This is the published version of a paper published in *Information Technology & Tourism*.

Citation for the original published paper (version of record):

Fuchs, M., Höpken, W., Zanker, M., Beer, T. (2010)
Context-based adaptation of mobile applications in tourism.
*Information Technology & Tourism*, 12(2): 175-195
http://dx.doi.org/10.3727/109830510X12887971002783

Access to the published version may require subscription.

N.B. When citing this work, cite the original published paper.

Permanent link to this version:
http://urn.kb.se/resolve?urn=urn:nbn:se:miun:diva-16185
CONTEMPORARY ADAPTATION OF MOBILE APPLICATIONS IN TOURISM

WOLFRAM HÖPKEN,*† MATTHIAS FUCHS,*‡ MARKUS ZANKER,§ and THOMAS BEER*

*eTourism Competence Center Austria (ECCA), Innsbruck, Austria
†University of Applied Sciences Ravensburg-Weingarten, Weingarten, Germany
‡Mid-Sweden University, Östersund, Sweden
§University Klagenfurt, Klagenfurt, Austria

Mobile guides (based on PDAs, smart phones, or mobile phones) play an increasingly important role in tourism, giving tourists ubiquitous access to relevant information especially during their trip. Due to a more difficult access to mobile applications in a ubiquitous usage environment, based on time constraints, lighting conditions, bandwidth, etc., user acceptance of mobile applications strongly depends on the application adaptation to the concrete usage context. This article presents a framework for mobile applications in tourism, enabling a flexible implementation of adaptive, context-aware tourism applications. The framework especially provides approaches for user interface adaptation, content adaptation (recommendation), and interaction modality adaptation. The framework has been prototypically instantiated and evaluated in two different application scenarios, a city guide for the city of Innsbruck and a skiing guide for the ski resort DolomitiSuperSki. Both application scenarios showed high usage rates and customer satisfaction and proved the applicability and effectiveness of the presented approach for developing adaptive mobile tourism applications.

Key words: Mobile applications; Ubiquitous computing; Adaptive systems; Context-aware systems; Personalization

Introduction

The importance and proliferation of mobile applications has remarkably increased in tourism over the last years, due to increasing bandwidth and hardware capabilities (ENTER, 2007; Riebeck, Strak, Modsching, & Kawalek, 2008). Mobile applications (based on PDAs, smart phones, or mobile phones) are nowadays already used in the form of city and sightseeing guides (Garcia, Linaza, Arbelaitz, & Vansteenwegen, 2009; Tomas & Ricci, 2009), skiing guides (Haid, Kiechle, Göll, & Soutschek, 2008), biking and hiking guides (Ahlers, Boll, & Wichmann, 2008), museum or exhibition guides (Cano, Manzoni, & Toh, 2006), etc. The main reason for the importance of mobile applications in tourism and their increasing success is their ability to support the tourist during all trip phases, in particular on the move and within the destination. The tourist is...
getting more demanding concerning available information and expects access to relevant information at any time and anywhere. This new customer demand can be hardly fulfilled by traditional information distribution channels (like brochures, travel guides, information offices) or stationary information systems, but requires ubiquitous access to information systems, for example in the form of mobile applications. Mobile applications enable access to information more or less anytime and anywhere. In their role as a personal assistant, they can easily take into account user preferences and even user location in order to filter relevant information and avoid information overload, which is especially important in the on-trip phase. Based on these circumstances, mobile applications already play a significant role in tourism marketing and information distribution. With the proliferation of more powerful and especially GPS-enabled mobile devices, mobile applications will play an increasingly important role in tourism in the future.

However, ubiquitous and especially mobile access to information systems causes new challenges for application development. Time constraints (e.g., using a mobile application to search for the next bus connection while on the move), a wide range of different especially mobile devices (PDAs, smart phones, mobile phones, etc.) with differing and restricted capabilities, for instance w.r.t. displaying or entering data like screen size or keyboard, limited bandwidth, and different usage environments (lighting conditions, noise level, executed activity like walking, hiking, biking, etc.) have to be dealt with when developing mobile applications. The user expects that information is provided in a highly personalized way, adapted to his profile and preferences as well as his current situation (i.e., location, time, surroundings, etc.). Ubiquitous tourism information systems have to show a highly adaptive and personalized behavior in order to fulfill the above requirements and their acceptance strongly depends on the adaptation of their content, user interface, and behavior to all dimensions of the usage context. Thus, this article focuses on the topic of adaptive mobile information systems, such as systems dynamically adapting their content, design, and behavior to the usage context, such as the user, his current situation, and especially the used access mode and device (Dey, 2001; Subramanian & Chung, 2002). Adaptive systems intend to increase information quality, the efficiency of the human–computer interaction, and the overall usability of information systems.

The objective of the presented work is to develop a conceptual and technological framework for dynamically adapting mobile applications to the complete usage context. Based on a comprehensive adaptation model the framework focuses on the following aspects:

- **User interface adaptation**: Adaptation of the user interface and behavior of the application to the user, the environment, and device capabilities can be performed in a flexible and highly dynamic way. Based on an interface-independent application representation, the user interface and its concrete dialogs are automatically generated during runtime and the application adaptation can be performed without any changes to the application (code) itself.

- **Content adaptation**: Content (or tourism products and services) are filtered based on customer profiles and preferences. **Recommender systems** filter information by taking into account product or customer similarities as well as additional domain knowledge (Adomavicius & Tuzhilin, 2005; Ricci, Wöber, & Zins, 2005). The article will present a hybrid recommendation approach that combines several recommendation techniques.

