distinguished, in which the implementations of adaptation and context awareness are on two different levels. The three methods are as follows.

- The application-transparent method represents one extreme where both adaptation and context awareness are taken care of by platform entities including, for example, the file access system, OS, middleware, component system, network system, resource management etc.
- The application-omniscient method is the other extreme where the application is responsible both for being context-aware and for providing the adaptation mechanisms itself. No platform support is available.
- The application-aware method lies between the two extremes. Entities both at platform level and at application level cooperate to achieve the objective of the adaptive service.

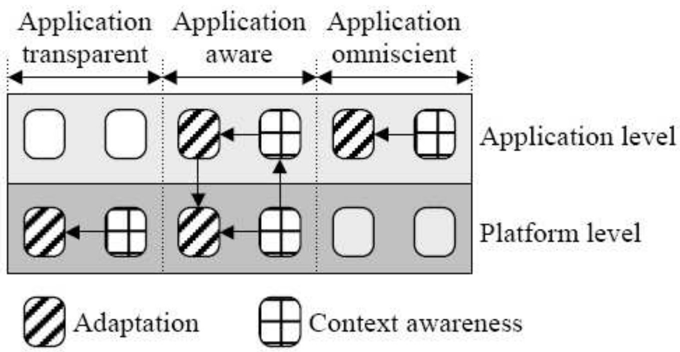

Figure 3.7: Service adaptation classification
(Source: (Sun and Sauvola (2003))

Note that when either the platform or the application is under consideration, the context awareness and adaptation functionality is not necessarily located only in the end-user's device, but may also be distributed to the entire system, including, for example, network gateways and servers.

The platform entity is good at collecting, organising and processing context information shared by all adaptive applications. As a result the OS and middleware are usually used for context organisation. The application is largely responsible for context use because it is familiar with its own requirements. The application can also collect some user interaction context for its own use.

As for adaptation, generally the application makes the final decision on how to adapt itself to various context information. While the platform is mostly responsible for performing the applications' commands, some common adaptation behaviours can also be executed there. One important adaptation mechanism is policy. The application does not explicitly control the adaptation behaviour by itself. Instead, it may express its demands in the form of policies to guide the execution of adaptation decisions at platform level. The adaptation of a service can take place in multiple serial stages, including adaptation triggering, approach selection and adaptation execution. Adaptation mechanisms are first triggered by a specific context which matches the predefined criteria. Then the decision should be made on which adaptation approach will be used. Finally, service adaptation can be achieved by automatically or manually executing a command and/or changing the external behaviours (and possibly internal states) of an entity that provides the service. Both predefined adaptation alternatives and run-time solutions could be employed. For example, in the case of a video streaming service, several versions with different resolutions can be saved on the server side in advance, each of which is intended for different capabilities of user device and network connection. Another approach is to maintain only the original (the highest resolution) version and decide which compression codec should be used for delivery on the basis of the transient context. The actual adaptation mechanisms used are closely related to and totally dependent on each specific application type.

Another important problem is the extent to which the user can be involved in the context awareness and adaptation process. This can be a very difficult problem. It is sometimes also highly critical to ensure that a user really benefits from the adaptive service. The user may sometimes want to be aware of or at least to be able find out what is going on in the depths of a system. There are also many situations where a user may even want to choose the adaptation mechanisms which he will use. In other cases, people may prefer to be totally unaware of any adaptation process and only focus on enjoying the adaptive services.

A simple example of this problem could be the different predefined profiles of a mobile phone, for example, general, silent, meeting, outdoor, customised etc. Do users want the mobile phone to change automatically to silent mode when they enter a meeting room and to return to general mode when they leave the room? Or to change automatically but alert the user? Or to remind the user to choose a new mode himself? Or perhaps the user would prefer to select a new mode without being prompted? There are two aspects to this problem: how accurate the prediction of a user's intentions can be and how visible users want an adaptive service to be.

