Context-awareness in Mobile Tourism Guides

W. Schwinger[1], Ch. Grün[1], B. Pröll[1], W. Retschitzegger[1], A. Schauerhuber[2]

[1] Johannes Kepler Universität Linz, Altenbergerstrasse 69, A-4040 Linz
http://www.uni-linz.ac.at
[2] Technische Universität Wien, WIT Wissenschafterinnenkolleg Internettechnologien
Favoritenstraße 9-11/188-4, A-1040 Wien
http://wit.tuwien.ac.at

Abstract. Today's tourists expect to get personalized access to tourism information at anytime, from anywhere with any media. Mobile tourism guides provide the user with such an ubiquitous access. The prerequisite for this is the notion of customisation, requiring awareness of the applications context together with appropriate adaptation mechanisms. Currently, there is a proliferation of mobile tourism guides, proposing an unmanageable number of diverse functionalities. This paper sheds light on those approaches by identifying their strengths and weaknesses, thus providing the basis for next-generation mobile tourism guides. For this, an evaluation framework is used comprising detailed criteria for the two orthogonal dimensions of context and adaptation.

1 Introdution

E-commerce and m-commerce have dramatically boosted the demand for services which enable *ubiquitous access*. Ubiquity offers new opportunities and challenges in terms of *time-aware*, *location-aware*, *device-aware* and *personalized services* which can be achieved by using *customization*, i.e., adapting an application towards the current context (c.f. [19]). The roots of customization are manifold and can be found in user interfaces being either adaptive [15] or even intelligent and advisory [9], information filtering and recommender systems [26], adaptive hypertext and hypermedia [8] and mobile computing [2], [28]. The pre-requisite for realizing customization is that an application is aware of it's context [1]. For this, the classical user model employed for personalization purposes [24] should be generalized to a context model adding primarily environmental data in terms of *time* [23] and *location of access* [16], together with *device* [12], [31] and *network* capabilities [4].

One of the application domains particularly suited for providing ubiquitous access on basis of customization is the *tourism domain*, not least since in this way, tourists can be assisted not only in the preparatory phase of a vacation but especially during the vacation itself (cf., e.g., [13], [30], allowing access with any media, at anytime, from anywhere (cf., e.g., [7]. Such applications supporting the tourist on the move by means of location-based services are often called *mobile tourism guides*. They provide the tourist, for example, with personalized on-site tourism information about points of interest (POIs) (e.g., environmental and landscape attractions or gastronomy), or assist the tourist in organizing an individual tour. A series of such mobile tourism guides have recently been proposed, offering a wide range of functionalities with respect to context-awareness and adaptation (cf., e.g, [3], [5], [10], [14], [17], [18], [20], [22], [29], [32], [35]). This paper addresses the urgent need for identifying the strengths and weaknesses of existing approaches. For this, the paper focuses on an in-depth survey of existing mobile tourism guides providing the basis for next-generation mobile tourism guides. In contrary to other surveys like [6] this paper deals with mobile tourism guides which are *web-based*, applying a broad view on context-awareness comprising not only location and device capabilities but also personalization and other context properties like time or network. In the light of that, this paper applies an evaluation framework comprising detailed evaluation criteria for context and adaptation.

The structure of this paper is as follows. Section 2 gives a brief overview on the evaluation framework. This evaluation framework is applied in Section 3 for in-depth investigation of SEVEN mobile tourism guides. The findings and implications for further research are listed in Section 4, before concluding the paper with a short summary in Section 5.

2 Evaluation Framework

This section briefly gives an overview on the evaluation framework used as the basis for comparing the customisation capabilities of Web-based mobile tourism guides. For a more detailed explanation of the evaluation criteria it is referred to our previous work [19]. This framework is characterised by two orthogonal dimensions, comprising context and adaptation, and the mapping in between represented by the notion of customisation (cf. Figure 1). To ensure traceability of the different criteria in the evaluation of approaches in Section 3, appropriate abbreviations are used for each criteria.

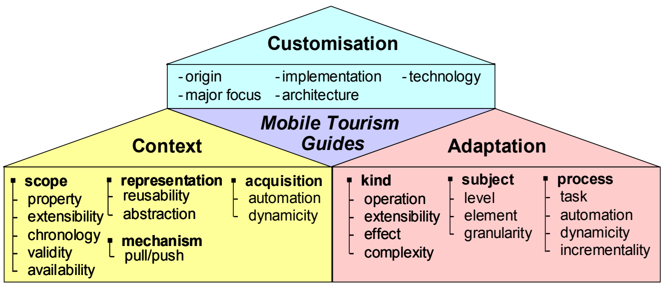

Fig 1. Evaluation Framework

In the following, the criteria for context and adaptation are outlined, resigning a description of the common criteria since they are self-descriptive. Concerning the criterion "Architecture" we distinguish between an internal customisation, meaning that customisation mechanisms are intermingled with the application to be adapted, whereas external customisation refers to a proxy-based customisation approach.

2.1 Context

Context characteristics which are relevant for the management of context data can be categorized into *scope* of context, its *representation* and *acquisition,* as well as the *access mechanism* used (cf. also [21], [25], [34]).

Scope. The *scope* of context comprises the different context *properties* supported by the system (such as location, time, device, network, and user) together with the ability to *extend* them to cope with unforeseen requirements. For each of those properties also the time dimension needs to be regarded, represented by *chronology in terms of historical, current* and anticipated *future context* along with *validity* and *availability* of context data.

Representation. The *representation* of context comprises two important issues. First, mechanisms for enhancing context *reusability* by, e.g. drawing from various context sources like GIS or existing device profiles. Second, the level of *abstraction* at which context is represented, separating physical context in form of sensed context data and logical context which is derived on bases of inference mechanisms or provided by profiles.

Acquisition. The *acquisition* of context can be characterised by the degree of *automation,* considering *who* is responsible for acquiring the context that is either a human (manual), the system (automatic) or a combination thereof (semi-automatic) and the degree of *dynamicity* in terms of *when* the context is acquired being either statically considered at system startup or dynamically at run-time.

Mechanism. The *mechanism* by which context is acquired and made accessible to the services using the context can be either *pull-based* when a request to the system is issued or *push-based* when context changes.

2.2 Adaptation

The second dimension of our evaluation framework is covered by the notion of adaptation, characterised by the *kind of adaptation*, i.e., what changes have to be done, the *subject of adaptation* in terms of what to change and the *process of adaptation* characterising how adaptation is performed.