- **Interaction modality adaptation**: The framework supports the usage of different interaction modalities, depending on the current usage context. Beside a Web-based PULL interface, making use of keyboard and screen-based interaction as well as speech input/output, the framework provides proactive PUSH technology. Depending on relevant context parameters, a rule-based PUSH service actively sends information to the user via SMS/MMS or e-mail.

The article starts with an introduction to the topic of adaptive systems and mobile services in tourism. It describes a comprehensive adaptation model, defining relevant adaptation and context dimensions, as conceptual framework for adaptive applications in tourism. The article then presents a
technical framework for adaptive mobile applications, especially providing innovative approaches in the area of user interface adaptation, content adaptation, and context-aware PUSH technology. Subsequently, the presented approaches are evaluated based on mobile applications in two different usage scenarios, a city guide for the destination of Innsbruck and a skiing guide for the ski resort DolomitiSuperski. The article closes with a conclusion on the effectiveness of the presented adaptation approaches for increasing user acceptance of mobile applications and gives an outlook to related future research topics.

Adaptive Mobile Applications in Tourism

Adaptive systems are characterized by their ability to adapt themselves to their usage context, where context can be viewed as all information to characterize the current situation in which the user interacts with the system (Dey, 2001; Subramanian & Chung, 2002). Here, adaptive systems have to be separated from adaptable systems. The latter can be extrinsically adapted, for example, by the user changing the language or the font size, in contrast to adaptive systems, which intrinsically adapt themselves.

The objective of adaptive systems is the increase of usability and usage efficiency. Individually adapting a system to the user and his usage context can increase the efficiency of the human–computer interaction. A personalization of the content and interaction flow can increase the overall usability, especially in the case of an unknown or only temporarily used system. A situation–conform interaction modality will reduce the communication effort and facilitate the intuitive usability of the system. These advantages of adaptive systems are especially important in the area of mobile applications. Their ubiquitous usage in quite different situations by different access devices and the potential unfamiliarity of the user with the concrete application, especially in the tourism domain, requires a highly personalized and situation–conform content, presentation and behavior of the application. The following section gives an overview on typical types of adaptation and corresponding existing adaptive systems (cf. Höpken, Scheuringer, Linke, & Fuchs, 2008; Kappel, Pröll, Retschitzegger, & Schwinger, 2003).

Device Adaptation

The existence of a multitude of different types of mobile devices, significantly differing in their capabilities (e.g., display size, input modalities), enforces a dynamic adaptation of mobile applications to different device types. The Monsee Web (Sendin, Lorés, & Solàa, 2002), DIWE framework (Kerer & Kirda, 2004), SmartWeb (Cserkúti, Szabó, Eppel, & Pál, 2006), and Thiede, Schumann, and Rosenbaum (2009) are examples of adaptive systems, dynamically generating a device–conform user interface.

Multimodality

Multimodal systems intend to provide the optimal interaction modality depending on the situation of the user and the user itself. For example, the multimodal tourism information system AccesSights (Klante, Krösche, & Boll, 2004) supports visually impaired users by an acoustic navigation and information output. The ISlide system (Bristow, Baber, Cross, Woolley, & Jones, 2002) enables user navigation by gesture recognition within a museum guide. The Multimedia InfoTainer (Salden et al., 2005) enables contextual user interface modality adaptation supporting different output modalities on different devices like audio output on a mobile phone or hi-fi system, video output on a television, or text on a PC/laptop or smart phone. Finally, the TravelMan mobile multimodal public transport guidance application (Turunen, Hakulinen, Kainulainen, Melto, & Hurtig, 2007) supports speech synthesis, speech recognition, a fisheye GUI, haptics, contextual text input, physical browsing, physical gestures, and non-speech audio.

User Adaptation (Personalization)

Personalization deals with adapting the content, presentation (e.g., length and style of textual information), and behavior of an application (e.g., filtering of menu items based on usage history) to the user profile (demographic data and prefer-
ences) and history. Recommender systems are a well-known type of personalized systems, concentrating on filtering content items relevant to the user. For instance, IMPS (Bristow et al., 2002) is a mobile museum guide, adapting the provided presentations of artifacts to the user age and user type. COMPASS (Van Setten, Pokraev, & Koolwaaij, 2004) is a context-aware mobile personal assistant adapting its offer to the user profile which is automatically generated due to the user's feedback. PECITAS (Tumas & Ricci, 2009) is a personalized mobile city transport advisory system for the city of Bolzano, Italy, adapting route suggestions to the user's travel-related preferences. Garcia et al. (2009) present an intelligent routing system for a personalized electronic tourist guide adapting routes to tourists' profiles and current circumstances (e.g., status of attractions, weather, public transportation).

**Location Adaptation**

Mobile applications often provide location-based services, typically providing information depending on the user’s current location. X² (Bristow et al., 2002), COMPASS (Van Setten et al., 2004), The Monsec Web (Sendin et al., 2002), CRUMPET (Poslad et al., 2001), LoL® (Anegg, Kunczier, Michlmayr, Postpischil, & Umlauf, 2002), TourGuide (Haid et al., 2008), LOCCATA (Ahlers et al., 2008), and DIWI (Goossen, Lammeren, & Ligtenberg, 2010) are examples of location-based services, providing information about the user’s current position and points of interest nearby and visualize them on a map or provide proactive information when they are reached. PECITAS (Tumas & Ricci, 2009) and the intelligent routing system presented by Garcia et al. (2009) adapt routing information to the user's current position and spatial behavior.