3.6 The Adaptation Principle and Adaptation Process

Adaptation in the context of information systems refers to the ability of flexible systems to be changed by the user or the system in order to meet specific requirements. The prerequisite is that the system is adaptable, in other words that it can be changed in some way and that adaptation is in principle possible. An adaptive system is capable of changing its own characteristics automatically according to the user's needs. An adaptable system can be changed by the user in an interactive way by explicit intervention, whereas for adaptive systems the intervention is implicit. The fundamental structure of any adaptive system is based on an adaptive object that is adapted to an adaptation target using an adaptation method. The adaptation method will need information about the adaptation target in order to work successfully.

The transfer of the general adaptation concept to the field of mobile cartography has been investigated by Reichenbacher (2004). Figure 3.8 shows the general framework of mobile cartography. In the mobile usage process, geospatial information is demanded by the user and visualised via a user interface. These three items constitute the objects that are adaptable to the adaptation target, in other words the mobile usage context, by applying different adaptation methods. The adaptation objects will be analysed in more detail below and the adaptation methods will be discussed in the next section.

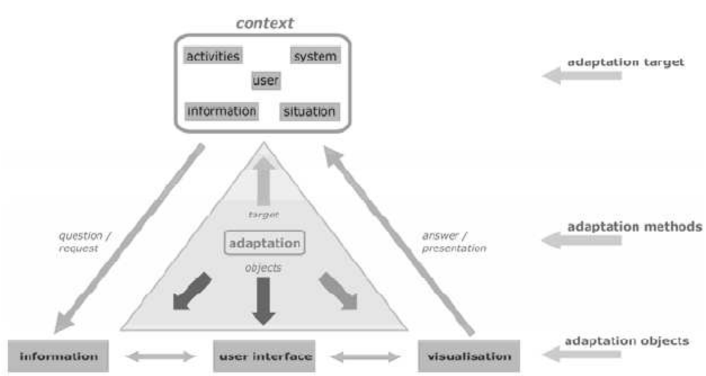

Figure 3.8: The general mobile cartography framework with an adaptation component
(Source: Reichenbacher (2004))

3.7 Adaptable Objects in Mobile Map Services

In mobile cartography the set of objects that can be adapted is extensive. Examples of adaptable objects with possible values which have been broken down into global (for example, map style) and local (for example, symbol size) aspects are shown in Figure 3.9.