Kind. The *kind* of adaptation subsumes built-in *adaptation operations* such as filter content, add links or change image resolution and possible *extension* mechanisms to introduce new user-defined adaptation operations into the system. Adaptation can *effect* the system in that certain parts of the system are added, removed or transformed and can be, as a series of adaptation operations, combined to *complex* adaptations

Subject. The *subject* of adaptation can be characterised by looking at the *level* of the Web application which is affected by the adaptation, comprising content level (i.e., domain-dependent data), hyperbase level (i.e., the navigation structure) and presentation level (i.e., the layout of each page together with user interaction facilities). Each of the levels contains several application *elements* that can be adapted (e.g., pages, links, access structures, input fields or media types). Finally, the *granularity* of adaptation indicates the number of application elements affected by a certain adaptation distinguishing between micro- and macro-adaptation.

Process. The *process* of adaptation comprises a number of tasks which should be separated into, e.g., initiation, proposal, selection to allow a fine grained control of their degree of *automation.* The degree of *dynamicity* defines whether the adapted versions are already available or generated at run-time [25]. Finally, adaptation can be either conducted from scratch or incrementally meaning that adapted versions are made persistent so that subsequent adaptations are conducted on basis of results of previous adaptations.

3 Evaluation of Mobile Tourism Guides

There are numerous web-based mobile tourism guides proposed (cf., e.g, [3], [5], [10], [18], [20], [22], [29], [32], [35]). In this survey we particularly focus on those web-based mobile tourism guides offering the user a map-oriented interaction paradigm because such systems are more and more entering every day live. Furthermore, we particularly chose those approaches offering unique features with respect to context-awareness and adaptation. Each of the five selected approaches is described in the following in a separate sub-section according to the evaluation framework, thus giving an overall understanding of each approach before discussing particular results of this investigation in Section 4. It has to be emphasized that due to space restrictions, this paper presents only a subset of a more comprehensive study investigating nine approaches [38].

3.1 COMPASS

COMPASS [35] – short for COntext-aware Mobile Personal ASSistant – provides tourists with context-aware recommendations and services. It uses various external map services through proprietary interfaces, such as Microsoft’s Mappoint for regular maps, a map service providing orthophotos and a map service providing cadastral maps. It builds on the open Web Architectures for Services Platform *(WASP)*[1], which supports context-aware applications based on web services. The WASP platform operates on top of 3G networks and requires a permanent network connection (e.g. GPRS). Realizing external customization, COMPASS uses a r*egistry* that contains information about the third party services providing the content such as museums and restaurants information. For service description, semantic web technology such as OWL is used.

The system retrieves and provides information about the user’s context by contacting the appropriate context services (C.R). Location context (C.P) is considered as the primary criterion to select relevant services in the near surroundings of the user. User context (C.P) comprises a manually provided profile, which is described using an extended P3P specification indicating the user's interests that is further automatically (C.Au) updated by the system based on the user’s feedback for specific POIs. New context properties like weather or traffic information can be

[1] WASP Project. http://www.freeband.nl/kennisimpuls/projecten/wasp. 2005.

incorporated into the system via web services (C.Ex, C.V). The time context (C.P) is considered in the following way: The more recent the user has been at certain POIs, the lower is the predicted relevance that the user wants to visit a POI of that class (C.C). Arbitrary logical context, for example, whether a user walks or drives can be derived on bases of the given current speed and the geographical properties by domain specific rules (C.Ab) not mentioning the underlying rule technology. A subscription mechanism is offered so that components of the system can be notified as soon as the context changes (C.M). The location information is obtained automatically (C.Au) through GPS or from the mobile network (e.g. GPRS, UMTS). Changing context both for physical as well as for logical context is considered dynamically (C.D). Validity is not explicitly supported (C.V).

The adaptation operations comprise filtering the services provided by external services based on the user profile and location (A.O, A.G). The map and the POI symbols are dynamically (A.D) and automatically (A.Au) updated as soon as the context changes. The adaptation process comprises several successive steps (A.C) such as filtering content, making recommendations as well as displaying the result on the screen. All three levels are subject of adaptation. At the content level this comprises elements such as text, images, links and maps (A.El), which are added or removed according to the user's context (A.Ef). At the hypertext level, links of the POIs, which are relevant to the user's location and preferences, have to be generated. The presentation level is affected by displaying the map with the user's position as well as the POI symbols. The user can not engage in the adaptation process (A.T). Adaptation extensibility is not provided (A.Ex). The adaptation process is done fully automatically (A.Au) and non-incrementally (A.I).

3.2 CRUMPET

CRUMPET [29], [36] is a EU project aiming at the "Creation of User-friendly Mobile Services Personalized for Tourism" relying particularly on agent technology. The user can request information and recommendations about tourist attractions, restaurants and tours. The system provides pro-active tips when the user gets near a sight that might be of interest, supports interactive maps showing the position of the user as well as interesting sights (e.g. relying on OGC[2] standard interfaces such as the Web Map Service). Its architecture foresees an external customisation approach allowing the integration of various service and content providers via a dedicated interface.

CRUMPET considers location, device, network, and user context properties (C.P). With respect to extensibility, it is not foreseen to incorporate further context information (C.E). Physical location context is provided in form of GPS sensor coordinates. However, no information is provided about how the other physical context properties are acquired. Transformation of physical location context to logical location context encompasses, in a first step, sending only "relevant" location changes to the server and based on that, utilizing geo-coding services to infer addresses from the given coordinates in a second step. The logical device context is taken into account in terms of the device type determining, e.g., size of display, colour depth,

[2] Open Geospatial Consortium. http://www.opengeospatial.org. 2005.

and raster vs. vector graphics capability. The logical network context is considered with respect to the quality of networking service (QoS) and the type of wireless connection that is available (WLAN vs. GSM). The user's interests representing the logical user context are learned dynamically (C.D) by tracking user interaction, thus taking into account the history of user context (C.C). Availability issues of context are not considered in CRUMPET (C.Av). The representation of logical context is different for the various context properties supported. Logical location context is represented explicitly in a GIS storing geographic data of the region. The logical user context is based on a domain taxonomy of tourism related services and a probability function of user's interests in that services (C.Ab, C.R). An explicit representation of logical device or network context is not stated. All context properties are acquired dynamically (C.D) and in an automatic way. The user context additionally can be accessed by the user and augmented manually (C.Au). As long as few data about the user's interests are available because little user interaction with the system has occurred, the system uses stereotypes to avoid unreliable information. Validity of other context properties is not accounted for (C.V). Context information is accessed both pull-based and push-based as soon as the user approaches a certain POI (C.M).