As stated earlier, application adaptation means adapting different aspects or parts of an application (adaptation dimensions) according to different environmental and contextual variables (context dimensions). The adaptation model shown in Table 1 defines adaptation and context dimensions extracted from existing adaptive applications as well as corresponding literature (Adorini, Arcelli, & Bandini, 2006; Ahlers et al., 2008; Anegg et al., 2002; Bristow et al., 2002; Cserkúti et al., 2006; Dey, 2001; Galinski, 2006; Garcia et al., 2009; Goossen et al., 2010; Haid et al., 2008; Kerer & Kirda, 2004; Klante et al., 2004; Liebermann & Selker, 2000; Salden et al., 2005; Schilit, Adams, & Want, 1995; Sendin et al., 2002; Tarasewich & Campbell, 2004; Tazari, Grimm, & Finke, 2003; Turunen et al., 2007; Thiede et al., 2009; Tumas & Ricci, 2009; Van Setten et al., 2004).

The columns of Table 1 represent adaptation dimensions. The dimension content is comprised of the topic (i.e., the category or characteristics of items or products), the textual representation, size, and representation of images as well as multimedia elements. The interface design of the application includes its dialogues and navigation, the layout and structure of its pages, and the supported interaction modalities. The behavior of an application includes available interactions, assistance functionality, or fault-tolerance behavior. The rows of Table 1 represent the context dimensions that are of relevance to trigger the adaptation of mobile applications in the tourism domain. The dimension time context is of high relevance for mobile applications in tourism because such applications are typically used during the trip or destination stay, where both content design and behavior of the application depend on the current time and season (e.g., season-dependent descriptions and pictures of a destination).

**Device context** is actually the dimension that most of all influences the characteristics of the user interface of a mobile application. How content is represented (e.g., text length, picture size), the dialog design, navigation and page structure, and the supported interaction modalities, as well as the behavior of the application strongly will depend on the used device—its hardware resources (e.g., size of memory), available interfaces (e.g., keyboard, screen size), network capabilities (e.g., bandwidth), and client technology (e.g., browser). The user context (Adorini et al., 2006), such as his demographic data and preferences, influences the content presented to the user as well as the interface design or behavior of the application, for example, a preferred interface modality like voice input/output for handicapped users. The spatial context, consisting of the current location (Adorini et al., 2006), the weather, and the local environ-
### Table 1: Adaptation Model

<table>
<thead>
<tr>
<th>Context Dimensions</th>
<th>Adaptation Dimensions</th>
<th>Content</th>
<th>Interface Design</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Topic</td>
<td>Multimedia</td>
<td>Language</td>
</tr>
<tr>
<td>Time context</td>
<td></td>
<td>Textual content</td>
<td>Images</td>
<td>Content</td>
</tr>
<tr>
<td>Time</td>
<td></td>
<td>2^I^D</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Season</td>
<td></td>
<td>2^D</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Device context</td>
<td></td>
<td></td>
<td>Hardware resources</td>
<td>0</td>
</tr>
<tr>
<td>Interfaces</td>
<td></td>
<td></td>
<td>Interfaces</td>
<td>0</td>
</tr>
<tr>
<td>Network</td>
<td></td>
<td></td>
<td>Network</td>
<td>0</td>
</tr>
<tr>
<td>Client technology</td>
<td></td>
<td></td>
<td>Client technology</td>
<td>0</td>
</tr>
<tr>
<td>User context</td>
<td></td>
<td></td>
<td>Demographic data</td>
<td>2^I</td>
</tr>
<tr>
<td>Preferences</td>
<td></td>
<td></td>
<td>Preferences</td>
<td>2^D</td>
</tr>
<tr>
<td>Spatial context</td>
<td></td>
<td></td>
<td>Location</td>
<td>2^I</td>
</tr>
<tr>
<td>Weather</td>
<td></td>
<td></td>
<td>Weather</td>
<td>2^I</td>
</tr>
<tr>
<td>Environment</td>
<td></td>
<td></td>
<td>Environment</td>
<td>2</td>
</tr>
<tr>
<td>Travel context</td>
<td></td>
<td></td>
<td>Purpose</td>
<td>2^I</td>
</tr>
<tr>
<td>Itinerary</td>
<td></td>
<td></td>
<td>Itinerary</td>
<td>2^I</td>
</tr>
<tr>
<td>Logistics</td>
<td></td>
<td></td>
<td>Logistics</td>
<td>1</td>
</tr>
</tbody>
</table>

0 = no occurrences in analyzed adaptive mobile systems; 1 = occurred at least once; 2 = occurred several times or key functionality of an adaptive mobile system; I = implemented within Innsbruck.mobile; D = implemented within DolomitiSuperski.mobi.
ment mainly influences the presented content (location-based services). The local environment (e.g., noise level, lighting conditions, etc.) may also trigger the interface design and behavior of the application, such as voice input/output in an outdoor environment with inappropriate lighting conditions. Within the travel context the purpose of traveling, such as the travel intention (e.g., cultural journey, recreation, or education) and the travel type (e.g., business trip or leisure trip, further divided into single trip, family trip, group trip, etc.), mainly influences the presented content. The travel itinerary enables content to be filtered based on already consumed or planned tourism services or activities (e.g., avoid repeated or overlapping activities). Finally, logistics subsumes factors like kind of available transportations, length of stay, or size of tour group, influencing for example whether a specific location within the destination is reachable during the stay.

Not all combinations of adaptation and context dimensions are equally important or meaningful, depending on the specific application or application domain.