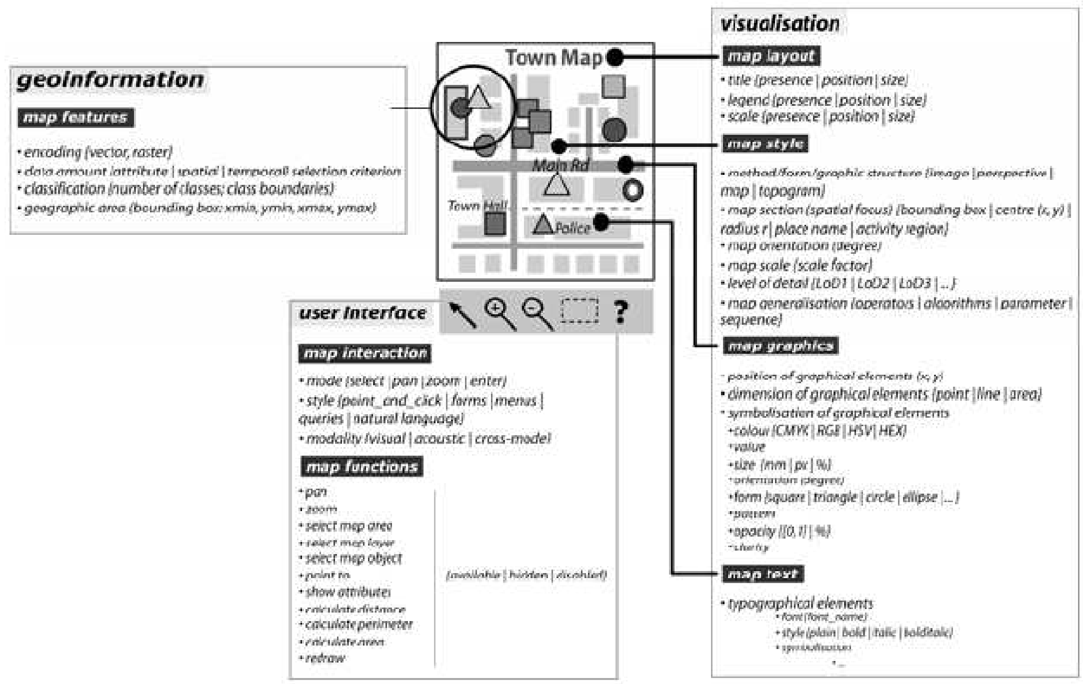

Figure 3.9: Adaptable objects in maps
(Source: Reichenbacher (2004))

The user interface is basically determined and constrained by the device in use. For instance, a PDA with a touch-sensitive screen allows for different types of interaction than a smartphone with a keypad. Therefore, the interaction style probably needs to be modified. The availability or granularity of the interactive map functions could also be adapted. Certain functions need to be hidden or aggregated to create more general functions. Furthermore, the interaction mode can be adapted, in other words the interaction mode can change depending on the current function (for example, from pointer to text entry). The geographic information can be adapted in different ways. Selecting, adjusting the amount and level of detail (LoD), classifying and grouping information are all forms of information adaptation. Another aspect is the adaptation of information encoding as a result of the capabilities of devices or the constraints of the mobile network (for example, bandwidth). However, the most important factor for mobile cartography is the adaptation of the visualisation. The map section and the map scale are global objects which can be adapted in the visualisation. The visualisation method used, for example graphics or photo, 2D or 3D, photo-realism or abstraction, can also be an object of adaptation. A landmark could for instance be displayed as an abstract symbol or a small photograph. The dimension is another adaptable object in the visualisation process. As in generalisation, it refers to the dimension in which a feature is represented, for example, a city can be displayed as an area or a dot element. Last but not least, symbol parameters (graphical variables) and text attributes are adaptable. It is obvious that certain objects are constrained by or dependent on others. Not all of these potentially adaptable features are equally suited to adaptation.

A good understanding of how to use the general context concepts in the design of various adaptive applications can greatly benefit application designers by helping them to determine which context should be considered in their applications and to understand how applications adapt to context information.

According to the definition of context given above, context also includes the interactions between the three domains. The domain to which an interaction context belongs is dependent on the entity which is the main object. The interaction also includes the interrelation between entities within one domain. Time is an important dimension for all the context entities in each domain. This is due to the fact that context information is dynamic by nature. The past and the future could both be significant to the context of an entity. Figure 3.10 shows the context space, including the three domains, the interactions between the domains and the time dimension.

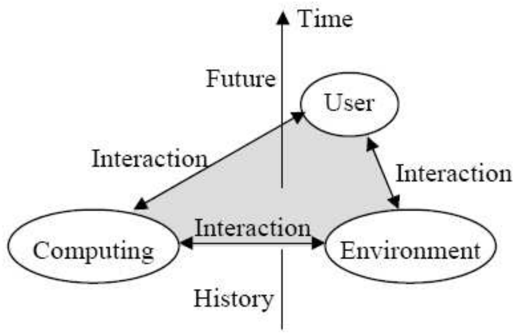

Figure 3.10: Context space
(Source: Sun and Sauvola (2003))

Two key issues are context organisation and context use. Figure 3.11 illustrates the two processes and the main operations that form part of each process. Note that not all the operations are required in any given context-aware application.