Approach	*Origin*	*Major focus*	*Architecture*	*Implementation*	*Technology*	*Application*
COMPASS	context-aware recommendation system	integrating context-aware and recommender systems in mobile tourist applications	internal/ external	implemented	Web Service Technology, XML, OWL, P3P	context-aware tourist guide
CRUMPET	mobile services personalised for tourism	supporting nomadic users with services	internal/ external	implemented	Agents	location-aware tour guide
GUIDE	location-based services	context-aware generation of tourist information	internal	implemented	HTML meta-tags	location-aware tour guide
Gulliver's Genie	artificial intelligence and agent systems	agent-based, intelligent content delivery	internal	implemented	Agents, Java	location-aware tour guide
LoL@	application development for restricted devices, i.e. GSM / UMTS phones	UMTS application	internal	implemented	Java, Java Applets, XML, XSL	location-aware tour guide
MobiDENK	location-aware system	location-aware system for historic sites	internal	implemented	Java	tour guide for historic sites
m-ToGuide	location-based services	promoting use of 2.5/3G cellular networks	internal	implemented	Microsoft's .net framework, Microsoft's C#	context-aware tourist guide
PinPoint	context-aware web applications	making web-pages context-aware	internal	implemented	HTML meta-tags, Java	web-based tourist guide
Sightseeing4U	adaptive multimedia	dynamic creation of personalised multimedia content	internal	implemented	Java	personalised city guide

Fig 2. Overall Comparison of Approaches

Adaptations are performed by locating and querying suitable content and service providers and adapting the query outcome to the context. Although device and networking conditions are considered as context - apart from stating that results are combined into "digestible" information suitable for the user's device - there is no further information on how CRUMPET makes use of this information for adaptation purposes. CRUMPET allows to dynamically (A.D) perform a stepwise adaptation (A.C) on the results of the previous step, starting each time a service is requested on basis of the original query result (A.I). CRUMPET offers a series of adaptation operations covering sorting, filtering, highlighting and generation of new information (A.O), which cannot be extended (A.Ex). In principal, CRUMPET offers adaptation on all levels (A.L): On the content level, sorting is applied on the query results received from the content and service providers according to the user's preferences. Filtering is executed for the pro-active tips. New information is generated as personalized tour proposals are assembled. On the hypertext level, a list of links to POIs is generated dynamically (A.D) according to the user's position and interests. On the presentation level, the user's current position is highlighted on the map and objects of interest are highlighted. Thus, the adaptation operations considered by CRUMPET comprise adding, removing and transforming adaptations (A.Ef) focusing on elements such as texts, links, images and maps (A.El). The application is adapted on a macro level by adaptation operations affecting the selection of services, the computation of tour recommendations and pro-active tips as well as the adaptation towards the connection type. On the contrary, sorting the list of query results and indication of the user's current position on the map represent examples of micro adaptations (A.G). All adaptations are performed automatically (A.A). The user is given some control over the adaptation process in that pro-active tips can be enabled or disabled and the assumptions about his/her interests can be influenced (A.T).

3.3 GUIDE

The *GUIDE system* [10], [11] stems from the area of location-based services. The focus is to provide tourists with up-to-date and context-aware information about a city via a PDA. The GUIDE system is based on a client/server architecture, with a Fujitsu TeamPad 7600 used as a terminal. Based on the closest access point, the client determines the approximate location of the user and provides tourists with information about sights, a map and the possibility of creating a tour. The access points broadcast pages of information frequently accessed by users in the geographic area of the cell. The pages are cached on the end system.

Although the focus of the system is on providing location-based services, a more comprehensive logical context model in terms of profiles is provided (C.Ab), distinguishing between so-called *personal context* in terms of information about the user (e.g., preferences, current location and a history of already visited attractions (C.C)) and so-called *environmental context*, comprising information about attractions (e.g., links between nearby attractions, opening and closing times, relevance to user interests) (C.P). Thus, logical context is mainly specific to the tourism domain (C.R). There are neither mechanisms to extend the pre-defined context properties (C.E) nor inference mechanisms to automatically derive higher-level logical context (C.Ab). The current physical location context is gathered automatically (C.Au) at runtime (C.D), although the user is able to manually enter the current location in case that cell coverage is temporarily left. By providing the physical location in this way the location information can be regarded as valid (C.V). Logical context is, in principle, entered manually, certain information about the user, e.g., user interests are acquired semi-automatically based on the interaction history (C.Au). Context information is accessed in a pull-based manner (C.M). Availability of location context is regarded by informing the user about location information updates (C.Av).

Taking a look at the adaptation features of GUIDE, it is distinguished between *coarse-grained adaptation*, e.g., changing the language of the descriptions and *fine-grained adaptations*, e.g., presenting information about the current context or filtering/sorting information depending on a certain context (A.O), (A.Ef). Thus a certain adaptation cannot only affect a single but rather numerous web pages as is the case when changing the language or when generating a complete guided tour (A.G). In particular, the subject of adaptation comprises all three levels (A.L), focusing on text and link adaptations without changing the modality (A.El). Adaptations can be complex, e.g., location-based filtering can be followed by a sorting operation before presenting the adapted web page to the user (A.C). Adhering to an integrated architecture, adaptations are realised by web pages intermingling with proprietary HTML meta tags which allow to query the context (e.g., determining the user's interest in that particular attraction which has a certain historical value associated) and to perform the appropriate adaptation (e.g., insert a user's location or insert nearby attractions). Extensibility of this pre-defined tag set is not foreseen (A.Ex). In principle, there is no separation between the different tasks of the adaptation process (A.T), adaptation is done fully automatic (A.A). The tags are interpreted on the fly, thereby realising dynamic adaptation (A.D) as soon as the user accesses a context-aware web page. Concerning dynamic adaptation however, there is a separation of tasks with respect to the computation of nearby attractions. This production task is done automatically, immediately after the location context has changed, whereas the presentation itself is done upon a user's request. Finally, incremental adaptation is not supported (A.I).

3.4 Gullivers Genie

Gulliver's Genie [14], [18] is a prototype stemming from the areas of *artificial intelligence* and *agent systems* focusing on intelligent content delivery considering the tourists location and needs during their trip in a proactive manner. As tourists explore the area, they see a map with their current position and orientation all time as well as presentations of POIs in the proximity. These are pre-cached on the device over a wireless network connection. As soon as the tourist actually reaches one of them, the presentation of the certain POI is displayed on the screen including overall sight information, follow-up links as well as images and audio files. Additionally, the system provides the interesting feature to allow the user to add personal comments to the information presented, which can further be shared with other tourists. It is realised as a *multi-agent system* using Java. On the server, a DB2 database manages the multimedia-, geo-spatial data as well as the user model. The client application runs on a PDA, equipped with GPS, electronic compass and wireless communications support.

Gulliver's Genie considers location, in terms of the user's current position, the orientation and movement as well as user and device context (C.P). A GPS receiver provides the user's current location and an electronic compass provides the user's orientation automatically (C.Au). These context information is dynamically (C.D) monitored by an agent. The physical location is abstracted in that a set of GPS readings can be associated with a particular tourist attraction and that movements are only considered if more than 20 meters have been covered (C.Ab). In case of a considerable movement or change in the user's movement, e.g. from walking to standing, an update of the user's last position is requested (C.M). Another dedicated agent manages user context comprising demographic data such as age, gender, language and nationality together with an interest profile containing preferences about literature, art, etc (C.P). After once initialised by the user (C.Av) the profile which is stored in a database (C.R) is dynamically (C.D) and automatically (C.Au) updated by an observation agent. Additionally to the history of context also future context is considered by anticipating what the tourist will visit in the future (C.C). Validity of context is considered with respect to location information by performing a "Quality of Service audit" on the location sensor data received (C.V). Gulliver's Genie does not offer functionality to extend the considered context information (C.E).