Table 1 specifies the relevance of single combinations of adaptation and context dimensions, based on their occurrence frequency in existing mobile applications in tourism. Thus, a value of 0 has been assigned in the case of no occurrences at all, a value of 1 if the adaptation occurred at least once, and a value of 2 if the adaptation occurred several times or has been a key functionality of an adaptive system (cf. adaptive applications and corresponding literature above). Additionally, Table 1 specifies which adaptations of the adaptation space have been implemented in the course of the two prototypically implemented framework instantiations Innsbruck.mobile (I) and Dolomiti-Superski.mobi (D) (cf. section Evaluation).

Framework for Mobile Applications in Tourism

This main part of the article gives an overview on the developed framework for mobile applications in tourism, its overall architecture as well as its most important approaches: Adaptive user interfaces are a flexible approach for a comprehensive adaptation along all adaptation and context dimensions; the hybrid recommender system provides an innovative approach for filtering content based on a combination of content-based and collaborative filtering approaches and represents a specific solution for content topic adaptation as well as behavior adaptation by a conversational recommendation approach; the context-aware PUSH service constitutes an alternative form of human–computer interaction, proactively and context-dependently providing information to the user, and can be considered as a specific form of modality adaptation.

Architectural Overview

The presented framework provides basic components and services (e.g., integration and persistence layer) as well as higher level business components (e.g., recommendation service, PUSH service, etc.) and enables the implementation of concrete applications by combining, configuring, and extending provided components (Jessenitsch & Zanker, 2009b). Figure 1 gives a schematic outline of the framework and its different components. Easy and flexible extendibility concerning the integration of tourism services or products are important requirements to the presented framework. The data access objects provide a central mechanism for storing and accessing data throughout the whole framework. The data may be stored in a database (persistence layer) as well as in external data sources accessed via Web services or XML message interfaces (integration layer). The data access objects provide a simple programming language object interface and encapsulate all details not relevant for upper-level components like storage structure or location. Data connectors allow the transparent integration of different categories of data sources by granting polymorph access to different data source-specific implementations, for example for relational databases grounded on Hibernate (www.hibernate.org [17.12.2009], an object/relational persistence and query service for java). The user model service constitutes a central user modeling component that manages users’ profiles and their interaction histories (Jessenitsch & Zanker, 2009a). The data for the user model is derived from explicit user input as well as user interactions within the search & browse, recommendation, or PUSH service and can be enriched.
by user profiles from external applications. The services layer provides business services or processes in the form of Web services, to be called by components of the application layer as well as any other third-party application. The search & browse service offers functionality to either search for specific tourism services (like accommodation or events) or browse through the complete catalogue of available services and makes use of either data access objects to directly search for tourism services or the hybrid recommendation service. The hybrid recommendation service and PUSH service are described in more detail in the following sections. The components of the application layer mainly deal with the client–server communication and the handling of user sessions and workflow. The PULL component handles all end user requests and is provided as an XML-based Web application, called by a PC/laptop, mobile phone, or PDA browser. The user interface generator enables the flexible and dynamic adaptation of the user interfaces and is described in more detail in the next section. The PUSH component handles the distribution of messages to the SMS or email client application on the mobile phone, PDA, or PC/laptop, based on an email or SMS gateway application (cf. Beer, 2009). The administration & control component handles the administration and parameterization of the overall application and its different modules, e.g. configuring the recommendation service or defining distribution rules for the PUSH service. This functionality is provided as Web application as well as rich client application, such as a standalone java application (admin client).

Adaptive User Interfaces

The adaptation approach, presented in this section, enables a comprehensive and flexible adaptation of an application’s user interface (cf. Höpken
et al., 2008). In contrast to the adaptation approaches mentioned above, focusing on a limited number of adaptation and context dimensions, the presented approach covers the adaptation dimensions content (textual representation, images, multimedia items), interface design (dialogues & navigation, layout & structure, modality), and behavior (interaction) and can handle all context dimensions. The objective of the presented approach is to enable the definition and execution of user interface adaptations in a flexible way with minimal development and configuration effort. Adaptations should be dynamic, in that their definition should be separated from the application code and should be possible without changing the application itself. This is reached by a strict separation of the application content and its concrete representation and application behavior, based on XML and XSLT (cf. Hitz, Kappel, Retschitzegger, & Schwinger, 2002; Hitz & Plattner, 2003). The content and general structure of the application is provided in a neutral format, independent of the concrete user interface, and the presentation and behavior of the application is dynamically generated based on transformation instructions (by the user interface generator in Fig. 1).

Figure 2 illustrates the adaptation mechanism. Content is stored in a database in a presentation and application behavior-independent way, annotated with contextual information where necessary (e.g., text sections annotated with their suitability for specific user types). Based on specific user requests and depending on the current context, suitable content is selected from the database (see section hybrid recommender system). The content and general dialog structure together with relevant context parameters is provided in the form of an intermediary format, based on XML (iXML, cf. Listing 1). Depending on the context parameters, a series of transformations (corresponding to the single relevant adaptations of the adaptation model in Table 1 is performed, based on XML stylesheet transformations (XSLT), and the concrete user interface description is generated in the appropriate language (e.g. HTML, XHTML, VoiceXML, WML, etc.). Typical transformations are, for example, adapting textual descriptions due to user type or time of usage, selecting appropriate graphical or multimedia elements due to device capabilities or network bandwidth, or finally generating the concrete interface description depending on the output device. Separating different adaptation aspects into flexibly combinable transformation steps is necessary to efficiently support

Figure 2. Adaptation mechanism.
all adaptation and context dimensions and allow an easy system extension (e.g., the integration of new devices) and high flexibility.