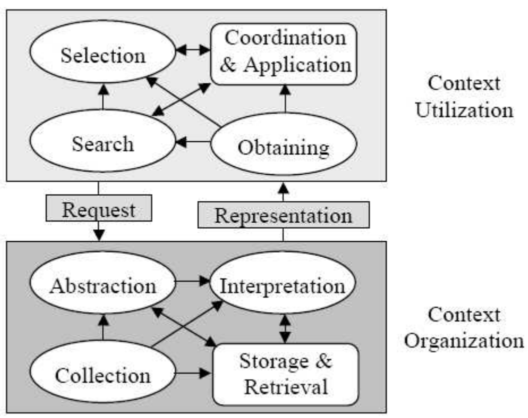

Figure 3.11: Context process
(Source: Sun and Sauvola (2003))

Context organisation is the process of finally representing the full context information. The operations include context collection, abstraction, interpretation and storage and retrieval.

The purpose of context collection is to capture the raw data in the form of the value of context features. There are three main methods for the collection of different context information, which are as follows:

- Using various sensors to obtain the environmental context and the user's physical information.
- Using software agents to collect computing context features.
- Using user profiling to gather other related personal information.

The aim of context abstraction is to obtain context information concerning external interaction features. There are two aspects of context abstraction, which are as follows:

- The raw data collected can be cleaned or calculated by, for example, sampling, averaging, statistics, calibration etc.
- The different context features need to be fused and aggregated in order to identify the correlations between them.

Abstraction methods are heavily dependent on context feature types. After the abstraction process, raw context data is translated into a format that can be understood by the application.

The objective of context interpretation is to obtain the semantics behind correlative context features. These contexts include, for example, a user's plan and future activity prediction. Interpretation can be considered as high-level abstraction that aims at understanding an activity or revealing its purpose. Basically it involves organising and representing context as knowledge. Training and previous experience is often needed in order to be able to obtain high-level context of this kind.

In the context storage and retrieval process, raw context data, as well as abstracted and interpreted context information, can be stored for later retrieval. The context should be carefully organised into various data structures, for example, tables, objects, trees, graphics etc., together with the complex referencing between elements. All these three operations can store context information, while raw context collection does not involve retrieving anything. Raw context data can be saved as a series of history records or maintained by the latest value, depending on the feature. Moreover, context storage and retrieval are necessary for abstraction and interpretation, in order to obtain history records and produce processed context information. The architecture for context storage can be either centralised or distributed. The centralised approach maintains context on a specific context server in order to be able to monitor and collect context features and provide applications with shared context information on request. A distributed architecture disseminates context across several locations, for example, application servers, network routers and end devices, without the dependency on a central server.

Context utilisation concerns the process of providing relevant context information to context-aware applications. Different applications may expect different types of relevant context to be supplied. The operations which make up the process of context utilisation include obtaining, searching, selecting, coordinating and applying context.

The purpose of obtaining context is to acquire original context from the context organisation. There are three main methods of doing this:

- Explicit query: asking for context when an application needs to initialise specific behaviours.
- Polling: fetching context periodically and updating application behaviours accordingly.
- Event-driven: subscribing to special context events and receiving notification when they take place.

The context search involves searching related context information that cannot be obtained directly by the process of obtaining context. This is an iterative process which functions as described below.

The index for all the features of an entity is the entity's name or identity. The values of an entity's features can be used as the indices when the values denote other entities. For example, if an application needs information about which of the user's friends are in the same area as the user, firstly the user's location and list of friends can be obtained directly by means of the user's name. Then, by using each value in the feature "friend list" as the index, the context search operation will search for the location value of each friend.

Context selection chooses only the relevant context information for the use required by the application. Generally the information is selected by setting a filter with corresponding conditions and then detecting incoming context accordingly. Using the "find friend nearby" example above, when the location information relating to all the user's friends has been searched, the context selection operation can then determine which friends are actually in the same area as the user on the basis of the condition "within a range of 100m".

Context coordination and application involve coordinating the processes of all the operations for context utilisation. The tasks include defining the context to be obtained and searched, defining context selection rules and controlling the utilisation operations. Context application is the final goal of context utilisation and there are essentially two ways of using context (Meng, Zipf and Reichenbacher (2005)):

- Active context awareness: an application automatically adapts to context that it has discovered by changing its behaviour.
- Passive context awareness: an application presents the new or updated context to an interested user or makes the context available for the user to retrieve later.