	Property						Extensibility	Availability	Chronology		Validity	Reusability	Abstraction	Automation			Dynamicity		Mechanism	
Approach	location	time	device	network	user	application			history	future				manual	semi-automatic	automatic	static	dynamic	push	pull
COMPASS	✓	✓			✓		✓	✓	✓		✓	✓	✓		✓	✓		✓	✓	✓
CRUMPET	✓		✓	✓	✓				✓		✓	✓	✓		✓	✓		✓	✓	✓
GUIDE	✓	✓			✓			✓	✓		✓		✓	✓	✓			✓		✓
Gulliver's Genie	✓		✓		✓				✓	✓	✓	✓	✓		✓	✓		✓	✓	✓
LoL@	✓							✓			✓	✓	✓	✓		✓		✓		✓
MobiDENK	✓							✓	✓		✓	✓	✓			✓		✓	✓	✓
m-ToGuide	✓	✓		✓	✓			✓	✓		✓	✓	✓		✓	✓		✓	✓	✓
PinPoint	✓	✓	✓					✓				✓	✓	✓		✓	✓	✓		✓
Sightseeing4U	✓		✓		✓		✓	✓			✓	✓	✓	✓		✓	✓	✓		✓

Fig 3. Comparison of Context Characteristics

The adaptation offered comprises several steps (A.C). An electronic map is displayed on the PDA with the user's position and orientation highlighted. Based on the prediction of the user's future location and personal preferences a new multimedia presentation is assembled for each POI the user is currently near (A.O) (A.Ef). Whereas the generation of the sight information affects the content level (A.L), the adaptation of the follow-up links affects the hypertext structure (A.L). Presentation concerns the adaptation with respect to the client's device (A.L). The sight information of the POIs in proximity is pushed automatically (A.Au) to the client for pre-caching and is kept up-to-date dynamically (A.D) which is presented on approaching one of these POIs requiring no user input (A.T). Since the presented information is always assembled from scratch (A.I), adaptation takes place on a macro level (A.G) affecting all content data including text, audio, pictures, links, maps, etc. (A.El). Introducing new adaptations operations is not possible (A.Ex).

3.5 Lol@

LoL@ [3], [27] – short for Local Location assistant - is a research prototype of a location-based mobile application for GPRS/UMTS, providing tourists with multimedia tourism information about the city of Vienna. Its main focus is on supporting tourists during their trip with predefined tours, information about POIs, routing functionality and multimodal interaction (e.g., speech control), based on a map. Virtual visits allow some pre-trip preparation and a tour diary allows the user to review the sightseeing tour after the trip. Textual information resides in an SQL database and multimedia content is stored in the file system of the content server. An external map server provides routing and map preparation functionality. The application is designed as client/server architecture using Java Applets and Java Servlets together with XML-technology for content preparation. For communication with the server, a permanent connection via UMTS or GPRS is assumed, based on the HTTP protocol.

The system is limited to location context (C.P). The device (C.P) is not considered as a context property since the application is dedicated to mobile phones only. In addition, user context is currently also not considered apart from some simple settings. It is envisioned to support pre-filtering of virtual visits based on user preferences (C.P) as well as sorting tour details according to time context (C.P), e.g. opening hours. LoL@ does not envision to provide extensibility (C.E). Physical location context is identified dynamically (C.D) on bases of the cellId or GPS coordinates and automatically (C.Au) requested from the mobile telecommunication provider or GPS sensor each time the user issues a request (C.M). In case that automatic positioning does not allow a unique assignment of a POI to the physical location or the user has just started the tour, the physical context can be supplemented with manual user input (C.Au) indicating the current street the user is located in. This hybrid approach also relaxes the issue of inaccurate location identification (C.V) and possible unavailability of location context if positioning is turned off (C.Av). Chronology of location context is not supported (C.C). Location context is made available in a reusable (C.R) and abstracted (C.Ab) way by a separate component of the system which offers functionality for location estimation and control.

Highlighting of the user’s current position on a map and a routing functionality pointing the way to the next point of interest in the tour are adaptations that are offered with respect to location context (A.O). In this way LoL@ adds new context-aware information (A.Ef) to the system in form of text, voice audio elements, images, links and maps (A.El). These are indecomposable monolithic (A.C) operations affecting the content and hypertext level (A.L). Although LoL@ provides the prerequisite for device adaptation through parameterized style sheets it demonstrates this ability for one dedicated restricted device only (A.L). LoL@ is fixed to those adaptations only and cannot be extended (A.Ex). To avoid overlapping several POIs are aggregated and represented on the map through an own symbol (A.Ef, A.L). The adaptation operations dynamically (A.D) adapt larger parts of the system (A.G) from scratch (A.I). The user can manually influence the adaptation by some basic settings (A.T). The adaptation itself is semi-automatic using automatic user positioning and user interaction to indicate the arrival at a certain route segment and initiating the retrieval of the next routing segment (A.A).

3.6 MobiDENK

MobiDENK [22], [5] – German acronym for mobile monuments - provides both navigation support and up-to-date information about POIs. The visualization of the own position as well as of the locations of the POIs enable the user to get a quick overview of the existing monuments. The map which is available either as street map or orthophoto as well as POI data can alternatively be stored on the mobile device itself or be dynamically loaded via a wireless network connection from a GIS server through an OGC Web Map Service[3]. In the latter case, active caching of content is not supported but the device can re-establish the network connection in case of a network interrupt. Information about the historical buildings is presented in multimedia form including historical images, which allow for comparing the historical with the current view. MobiDENK uses GPS as location sensing method. It is built on the modular *Niccimon* platform. The platform is implemented in Java and requires a Java VM running on the client. The platform consists of various system components, which allow for rapid development of mobile applications.

MobiDENK makes use of location context only (C.P). The location module of the *Niccimon* platform receives dynamically (C.D) and automatically (C.Au) the physical location in form of GPS coordinates by a GPS sensor device. In addition to the GPS coordinates, the location module derives from the user's physical location values for speed and movement direction as logical location context. For getting speed and movement direction the context history is considered in that GPS signals are observed over a longer period (C.C). Position quality allows for receiving valid information for the context (C.V). Location context information is further refined and encapsulated in a *Niccimon* position event (C.R), thereby gaining abstraction from the raw positioning sensing (C.Ab). If there is no valid GPS signal available, the user can determine his position by marking it on the map (C.Av). The Mediator component of the *Niccimon* Platform disseminates context in form of events to all subscribed modules (C.M). Context extensibility is not regarded (C.Ex).