Different approaches to XML-based formats for user interface descriptions have been analyzed according to their suitability as intermediary XML format, for example, the Extensible Application Markup Language (XAML), the User Interface Markup Language (UIML), the User Interface Extensible Markup Language (UsiXML, www.usixml.org [17.12.2009]), the Device-independent MultiModal Markup Language (D3ML), the XML User Interface Language (XUL), the Extensible Interface Markup Language (XIML, www.redwhale.com [17.12.2009]), and the Transformation Environment for interactive Systems representations (TERESA, http://giovе.cnucе.cnr.it/terеsа.html [17.12.2009]). Not all of those languages fulfill the basic requirement of being platform independent [XAML has been developed for Microsoft’s Windows Presentation Foundation (WPF) and is not fully operating system independent; XUL depends on the Mozilla browser or the XUL-Runner, respectively]. In general, such languages are designed for defining user interfaces in a device and interaction modality-independent way but not for dynamically adapting user interfaces to all possible combinations of concrete context characteristics in different dimensions as described in Table 1. TERESA supports the generation of specific interfaces during system implementation time which is not sufficient to support all context dimensions as mentioned above. UIML, even though supporting dynamic interface generation during runtime, does not enable to flexibly combine different transformation steps, required by the approach of a series of transformations for different context dimensions as described above. Consequently, a specific intermediary XML format is proposed by the presented framework, reusing the general idea of an abstract description of user interface elements of the languages above but supporting a more flexible and dynamic user interface generation triggered by different context dimensions.

As an example of the general approach of the intermediary XML format, Figure 3 illustrates the construction of a Web page out of several sections. The sections and their content are defined by the intermediary XML format but their exact representation (e.g., positioning) is left open and determined at transformation time when decisions of where and how to display the individual sections can be made, according to the device used or other context parameters. Figure 3 compares a usual layout structure for desktop PCs with a layout structure suitable for personal digital assistants (PDAs). Differences can not only be seen in the size of the displayed sections but also in their positioning or their appearance (e.g., panels and footer). Listing 1 shows the corresponding structure of an iXML document. The <head> element holds general information like language or title of the page, while the <body> element holds the actual content in the form of <section> elements. Based on semantic information (like the section type), content representation and layout can be adapted to specific user preferences or device capabilities. In general, the intermediary XML format provides all semantic information about elements of a dialog which are of relevance for the context-dependent adaptation. The attribute speech of the element paragraph, for example, specifies whether the paragraph has to be considered in case of speech output.

Listing 1:
<xml version="1.0" encoding="UTF-8" ?>
<page>
  <head> ... </head>
  <body>
    <section type="branding"> ... </section>
    <section type="menu"> ... </section>
    <section type="content">
      <heading>This is a heading</heading>
      <paragraph name="paral" speech="true">
        A paragraph can contain text in any combination with in-line links or in-line images.
      </paragraph>
    </section>
  </body>
</page>

Hybrid Recommender System

Recommender systems try to improve the interaction experience for users by dynamically proposing items of interest and by adapting the displayed content to make it more accessible. In general, this personalization behavior of recommender systems exploits the user context to adapt the content (cf. Table 1). Some approaches also
adapt the behavior of the application. In particular, conversational recommendation systems not only focus on the selection task of items but also on the interactivity aspects by explicitly eliciting user feedback by asking questions and proposing alternatives. Shimazu (2002), for instance, presented a system that interactively asks users for their requirements and proposes items that should be criticized by them. Critiquing signifies that users reject a proposed item and provide a reason as feedback such as “too expensive” or “too far away from the city center.” MobyRec (Ricci & Nguyen, 2007) is an example of such a mobile system that has been applied to the domain of restaurant recommendation and considers the spatial context of the user. The conversational aspect, which is built on the Advisor Suite system (Felfernig, Friedrich, Jannach, & Zanker, 2006; Jannach, Zanker, & Fuchs, 2009), takes this a step further by adapting the interaction flow based on users’ replies to questions and explicitly represented interaction knowledge that decides which questions are most appropriate. The selection of accommodations, sights, or events is realized by explicitly formulated domain restrictions. Furthermore, exceptional situations with assumedly wrong user input can be defined that lead to explicit follow up by the system. An alternate statistical approach using Markov Decision Processes (MDP) for achieving adaptivity of conversational recommender systems has been recently proposed by Mahmood, Ricci, and Venturini (2009).

Moreover, when it comes to the selection tasks, different recommendation paradigms exist. As depicted in Figure 4, a recommendation component produces a ranked list of items that is personalized for a given user. On the input side the recommendation system exploits a user profile including parameters describing specific context dimensions. Furthermore, community data, like ratings or transaction records of other users that are used by collaborative filtering mechanisms and product catalogues required by content-based recommendation techniques form additional input. Finally, explicit domain knowledge that constrains which item features can be related to which user requirements can be exploited by the knowledge-based recommendation component. For instance, users with small children should only receive recommendations for sights that are easily accessible from the city center. MobyRec (Ricci & Nguyen, 2007) is an example of such a mobile system that has been applied to the domain of restaurant recommendation and considers the spatial context of the user. The conversational aspect, which is built on the Advisor Suite system (Felfernig, Friedrich, Jannach, & Zanker, 2006; Jannach, Zanker, & Fuchs, 2009), takes this a step further by adapting the interaction flow based on users’ replies to questions and explicitly represented interaction knowledge that decides which questions are most appropriate. The selection of accommodations, sights, or events is realized by explicitly formulated domain restrictions. Furthermore, exceptional situations with assumedly wrong user input can be defined that lead to explicit follow up by the system. An alternate statistical approach using Markov Decision Processes (MDP) for achieving adaptivity of conversational recommender systems has been recently proposed by Mahmood, Ricci, and Venturini (2009).