Each of these application tasks can be executed manually or automatically. Active context awareness produces adaptive applications. Passive context awareness is also very interesting and produces many important context-aware applications, such as context-aware resources or service discovery.

4. Conclusion

In conclusion it is clear that mobile devices have limitations in relation to the presentation or visualisation of information, such as their small display size. For example, the representation of maps on a PDA is different to that on a laptop. The smaller the display, the more cartographic information is lost. In order to reduce the amount of information and ensure that it can be displayed effectively on a small device, a generalisation process is one possible solution. Unfortunately, cartographers and map designers should be aware that generalisation may not reduce the information content. Therefore the challenge involves conveying the entire message whilst reducing the amount of information and ensuring that the results of a generalisation process can be used by a range of different users.

In addition to research into the best solution for the problem of cartographic information, auxiliary information, the legibility of symbols and text, limited interactivity and generalisation processes, the application of adaptive systems and context awareness plays an important role. This should be taken into consideration when designing the application in order to push the correct amount of geo-information to the right person at the right time. These approaches will ensure that the user has the opportunity to specify which information he really needs for his purpose. The alternative is for the user to state his purpose and then allow the system to select the necessary data and discard unimportant details. The basic concept of this approach is that if the system can obtain sufficient data from the surroundings of the location where the map is being used and the user's current activity, then the map can be adapted to the context, so that the user is provided with the appropriate or correct amount of information which is suitable for the situation and the specific user. The fundamental principles behind context awareness are based on an adaptation service, as discussed in this report. It is essential to take these principles into account when designing the application to represent the cartographic information. Integrating adaptation and context awareness into mobile devices which are to be used to detect environmental crime will form an effective starting point for raising environmental awareness.

References & Bibliography

1. Austin, D., Barbir, A., Ferris, C. and Garg, S., 2002, Web Services Architecture Requirements, W3C Working Draft, http://www.w3.org/TR/2002/WD-wsa-reqs20020819.

2. Banavar, G., Beck, J., Gluzberg, E., Munson, J., Sussman, J., and Zukowski, D., 2000, Challenges: An Application Model for Pervasive Computing. Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom 2000), pp. 266-274.

3. Brouwer, P. and Samiei, E. S., 2003, Small Display Cartography, National Survey and Cadastre, Denmark 2003.

4. Chaari, T., Laforest, F. and Celentano, A., 2004, Design of Context-Aware Application based on Web Services, Technical Report CS-2004-5, Università Ca'Foscari di Venezia.

5. Chen, G. and Kotz, D., 2002, Solar: A Pervasive-Computing Infrastructure for Context-Aware Mobile Applications, Technical Report TR2002-421, Dept. of Computer Science, Dartmouth College, February 2002.

6. Chen, H., Finin, T. and Joshi, A., 2003, An Ontology for Context-Aware Pervasive Computing Environments, IJCAI Workshop on Ontologies and Distributed Systems, Acapulco, Mexico.

7. DAIDALOS, Designing Advanced Network Interfaces for the Delivery and Administration of Location-Independent, Optimised Personal Services, http://www.ist-daidalos.org/.

8. DeVaul, R. W., Pentland, A. S., 2000, The Ektara Architecture: The Right Framework for Context-Aware, Wearable and Ubiquitous Computing Applications, MIT Technical Report.

9. Dey, A. K., 2001, Understanding and Using Context, Personal and Ubiquitous Computing, Volume 5, No. 1, pp. 4 - 7.

10. Dey, A. K., 1998, Context-Aware Computing: The CyberDesk Project, in the Proceedings of the AAAI '98 Spring Symposium on Intelligent Environments (AAAI '98 IE), pp. 51-54. http://www.cc.gatech.edu/fce/cyberdesk/pubs/IUI98/IUI98.html

11. Dey, A. K., Salber, D. and Abowd, G.D., 2001, A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications, Human-Computer Interaction Journal, Volume 16, pp. 97-166.