The application's adaptation operations comprise the displaying of a map with both user position and POI symbols (A.O). The varying quality of the user's location is addressed by adapting the position icon's visualization corresponding to the changing quality of the location information (A.Ef) thus making the location context's validity visible for the user. To support user orientation, the path of the user walking around in the environment is visualized by the system (A.Ef, A.C). POIs are grouped along different topics, specified by the system which can be selected in turn by the user (A.T). Apart from that, the user has no influence on the adaptation process, which is dynamic (A.D) and initiated automatically (A.A). The adaptation operations comprise adding, removing and transforming (A.Ef) focusing on elements of the content level such as text and images (A.El) and of the hypertext level (A.L) such as hypertext documents (A.El). MobiDENK supports both micro- as well as macro-adaptations (A.G). In principal, a POI can use any available multimedia format for its presentation (A.L). Adaptation is done in a non- incremental way (A.I).

---

[3] Open Geospatial Consortium. http://www.opengeospatial.org. 2005

3.7 m-To Guide

The *m-ToGuide* prototype [20] has been developed within a European IST project promoting the use of 2.5/3G cellular networks with location-based services. Being explicitly designed for city travellers, m-ToGuide directs the tourist via GPS and offers location-specific multimedia information about POIs. m-ToGuide also allows for transactions (e.g. buying a ticket) by integrating external service providers. The system offers the user both street and picture maps, but it is not stated if an OGC service is used. Furthermore, it provides not only navigation support while *on tour* but also *pre-planning* a sightseeing trip by offering the tourist virtual tours of e.g. museums and *after tour support* by offering a personal diary function that allows the users to record personal impressions. The prototype is implemented as a *standalone thin client* using Microsoft’s .NET framework. It runs on a specific handheld device using GPRS network connection. XML technology is used to integrate the content from different content providers and exchange content between the client and server.

m-ToGuide considers location, network and time context as well as user context (C.P). Physical location information is either derived automatically (C.Au) from a GPS device and continuously updated (C.D) or entered manually (C.Au) by selecting the current position on the map, if the sensed location is incorrect or not available (C.V), (C.Av). Concerning logical context information, the system assigns the user a default personal profile (C.R, C.Ab) according to the kind of trip (e.g. family trip, business trip) the user indicated to be interested in. By mapping the user’s specific travelling purpose to the preferences in the profile, those POIs the user might be interested in are selected. These preferences are dynamically (C.D) and automatically (C.Au) updated by tracking user behaviour (C.C) or further refined through user input (C.Au). Availability of location and network context (C.Av) is regarded by informing the user about location information updates and the current network status. The time context which is entered manually (C.Au), is used by the system for planning the length of a tour and opening hours are used to inform the user automatically (C.Au) when a POI is visitable. Finally, context is accessed both in a pull-based and in a push-based manner, in that location-based notifications are shown if the user approaches a POI (C.M). The context can not be extended (C.E).

Adaptations include filtering of content, making tour proposals, creating navigational routes and highlighting (A.O). The filtering adaptation operation is used to deliver information about POIs dynamically (A.D) in form of proactive tips. It affects the content level (A.L) by adding or removing (A.Ef) content elements such as text, audio, pictures, links and parts of the maps (A.El). The user can activate a tour (A.T), which is generated by the system (A.D) but allows for some refinement such as removing certain POIs. Therefore, also the hypertext level is subject to adaptation since POI symbols are links to more detailed information (A.L). The route is then recalculated and saved as a new personal tour (A.Ef), which is a complex process (A.C). In addition, the user can make further adjustments (A.T), amongst them selecting an interface or content language thus allowing for adaptations on a macro level (A.G), or toggling street names on and off, which takes place at the presentation level (A.L). Route calculation takes into account the means of transportation, e.g. by foot or by bus. Highlighting is a feature, which is used to display POI information, the user’s position and the route on the map and affects the presentation level (A.L). The adaptation process (A.C) is conducted automatically (A.A) giving the tourist no other influence on the adaptation process than by the few user preferences (A.T) mentioned above. The extension of built-in adaptation operations is not foreseen (A.Ex). Adaptation is done from scratch (A.I), since the system has to recalculate the information displayed on the device each time one of the context properties changes.

Approach	Operation	Extensibility	Effect			Complexity		Level			Element						Granularity		Tasks	Automation			Dynamicity		Incrementality
			add	remove	transform	simple	complex	content	hyperbase	presentation	text	audio	image	video	link	map	micro	macro		automatic	semi-automatic	manual	static	dynamic	
COMPASS	✓		✓	✓			✓	✓	✓	✓	✓		✓			✓		✓		✓				✓	
CRUMPET	✓		✓	✓	✓		✓	✓	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓				✓	
GUIDE	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓				✓	✓	✓	✓		✓				✓	
Gulliver's Genie	✓		✓				✓	✓	✓	✓	✓	✓	✓		✓	✓		✓		✓				✓	
LoL@	✓		✓		✓		✓	✓	✓	✓	✓	✓	✓		✓	✓		✓	✓		✓			✓	
MobiDENK	✓		✓	✓	✓		✓	✓	✓	✓	✓		✓			✓	✓	✓	✓	✓				✓	
m-ToGuide	✓		✓	✓			✓	✓	✓	✓	✓	✓	✓		✓	✓		✓	✓	✓				✓	
PinPoint	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓		✓		✓	✓		✓		✓				✓	
Sightseeing4U	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓				✓	

Fig 4. Comparison of Adaptation Characteristics

3.8 PinPoint

PinPoint [32] is a framework for developing context-aware web applications based on an external architecture. A web-based mobile tourism guide is implemented as a prototype using the framework. PinPoint is based on a client/server architecture, which assumes a permanent wireless connection. The client hosts a web proxy and a so-called *Context Manager*. The proxy observes the data streams between browser and server and replaces dedicated tags with the corresponding context information. Context data is provided by the Context Manager, which collects relevant context data such as user information or location. It is completely decoupled from the Web application and runs as a background process on the client. In addition, the Context Manager can query an external *Semantic Server* for geo-coding functionality. It can be looked up via a service discovery protocol, via broadcast messages or with the help of the DHCP protocol. The server runs as servlet providing for map calculations and storing the GPS coordinates of the POIs as well as the URLs of the corresponding web pages. The prototype implementation runs only on devices which support full Java. It is envisioned to transfer the PinPoint system to small systems supporting only Java Micro Edition. Each time, the browser requests or refreshes the map, a URL with some context-tags as parameter, e.g. the GPS position, is sent to the server. As a result, the browser displays a map of the user's near area as well as a list of interesting sights in the local area. The system provides zooming functionality such that the map can be displayed at two different levels of detail. On this map the user's current position is visualized.