Moreover, when it comes to the selection tasks, different recommendation paradigms exist. As depicted in Figure 4, a recommendation component produces a ranked list of items that is personalized for a given user. On the input side the recommendation system exploits a user profile including parameters describing specific context dimensions. Furthermore, community data, like ratings or transaction records of other users that are used by collaborative filtering mechanisms and product catalogues required by content-based recommendation techniques form additional input. Finally, explicit domain knowledge that constrains which item features can be related to which user requirements can be exploited by the knowledge-based recommendation component. For instance, users with small children should only receive recommendations for sights that are easily accessible and that can be visited during a short period of time.

In order to service anonymous users and short interaction cycles, the recommendation component supports all three aforementioned recommendation paradigms. Knowledge-based recommendation has the advantage that it does not suffer from cold-start problems meaning that it can provide recommendations to new or anonymous users, and that new items can also be proposed. The latter is rather important because, for instance, item categories like events obviously change frequently. The disadvantage of knowledge-based recommendation systems is that they require an initial setup of the knowledge base and do not automatically improve over time (Jannach et al., 2009). Content-

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3.png}
\caption{Different layout structures of a Web page (PC vs. PDA/smartphone).}
\end{figure}
based recommendation is necessary to support for instance similarity-based browsing activities like show me similar items. As outlined in Zanker, Jesenitschig, Jannach, and Gordea (2007), the content-based recommendation functionality accepts item descriptions in a structured as well as free text format. Finally, the collaborative filtering functionality of the presented framework is capable of exploiting explicit user requirements, like search criteria in order to identify user groups with similar behavior and recommend items that were of interest to similar peers (Zanker & Jessenitschig, 2009a). In addition, the collaborative recommendation component can use different types of user feedback in addition to explicit requirements stating the informational needs of the current user, such as click-through data on items disclosing how often users have shown interest in them, navigational patterns or any other contextual parameters. Zanker and Jessenitschig (2009b) demonstrated how collaborative filtering improves when different types of implicit and explicit user feedback are combined.

In the base version the recommendation component employs the Advisor Suite system that constitutes a constraint-based recommender (Jannach et al., 2009). However, once enough user feedback has been collected by the central user model service (see framework architecture in Fig. 1) the system can be hybridized with the collaborative recommendation component, whose accuracy improves with growing community data. Experiments conducted in Zanker and Jessenitschig (2009a) show that combining collaborative and knowledge-based recommendation components can lead to considerable improvements. For instance, a weighted hybridization strategy can merge the recommendations of two or more recommenders while in a cascade strategy one recommender refines the output of an additional one as described by Burke (2002). Both hybridization variants are implemented within the framework. Experiments on historical data indicate that a cascade strategy where a knowledge-based recommender filters out irrelevant items based on explicit search criteria and where collaborative filtering ranks the remaining items based on their popularity among similar peers achieved best results (Zanker & Jessenitschig, 2009a). An additional noteworthy aspect of the recommendation component is its ability to compose bundles of recommended items. For instance, users can be recommended sets of items consisting of different leisure activities and a nearby restaurant. Consequently, explicit constraints can be defined that restrict which items can be combined in which situations, as defined using different context dimensions (see also Höpken, 2000). This particular feature has been tested

Figure 4. The recommendation process.
with different knowledge base sizes showing acceptable computation times. Furthermore, from an architectural and technical perspective the whole hybrid recommendation framework itself is described in Jessenitschnig and Zanker (2009b), while the generic user model service is presented in Jessenitschnig and Zanker (2009a).

Context-Aware PUSH Service

Supporting different interaction types is one of the key enablers of interaction modality adaptation. In general two different interaction styles can be distinguished, namely push and pull (Cheverst, Mitchell, & Davies, 2001; Frank, 1997; Kendall & Kendall, 1999). By using push based services the user is not forced to search for relevant information himself but automatically receives the information that is of high relevance to him, thereby reducing the effort for gathering information. As it is not necessary to continuously search for new relevant data, data transfer costs can be reduced (Franklin & Zdonik, 1998), which is particularly important in the case of mobile services used by foreign tourists due to usually high roaming costs. Complementing traditional push systems with “context-sensitivity” turns it into an excellent approach to reduce both, information overload and information retrieval effort. Such a context-aware information push service enables the service provider to reach the tourist at any time and location and provide exactly the right information in the right context (Beer, 2009; Höpken et al., 2006), thus supporting content, behavior, and interface modality adaptation.

Considered from a software architectural perspective the central idea behind the design of the context-aware push service is the combination of a set of well-known approaches, such as knowledge-based systems with event notification, context aware, and recommender systems. The knowledge-based approach plays a central role within the push service by providing the fundament for the declarative, such as rule-based definition of push messages and enables to declaratively add new and modify or delete existing messages. Thus, the complete message handling process can be controlled in a pure declarative way. A common procedure when dealing with knowledge-based systems is the combination of different knowledge representation technologies, such as frames, objects, ontologies, and rules. The push service combines object-oriented and rule representation technologies. The first one represents declarative knowledge, which is the factual information stored in memory and has a static structure defined during system design time, such as context data (cf. Table 1). The latter represents procedural knowledge which is formulated in a pure declarative way using a novel rule language, based on Event-Condition-Action (E-C-A) rules (May, Alferes, & Amador, 2005; McCarthy & Dayal, 1989; Paton & Díaz, 1999). The rule execution is triggered by the occurrence of events. The event part defines a trigger for executing a rule, using subscription languages such as A-MEDIAS Profile Definition Language (Hinze, 2003). Following a user-centric approach, the condition part determines a set of users that meet the explicitly posed contextual restrictions, based on query languages like HQL. For each of these message recipients the action part is executed, such as for each user a personalized message is created according to his context and his preferences using the recommendation component (cf. section hybrid recommender system). Besides externalizing PUSH behavior from application code, supporting maintainability especially by domain experts, Event-Condition-Action rules proved their ability to flexibly define powerful PUSH rules by making use or reoccurring patterns of events, conditions or actions, separately from each other. More details about the overall rule language and examples of such rules are presented in Beer (2009).