12. Dockhorn Costa, P., 2003, Towards a Services Platform for Context-Aware Applications, Master's Thesis, University of Twente, The Netherlands.

13. Ebling, M., Hunt, G. and Lei, H., 2001, Issues for Context Services for Pervasive Computing, Workshop on Middleware for Mobile Computing, Heidelberg, Germany.

14. Efstratiou, C., Cheverst, K., Davies, N. and Friday, A., 2001, An Architecture for the Effective Support of Adaptive Context-Aware Applications, Proceedings of the 2nd International Conference on Mobile Data Management (MDM'01), Hong Kong, pp. 15-26.

15. Esler, M. , Hightower, J., Anderson, T. and Borriello, G., 1999, Next Century Challenges: Data-Centric Networking for Invisible Computing, Proceedings of the 5th Annual International Conference on Mobile Computing Networking (MobiCom'99).

16. Eustice, K., Lehman, T. J., Morales, A., Munson, M. C., Edlund, S. and Guillen, M., 1999, A Universal Information Appliance. IBM Systems Journal, Volume 38, pp. 575-601.

17. Frank, D., 2005, Effectiveness and Efficiency of Tourism Maps in the World Wide Web and Their Potential for Mobile Map Services, in Meng, L., Reichenbacher, T. and Zipf, A., (editors), Map-Based Mobile Services, Springer-Verlag, Berlin/Heidelberg.

18. Garlan, D., Siewiorek, D., Smailagic, A., Steenkiste, P., 2002, Project Aura: Toward Distraction-Free Pervasive Computing. IEEE Pervasive Computing, pp. 22-31.

19. Gartner, G., 2007, Development of Multimedia – Mobile and Ubiquitous, in Multimedia Cartography, Cartwright, W. E., Peterson, M. P. and Gartner, G. (editors), pp. 52-62, Springer-Verlag, Berlin/Heidelberg.

20. Gellersen, H. W., Beigl, M. and Krull, H., 1999, The MediaCup: Awareness Technology embedded in an Everyday Object, in Gellersen, H.W., (editor) HUC'99, LNCS 1707, pp. 308-310, Springer-Verlag, Berlin/Heidelberg.

21. Gellersen, H. W., Schmidt, A. and Beigl, M., 2002, Multi-Sensor Context-Awareness in Mobile Devices and Smart Artifacts, in Mobile Networks and Applications Volume 7, pp. 341–351, Kluwer Academic Publishers.

22. GiMoDig, Small Display Cartography, National Survey and Cadastre, Denmark 2003, http://gimodig.fgi.fi/pub_deliverables/D3_1_1.pdf.

23. Hansen, K. M., MacIntyre, B., Mynatt, E. D., Tullio, J. and Voida, S., 2001, Hypermedia in the Kimura System: Using Spatial, Temporal and Navigational Relationships to Support Multitasking and Background Awareness, Proceedings of the ACM Conference on Hypertext and Hypermedia, pp. 14-18, ACM Press.

24. Harrie, L., Sarjakoski, L., Lehto, L., 2002, A Variable-Scale Map in Small-Display Cartography, http://gimodig.fgi.fi/pub_deliverables/ISPRS2002-Harrie-Sarjakoski-Lehto.pdf

25. Held, A., Buchholz, S. and Schill, A., 2002, Modeling of Context Information for Pervasive Computing Applications. Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI2002), Orlando, FL.

26. Indulska, J., Loke, S. W., Rakotonirainy, A., Witana, V. and Zaslavski, A., 2001, An Open Architecture for Pervasive Systems. Proceedings of the 3rd International Working Conference on Distributed Applications and Interoperable Systems (DAIS 2001), Kraków, Poland, pp. 175-188.

27. Kindberg, T. and Barton, J., 2001, A Web-Based Nomadic Computing System, Computer Networks, Elsevier, Volume 35, pp. 443-456.

28. Kolbe, T., 2004, Augmented Videos and Panoramas for Pedestrian Navigation, in the Proceedings of the 2nd Symposium on Location-Based Services and TeleCartography, 2004.