The Context Manager considers the context properties location, time and device (C.P). Physical location information is provided in form of GPS coordinates sensed by the Context Manager. Precision of the GPS signal is mentioned to be measured (C.V), but not further dealt with. The application can query a Semantic Server converting geographical coordinates into logical locations and vice versa (C.Ab). If no GPS signal is available, the coordinates may be derived from an alternative positioning system, if necessary, with the help of the Semantic Server (C.Av). The time context is considered in form of the current date, the time of day as well as additional information to the location context thus making available the age of the last measured location. Device context provides information about the type of device, the screen resolution and the capability to display colours. User context is considered only in form of login name and email address thus it is used rather for authentication purposes than for adaptation. It is not stated which technology is used for storing these context properties. The context properties location and time are considered dynamically (C.D) and provided automatically by the context manager (C.Au) whereas device and user information are determined in a static way (C.D). Context chronology (C.C) is not regarded by the system and there is no possibility to add further context properties (C.E). Context information is accessed in a pull-based manner either by the user making a request or by a JavaScript procedure refreshing the map regularly to visualize the current location while on the move (C.M). Since all context information is hosted by the Context Manager, it can be used by arbitrary web applications (C.R).

Adaptation is processed by analysing the context tags, which are either attached to a HTTP request or integrated in web pages and by performing the appropriate adaptation. The adaptation operations include displaying the current user's position on a map and presenting a list of interesting sights in the area correlating with the user's location (A.O, A.C). The subject of adaptation comprises all three levels (A.L), thereby focusing on elements such as text, images, links and maps (A.El). First, the POI data is filtered (A.Ef) according to the user's location (A.L). Second, the links leading to pages with more information about the POIs are composed (A.L). Third, the presentation is adapted according to the device's capabilities (A.L). In case of mobile devices with a small screen size, the amount of information is not only reduced, but a new map, suitable for the small screen, is generated (A.Ef). Therefore, adaptation can be complex (A.C) and concern various parts of the web application (A.G). Adaptation tasks are not separated (A.T) and the adaptation process is done automatically (A.A) and non-incrementally (A.I). The tags are processed on the fly as soon as the client makes a request or a page is transferred from the server to the client (A.D). Adaptation tasks are not supported (A.T). Extensibility of adaptation is not envisioned (A.Ex).

3.9 Sightseeing4U

Sightseeing4U [33] is a prototype[4] of a personalised city guide. Its foundation are the *mobileMM4U* framework and the *Niccimon* [BBK+04a] platform. The *mobileMM4U* framework provides generic components for developing applications providing the dynamic creation of personalized multimedia content for (mobile) devices. The *Niccimon* platform provides functionality for location-aware services and consists of several modules offering mobile navigation and orientation support, a multimodal user interface and location-based information and services. It supports tourists during their trip by offering a street map of the city of Oldenburg with thematic overlays, such as clickable POIs, containing information in form of text, video and audio. Map functionality such as panning or zooming is not supported. Sightseeing4U is based on a client/server architecture with a mobile device acting as client that receives up-to-date multimedia presentations from a server over a wireless communication network and renders them using a software player for standard multimedia formats. It is possible to pre-generate multimedia presentations on a desktop PC and copy them to the mobile device so that they can be used as a fallback during network connection losses. Parts of the framework could also be executed on the mobile device allowing to generate multimedia presentations using the minimal media assets previously copied to the mobile device.

Sightseeing4U especially considers the user and device context (C.P). Both are acquired once (C.D) at application start-up through a dialog which also can be used for later refinement (C.Au), (A.Av), (C.V). Concerning personalisation, users can state their interests by selecting several thematic layers, e.g. churches, museums and choosing their preferred language. Device context is collected by selecting output formats and target platforms from a list of different presentations. Besides user and device context, the underlying framework supports other context properties such as location and time, weather or noise information (C.P). In this respect, location context is provided by the *Niccimon* platform. This layered framework offers unified interfaces for abstracting from the actual access to context, allowing to integrate existing context profiles (e.g., CC/PP) and new context properties (C.R), (C.E). A uniform internal representation is used to make context available to the application (C.Ab). Chronology of the context is not regarded (C.C). Access to context is performed in a pull-based manner (C.M) at each user request.

The Framework provides built-in, domain-independent adaptation operations (A.O) which can be aggregated to form new complex, probably domain-dependent adaptations (A.Cx), (A.E). The subject of adaptation comprises all levels (A.L): First, multimedia elements such as text, links, video and audio elements (A.El) are selected with regard to the user's interests and device context. This is realized by matching the user's interests with meta data associated to the POIs. The composition of the multimedia elements in time and space as well as generating POI symbols as clickable links comprise the second step. Third, they are transformed into different output formats for mobile devices with respect to the mobile device's capabilities, such as the screen size. Besides HTML and FLASH, the system offers special support for the mobile versions of SMIL as well as SVG. The adaptation process comprises different

---

[4] http://www.offis.de/projekte/projekt_e.php?id=56

tasks such as scaling the map image to fit on the display of the mobile device and recalculating the positions of the POI symbols on the map. The system recognizes changes of user preferences and adapts its appearance by adding or removing POI symbols on the map. Thus, adaptations encompass adding, removing and transforming certain parts of the application (A.Ef). As for granularity, mobileMM4U supports both micro- as well as macro-adaptations (A.G). All adaptations are performed automatically at runtime (A.A), (A.Dy). Separation of adaptation tasks (A.T) and incremental adaptation are not supported at all (A.I).

4 Lessons Learned

In this section, we will briefly summarize the results of our comparison by pointing out the major issues of the approaches surveyed by reporting on lessons learned.

Tourism as a social activity not considered. Social factors are important in tourism since people tend to go sightseeing in groups. Some systems provide limited support by offering so called "friend-finders" or the possibility to add personal information to POIs and share this with other people. However, social activities comprise more functionalities than just displaying positions of personal information. Future systems should consider communication technologies, e.g. the integration of instant messaging services.

Balance between thin and thick clients problematic. Limited mobile devices, which suffer from few CPU power and memory, should act as a thin client, i.e., all application logic resides on the server-side and the client just renders the presentation through a browser. In this case, however, a permanent network connection is required which might be problematic due to well-known problems such as loss of network connection or limited bandwidth. In contrast to that, a thick client is capable of processing content data and composing a presentation on the client side, calling for appropriate computational power. When caching technologies are applied, a thick client can overcome the before mentioned network limitations. Thus, a careful balance is necessary between both approaches. Only Sightseeing4U supports both thin and thick clients.

Potential of incorporating external content not exploited. Most of the evaluated approaches are based on an internal customisation architecture, using their own repository for storing content data. Only COMPASS and CRUMPET exploit the possibilities of incorporating external content via Web Services.

OGC standards for exchanging geospatial information widely ignored. Most of the evaluated approaches receive geospatial information from a GIS server using a proprietary interface. Only MobiDENK and m-ToGuide use the OGC Web Map Service. Other OGC services (such as the Open-LS Initiative[5]) are not considered at all.