The entire process of context-aware information dissemination considered from the perspective of a service provider as well as the tourist is summarized in Figure 5. The tourism expert creates messages by stating rules using the specific rule language CAITL, supported by a graphical wizard. In the simplified example shown in Figure 5 he creates a rule for a Good Morning Message. By stating the rule’s event and condition part all users are selected, whose current situation matches the defined restrictions, such as all users who are located in Orlando at 9:00 a.m. The appropriate message instance is derived subsequently for each of these users by using the recommendation system (cf. section hybrid recommender system),
Figure 5. Exemplification of the message creation and sending process.

computing the appropriate tourism products according to each user’s context and preferences. Finally, the message is rendered and sent according to each user’s preferred communication channel, an interface modality (cf. Table 1), such as SMS or email.

Evaluation of DolomitiSuperski.mobi

The presented framework has been instantiated and evaluated for the skiing area DolomitiSuperski in the winter season 2007/08 and 2008/09. Figure 6 shows the welcome page and demonstrates one of the most powerful mechanisms of the framework—the dynamic adaptation to the used device. The left part shows the dialog on a PC/laptop, whereas the right part shows the dialog automatically adapted to the specific capabilities and restrictions of a smart phone.

DolomitiSuperski.mobi offers the following features as pull functionality: Livecam pictures, weather information (i.e., mountain and valley temperature, wind, 1–3-day forecast), information about lifts and slopes (length of open slopes, open lifts, snow type, snowfall at mountain and valley as well as last snowfall), information about gastronomy, skiing huts, events of the day, tourist offices, ski pass offices, ski rental and ski schools, and downloadable wallpapers, as well as a Web 2.0-based community platform that helps to find and to be found by others in order to jointly perform leisure activities in the various destination areas (e.g., skiing, cross-country, nightlife, meet and eat, etc.). The SMS service provides a push-based report on snow quality and open slopes as well as a Good Morning Post offering tips of the day and a weather forecast. Table 1 gives an overview on all single adaptations covered by the mobile guide DolomitiSuperski.mobi.

During the winter season 2008/09, at www.dolomitisuperski.mobi a total number of 20,240 unique visits were registered (Fig. 7). A look on the usage patterns shows that access peaks are reg-
istered at 9:00 a.m., such as at the opening hours of the ski arena, 12:00 p.m., such as during the lunch break, and at 6:00 p.m., such as during and after the closing hours of the ski arena (Fig. 8). Interestingly, usage statistics keep up on a high level until 10:00 p.m., based on the fact that besides providing information on the ski arena itself, the mobile guide is used for information about events and gastronomy and nightlife, as well. Figure 9 shows the mostly accessed information categories or functions. A closer look at the distribution of the users’ origin (Fig. 10) shows that the mobile guide is also used by foreign visitors despite probably relatively high roaming costs, clearly demonstrating a significant willingness of users to pay for using the mobile skiing guide.

Finally, a laboratory-based evaluation took place at Innsbruck (Austria) in April, 2008. To avoid biases caused by the various devices and costs differently perceived by test persons, similar devices were provided. Thus, all test persons were asked to get familiar with the service dolomitisuperski.mobi using a Nokia E61 Smartphone before answering a series of evaluative and demographic questions. In total 207 test persons (57% male) participated in the study. Figure 11 shows the age groups of the sample. The educational level was relatively high (i.e., 28.5% graduated from college.

![Figure 6. www.dolomitisuperski.mobi—PC version versus Smart Phone.](image1)

![Figure 7. DolomitiSuperski.mobi—monthly usage statistics (season 2008/09).](image2)
Evaluation of Innsbruck.mobile

Complementary to the previous application scenario \textit{winter destination}, the discussed framework has also been prototypically instantiated in the application scenario \textit{urban destination} by implementing the mobile city guide Innsbruck.mobile. The entry page shows the modules configured for the destination of Innsbruck (Fig. 12, left). More precisely, services for providing information on “events,” “sights,” “gastronomy,” “accommodation,” and “weather” were plugged into the Innsbruck-version of the architectural framework. Additionally, a Web 2.0-based “mobile blogging” module and a mobile feedback platform were added. As a demonstration of the dynamic adaptation capabilities, on mobile phones, search results are presented in a simple list, but if the screen width exceeds a certain value, thumbnails are displayed to give a first impression of the corresponding event. The screenshot to the right shows a section of an event recommendation dialog. After answering a series of questions concerning his time restrictions and personal tastes, the user is presented three event recommendations that best
fit his profile. In order to receive personalized context-sensitive information, the user can finally sign up for a push service which actively provides her with SMS messages relevant to the current situation. Table 1 gives an overview on all single adaptations covered by the mobile guide Innsbruck.mobi.