29. Kopczynski, M., 2003, Localization with Sketch Based Maps, in LBS & TeleCartography, in Gartner, G. (Editor), Geowissenschaftliche Mitteilungen, Volume 66, pp. 117-123.

30. Mark, V. S., Stanislar, P. and Johan, K., 2004, Context-Aware Recommendation in the Mobile Tourist Application COMPASS, in Nejdl, W. and De Bra, P. (editors), AH 2004, LNCS 3137, pp. 235-144.

31. Meng, L., Zipf, A. and Reichenbacher, T., (editors), 2005, Map-Based Services, Theories, Methods and Implementations, Springer-Verlag, Heidelberg.

32. Nagi, R.S., 2004, Cartographic Visualization for Mobile Application, Master's Thesis, ITC. http://www.itc.nl/library/Papers_2004/msc/gfm/nagi.pdf.

33. Reichenbacher, T., 2001a, Adaptive Concepts for a Mobile Cartography, Supplement Journal of Geographical Sciences, Volume 11, pp. 43-53.

34. Reichenbacher, T., 2001b, The World In Your Pocket – Towards A Mobile Cartography, Proceedings of the 20th International Cartographic Conference, Beijing, China, pp. 2514-2521.

35. Reichenbacher, T., 2004, Mobile Cartography - Adaptive Visualizations of Geographic Information on Mobile Devices, PhD Thesis, Institut für Photogrammetrie und Kartographie, Germany.

36. Reichenbacher, T., 2007, Adaptation in Mobile and Ubiquitous Cartography, in Multimedia Cartography, second edition, Springer-Verlag, pp. 383-397.

37. Salber, D., Dey, A. K. and Abowd, G. D., 1999, The Context Toolkit: Aiding the Development of Context-Enabled Applications, in the Proceedings of CHI'99, Pittsburgh, PA, May 15-20, 1999, ACM Press.

38. Strang, T. and Linnhoff-Popien, C., 2003, Service Interoperability on Context Level in Ubiquitous Computing Environments, International Conference on Advances in Infrastructure for Electronic Business, Education, Science, Medicine, and Mobile Technologies on the Internet (SSGRR2003w), L'Aquila, Italy.

39. Sun, J. Z. and Sauvola, J., 2003, Towards a Conceptual Model for Context-Aware Adaptive Services, in Parallel and Distributed Computing, Applications and Technologies, PDCAT’2003, Proceedings of

the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 90-94.

40. Verbree, E., Tiberius, C. and Vosselman, G., 2003, Combined GPS-Galileo Positioning for Location-Based Services in an Urban Environment, 2nd Symposium on Location Based-Services and TeleCartography, Vienna, Austria.

41. Want, R., Hopper, A., Falcoa, V., Gibbons, J., 1992, The Active Badge Location System, ACM Transaction on Information Systems, Volume 10, Issue 1, pp. 91-102.

42. Want, R., Schilit, B. N., Adams, N. I., Gold, R., Petersen, K., Goldberg, D., Ellis, J. R. and Weiser, M., 1993, The ParcTab Ubiquitous Computing Experiment. http://systems.cs.colorado.edu/~grunwald/MobileComputing/Papers/parctab.pdf

43. Wikipedia, The Free Encyclopedia, Wikipedia Foundation Inc. http://en.wikipedia.org/wiki/Main_Page

44. Zlatanova, S., Verbree, E., 2003, User Tracking as an Alternative Positioning Technique for LBS, in LBS and TeleCartography, in Gartner, G., (Editor), Geowissenschaftliche Mitteilungen, Volume 66, pp. 109-117.

Corresponding address

Delft University of Technology
OTB Research Institute for Housing,
Urban and Mobility Studies
Jaffalaan 9, 2628 BX Delft
PO Box 5030, 2600 GA Delft
The Netherlands
tel. +31 (0)15 278 30 05
fax +31 (0)15 278 44 22
e-mail: mailbox@otb.tudelft.nl
www.otb.tudelft.nl