Time and Network context are seldomly used. For a comprehensive adaptation of the application to the user's situation, a full range of context properties should be considered. All approaches provide location-based services and use satellite or

[5] OGC OpenLS. http://www.opengeospatial.org/initiatives/?iid=73. 2005.

networks sensing (WLAN, GSM etc.) based technologies to determine the current user's position. User preferences play an import role, whereby often dynamic adaptation of initial user profiles is done by monitoring the user interaction with the system. As a positive effect, usability of the system is increased by reducing user interaction. Although time is an important property in the tourism area, it is widely neglected by the evaluated approaches. Only m-ToGuide uses time in an effective manner by informing the user about opening hours or planning the length of a city tour. It is surprising that also network context is hardly considered. Especially in mobile environments with network fluctuations, the network context would be very important.

Potential of combining context properties not exploited. Most approaches treat the information obtained by the different context properties independently and do not aggregate them and generate new context information. They just combine location and user context allowing for a better adaptation of the system to user requirements. For example, COMPASS uses location as the criterion to filter POI data according to the user's position. A recommendation service then assigns scores to the POIs indicating the predicted relevance of the POI for the user. A combination using more factors would enable the system to react even more appropriately to the current user's situation. For example, a combination of location, time and user would allow the system to suggest a restaurant in proximity suiting the user's needs at the correct time.

Context Chronology not widely supported. The majority of current approaches use historic contexts for updating the user profile only. Only Gulliver's Genie extrapolates future context. This way the system is capable of proactively pre-caching the appropriate content data on the user's mobile device based on anticipated user location, thereby addressing the fluctuating network connections as mentioned above.

Context availability partly regarded. Context availability allows the system to determine whether a certain context property is currently available. In a mobile, outdoor environment, location and network context are at a high risk of being not available. Some evaluated systems, e.g. m-toGuide, provide means to indicate the location, e.g. number of satellites used, and network status, e.g. network quality, by symbols on the screen. In this way, usability is improved by informing the user about the system's possible dropouts. If using GPS for determining the user's position, the accuracy of the position information can vary due to disturbances in the atmosphere or fluctuations in the satellite orbit. LoL@, for example, addresses this problem by supplementing the physical position information with manual user input indicating the current street the user is located. None of the approaches uses a combination of different location sensing technologies, such as assisted GPS (AGPS) [37].

Proprietary representation of context data. All evaluated approaches store context data in proprietary formats, preventing an exchange of context information between systems. Sightseeing4U is the only approach, which provides so-called User Profile Connectors for accessing user profile information in different formats (CC/PP, user profile DB, etc.) and stores them in an internal data model.

Varying automation of context acquisition. Automatic context acquisition is predominant for location context. In contrast, user context is usually derived in a semi-automatic way, in that the user fills out an initial profile, which is further updated by the system. Different to that, m-ToGuide assigns the user a default personal profile according to the kind of trip (e.g. family trip, business trip) the user indicated to be interested in. By mapping the user's specific travelling purpose to the preferences defined in the profile, those POIs the user might be interested in are selected. These preferences are dynamically and automatically updated by tracking user behaviour.

Push-based access to context not widely supported. All evaluated approaches employ pull technologies for accessing context data, only some of the approaches realize push-based access too.

Dynamic adaptation of guided tours not provided. Several approaches offer a "touring guide" function, i.e., presenting a pre-determined tour based on user interests. Those approaches, however, do not dynamically adapt the tour if a context change occurs. If, for example, the user chooses to interrupt the tour or the weather changes unexpectedly, the tour is not adjusted to the new conditions. Ideally the system should recommend tours more suited to the permanently changing context conditions.

Extensibility of adaptation operations is not commonly recognised. Extensibility of adaptation seems not to be an important issue. Adaptation operations are not seen as self-contained, explicit built-in operations but are somehow intermingled into the application logic. Only mobileMM4U, the framework underlying Sightseeing4U which stems from the area of multimedia offers a plug-able architecture which allows for an easy integration of new adaptation operations (e.g. a CityMap Operator).

Adaptation is mostly done in a monolithic way. None of the approaches supports incremental adaptation and the adaptation is done in a fully automatic way. Only few approaches, such as m-ToGuide allow for some very basic user intervention.

5 Conclusion

This paper presented an in-depth survey on existing web-based mobile tourism guides. The evaluation framework used particularly focused on context and adaptation criteria. It has been shown that these approaches support a wide-range of different functionality, but none of them comprehensively supports all proposed criteria. There are considerable lacks with respect to standards, reusability, extensibility and interoperability. Thus we draw the conclusion that there is an urgent need for providing a customisation framework for uniformly support a comprehensive context model with appropriate adaptation operators.

References

[1] Abowd., G.D.: Software Engineering Issues for Ubiquitous Computing, Int. Conf. on Software Engineering, Los Angeles. (1999)

[2] J. Altmann, J., Leonhartsberger, G., Pichler, M., Schwinger, W., Hofer, T., Retschitzegger, W.: Context-Awareness on Mobile Devices - the Hydrogen Approach, Proc**.** of the 36th Hawaii International Conference on System Sciences (HICSS-36), Minitrack on Mobile Distributed Information Systems, Big Island, Hawaii, January 6-9 (2003).

[3] Anegg, H., Kunczier, H., Michlmayr, E., Pospischil, G., Umlauft, M.: LoL@: Designing a Location Based UMTS Application. ÖVE-Verbandszeitschrift e&i, Springer (2002)

[4] Badrinath, B., Fox, A., Kleinrock, L., Popek, G., Reiher, P., Satyanarayanan, M.: A conceptual framework for network and client adaptation. IEEE Mobile Networks and Applications (MONET), Vol. 5 No. 4, pp 221-231. (2000)

[5] Baldzer, J., Boll, S., Klante P., Krösche, J., Meyer, J., Rump, N., Scherp, A., Appelrath, H.: Location-Aware Mobile Multimedia Applications on the Niccimon Platform. In 2. Braunschweiger Symposium – Informationssysteme für mobile Anwendungen (2004)

[6] Baus, J., Cheverst, K., Kray, Ch.:A Survey of Map-based Mobile Guides, in Liqiu Meng and Alexander Zipf (eds.): "Map-based mobile services – Theories, Methods and Implementations". Springer, Berlin, Heidelberg, New York, (2005).

[7] Berger, S., Lehmann, H., Lehner, F.: Location-based Services in the tourist industry, International Journal on Information Technology & Tourism, Cognizant, 5(4), (2003).