The scene of the world’s third biggest sports event, the European Football Championship—EURO2008™—provided the ideal setting to make the application accessible to a broader group of users and evaluate the framework in a third typical application scenario, a mega event. Innsbruck was the venue for three matches of the international football tournament in June, 2008 and Innsbruck mobile was the official mobile guide of the host city Innsbruck. In order to optimally support the requirements of the EURO2008™, additional service features have been implemented: The “Daily Programme” chronologically lists events exclusively offered in the course of the EURO2008™. A “Match Schedule” provides match dates, actual team scores as well as general access information to reach the stadium. Furthermore, an “SMS service” actively informs about traffic jams and the actual score immediately after a match. Finally, a pull-based “traffic information service” was implemented offering traffic notifications in real-time as well as an interactive route-planner for public busses (i.e. “Bus-station Live”) and trains (Piazzolo, Fuchs, Höpken, & Promberger, 2009).

The prototypicaly implemented tourist guide
Innsbruck.mobile is online since January, 2007. Although no extensive marketing activities have been undertaken, an average of 600 distinct test users accessed the application each month. By contrast, in the course of the EURO2008 in June, 2008, the number of distinct visits increased up to almost 3,200, thereby opening 33,268 pages. Figure 13 shows the mostly accessed services. To sum up, during the EURO2008 in June, 2008 the official city guide Innsbruck.mobile showed satisfactory usage rates up to 200 unique users per day.

Conclusion and Outlook

This article presented a conceptual and technical framework for adaptive mobile applications in tourism. The suggested framework enables a dynamic adaptation of Web applications along adaptation dimensions and triggered by context dimensions, both defined within an underlying adaptation model (Table 1). The framework supports (a) user interface adaptation, such as adapting the user interface and behavior of the application to the user, the environment and device capabilities by an interface-independent application representation, (b) content adaptation, by providing a hybrid recommender system, combining different approaches to filter content based on product and customer similarities as well as additional domain knowledge, and (c) interaction modality adaptation, combining a PULL-based user interface with a rule-based adaptive PUSH service, proactively sending information to the user via SMS/MMS or e-mail.

The framework for adaptive mobile applications, containing the functions described above, has been implemented and prototypically instantiated in different typical tourism application scenarios namely a mobile city guide for the city of Innsbruck, a mobile event guide for the EURO‘08 football championship as well as a mobile skiing...
guide for the ski arena DolomitiSuperski. In all three application scenarios the framework proved its capability to flexibly and effectively generate a concrete mobile application based on application specific requirements and circumstances. All three core approaches, described above, showed their ability to increase user acceptance and have been clearly identified as important if not indispensable functionalities of mobile applications across different tourism settings. High usage rates and user satisfaction especially in the application scenario mobile skiing guide demonstrate the ability of adaptive and context-aware mobile applications to reach an adequate user acceptance in a real-world setting. Furthermore, high usage rates of foreign users show a clear willingness to pay for mobile services (here in the form of roaming costs).

The adaptation mechanisms, incorporated in the presented application framework for mobile tourism services, concentrate on adapting (a) content topic to time, demographic data, user preferences, purpose of travel and itinerary, (b) textual content, images, and multimedia content to the complete device context, (c) dialogues and navigation, and layout and structure to the device context, (d) interface modality to time, user context, spatial context, travel purpose and itinerary, as well as (e) interactions to the complete device context (cf. Table 1). Although these adaptations constitute the most important elements of the adaptation spaces, spanned by the adaptation dimensions on the one side and context dimensions on the other side (cf. Table 1), there are obviously many more meaningful and promising elements in order to further improve usability and ease of use of mobile services in tourism, e.g. adapting (a) the textual content, images and multimedia content to the time context, user context, environment or travel context, (b) dialog and navigation, and layout and structure to the user context environment and purpose of travel, as well as (c) assistance and fault tolerance to the device context, user context and environment. Thus, extending the adaptation framework by extending existing or adding new adaptation mechanisms in order to reach a more comprehensive coverage of the adaptation space is the naturally next step in this vein of research. The single elements of the adaptation space can then be evaluated in a real-world setting, concerning their increase of usability and ease of use of mobile services in different application scenarios.

Acknowledgments

The research was kindly financed by the Austrian Research Promotion Agency (FFG) and the Tiroler Zukunftsstiftung.

Biographical Notes

Wolfram H öpken is professor for Business Informatics and eBusiness at the University of Applied Sciences Ravensburg-Weingarten. He is director of the E-Commerce Competence Centre Swabia (KECoS) and scientific director of the eTourism Competence Center Austria (ECCA). His main fields of interest are intelligent trip planning and recommendation systems, semantic web and interoperability, mobile services and business intelligence, and data mining.

Matthias Fuchs is full professor at the European Tourism Research Institute (ETOUR), Mid-Sweden University, Sweden. He is also the scientific director at the eTourism Competence Center Austria (ECCA). His research interests include destination monitoring, tourism marketing, e-business adoption and impact, m-commerce as well as knowledge retrieval and management in tourism.

Markus Zanker is an assistant professor in the Department for Applied Informatics and the director of the study program Information Management at the University of Klagenfurt. He is also a cofounder and director of ConfigWorks GmbH, a provider of interactive selling solutions. His research interests focus on knowledge-based systems,
particularly in the fields of interactive sales applications such as product configuration and recommendation.

Thomas Beer worked for several companies in the field of tourism and automotive such as BMW, ECCA, German Lufthansa, or PASS IT Consulting. He also worked for the German Research Center for Artificial Intelligence where he joined the SmartWeb project. Currently, he is finishing his Ph.D. where he dedicates himself to context-aware information push services.

References


R. Law (Eds.), *Information and communication technologies in tourism* (pp. 173–183). Wien: Springer.