[8] Brusilovsky, P., Maybury, M.T.: From adaptive hypermedia to adaptive Web. Communications of the ACM (CACM), Vol. 45, No. 5, Special Issue on the Adaptive Web, P. Brusilovsky and M. T. Maybury (eds.), (2002)

[9] Carroll, J. M., Aaronson, A.P.: Learning by Doing With Simulated Intelligent Help. Communications of the ACM (CACM), Vol. 31, No. 9. (1988)

[10] Cheverst, K., Davies, N., Mitchell, K., Friday, A.: Experiences of developing and deploying a context-aware tourist guide: the GUIDE project, Proceedings of the 6th annual international conference on Mobile computing and networking, August, Boston. (2000)

[11] Cheverst, K., Davies, N., Mitchell, K., Friday, A., Efstratiou, C.: Developing a context-aware electronic tourist guide: some issues and experiences. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (The Hague, The Netherlands, April 01 - 06, 2000). CHI '00. ACM Press. (2002)

[12] Fox, A., Brewer, E., Gribble, S., Amir, E.: Adapting to Network and Client Variability via On-Demand Dynamic Transcoding. ACM Seventh International Conference on Architectural Support for Programming Languages and Operating Systems. (1996)

[13] Garzotto, F., Paolini, P. Speroni, M., Pröll, B. Retschitzegger, W., Schwinger W., Ubiquitous Access to Cultural Tourism Portals**,** Proc. of the Third International workshop on Presenting and Exploring Heritage on the Web (PEH'04), in conjunction with DEXA, Zaragoza, Spain, Aug/Sept. 2004.

[14] O'Grady, M.J., O'Hare G.M.P.: Gulliver's Genie: Agency, Mobility & Adaptivity. Computers & Graphics, Special Issue on Pervasive Computing and Ambient Intelligence - Mobility, Ubiquity and Wearables Get Together, Vol. 28, No. 4, Elsevier. (2004)

[15] Good, M.D., Whiteside, J.A., Wixon, D.R., Jones, S.J.: Building a User-Derived Interface. Communications of the ACM (CACM), Vol. 27, No. 10. (1984)

[16] Großmann, M., Leonhardi, A., Mitschang, B., Rothermel. K.: A World Model for Location-Aware Systems. Informatik, Vol. 8, No. 5. (2001)

[17] Hinze, A., Voisard, A., Location- and Time-based Information Delivery in Tourism, Proc. of the 8th Symposium on spatio-temporal databases (SSTD'2003), July 2003.

[18] Hristova, N., O'Hare, G.M.P., Lowen, T.: Agent-based ubiquitous systems: 9 lessons learnt. In Workshop on System Support for Ubiquitous Computing (UbiSys'03), 5th International Conference on Ubiquitous Computing (UbiComp), Seattle, WA, USA (2003)

[19] Kappel, G., Pröll, B., Retschitzegger, W., Schwinger, W.: Customisation for Ubiquitous Web Applications - A Comparison of Approaches. Int. Journal of Web Engineering and Technology (IJWET), Inaugural Volume, Inderscience Publishers. (2003)

[20] Kamar, A.: Mobile Tourist Guide (m-ToGuide). Deliverable 1.4, Project Final Report. IST-2001-36004 (2003)

[21] Kapsammer, E., Schwinger, W., Retschitzegger, W., Bridging Relational Databases to Context-Aware Services, Proc. of the CAiSE Workshop Ubiquitous Mobile Information and Collaboration Systems (UMICS2005), Porto, Portugal, 13-14 June 2005.

[22] Krösche, J., Baldzer, J., Boll, S.: MobiDENK-Mobile Multimedia in Monument Conservation. IEEE MultiMedia, 11(2), (2004)
[23] Kleinrock, L.: Nomadicity: Anytime, Anywhere In A Disconnected World. Mobile Networks and Applications, 1(4). (1996)
[24] Kobsa, A.: Generic User Modeling Systems. User Modeling and User-Adapted Interaction, Vol. 11, No. 1-2, Ten Year Anniversary Issue, (2001)
[25] Kobsa, A., Koenemann, J., Pohl, W.: Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships. The Knowledge Engineering Review, Vol. 16, No. 2, (2001)
[26] Loeb, S., Terry, D.: Information Filtering. Communications of the ACM (CACM), Vol. 35, No. 12. (1992)
[27] Michlmayr, E. Flexible Content Management for the LoL@ UMTS Application. Masters Thesis. Vienna: University of Technology Vienna (2002)
[28] Oppermann, R., Specht, M.: A Nomadic Information System for Adaptive Exhibition Guidance. Int. Conf. on Hypermedia and Interactivity in Museums, Washington. (1999)
[29] Poslad, S., Laamanen, H., Malaka, R., Nick, A., Buckle, P., Zipf, A.: CRUMPET: Creation of User-Friendly Mobile Services Personalised for Tourism. In: Proceedings of: 3G 2001 - Second Int. Conf. on 3G Mobile Communication Technologies. 26-29.03.2001. London. UK. (2001)
[30] Pröll, B., Retschitzegger, W., Discovering Next-Generation Tourism Information Systems - A Tour on TIScover, Journal of Travel Research, Sage Publications, Inc., Vol. 39/2. November (2000)
[31] Rodriguez, J.R. et al.: Extending e-business to Pervasive Computing Devices - Using WebSphere Everplace Suite Version 1.1.2. IBM Redbooks, International Technical Support Organisation, SG24-5996-00. (2001)
[32] Roth J.: Context-Aware Web Applications Using the PinPoint. IADIS International Conference WWW/Internet 2002, Lissabon (Portugal), IADIS Press, 3-10 (2002)
[33] Scherp, A., Boll, S.: mobileMM4U-framework support for dynamic personalized multimedia content on mobile systems. In: Proc. des Techniques and Applications for Mobile Commerce (TaMoCO) Track der Multi-Konferenz Wirtschaftsinformatik 2004, Essen, Deutschland, März 2004, 3, Aka GmbH, S.204-215 (2004)
[34] Schmidt, A., van Laerhoven, K.: How to Build Smart Appliances?. IEEE Personal Communications, Vol. 8, No. 4, (2001)
[35] van Setten, M., Pokraev, S., Koolwaaij J.: Context-Aware Recommendations in the Mobile Tourist Application COMPASS. In Nejdl, W. & De Bra, P. (Eds.). AH 2004, LNCS 3137, Springer-Verlag, (2004)
[36] Schmidt-Belz, B., Poslad, S., Nick, A., Zipf, A.: Personalized and Location-based Mobile Tourism Services. Workshop on "Mobile Tourism Support Systems" in conjunction with Mobile HCI '02. Pisa. (2002)
[37] Swedberg, G.:Ericsson's mobile location solution. Ericsson Review(4) (1999).
[38] Schwinger, W., Grün, Ch., Pröll, B. Retschitzegger, W., Schauerhuber, A.: Context-Awareness in Mobile Tourism Guides – A Comprehensive Survey. Technical Report, Johannes Kepler University Linz, IFS/TK, July (2005). [ftp://ftp.ifs.uni-linz.ac.at/pub/publications/2005/0405.pdf]