Task-oriented Information Access in Personalized Mobile Travel Guides

Stefan Pfennigschmidt, Eleni Tsigka

Fraunhofer Institute for Software and Systems Engineering ISST, Germany,
{stefan.pfennigschmidt, eleni.tsigka} @isst.fraunhofer.de

Abstract

Many location-based and personalized mobile services have been developed in recent years. Nevertheless, user acceptance has not been as high as expected. Aiming at making this kind of services more accessible and appealing to users, we are working, in the framework of the European project TALOS, toward providing a novel way of accessing information in mobile environments. Instead of the usual provision of content, which is structured around semantic categories and can be searched with keywords, we adopt the idea of task-computing in the organization of content. Our goal is to combine context-awareness and task-based information selection in order to offer accurate and personalized content to the users. We apply this approach to the area of mobile travel guides. This paper presents the idea of task-based travel guides and discusses the application of this concept in a mobile guide and an eBook-based guide prototype.

Keywords: *mobile services, personalized services, situation-based services, task computing, travel guides*

1. Introduction

Mobile applications can provide access to a wide range of relevant information and services by ensuring personalization. However, the user adoption of such kind of applications and of location-based services (LBS) (see, for instance, [9]), in particular, has not reached significant levels and is restricted to specific types of services (navigation systems, weather information, and directory services). Apart from the financial cost involved, users often underestimate the value of such applications because they consider them time consuming, too complex, or inconvenient. In the framework of the European project TALOS [3], an international consortium of research institutions and SMEs from Greece, Belgium, Austria, and Germany develop a novel approach to simplify information access in mobile information services. The fundamental idea behind it is to employ the metaphor of task-computing to ease organization, selection, and provision of content in order to deliver personalized mobile information services. Context information (e.g., location) is used to recommend tasks from predefined or user-adapted task hierarchies, which in turn are used to select and present relevant content. The approach is applied to the area of mobile tourist guides, where we target two different output channels; (1) a simple, personalized, task-oriented e-book (providing access to static content); and (2) an advanced mobile guide that also includes access to dynamic services.

Task Computing has been identified as a paradigm to fill the gap between *tasks* (what a user wants to do) and *services* (functionalities that are available to the user) [6]. The original idea was to utilize semantic technologies to enable non-technical users to accomplish complex tasks in device-rich and service-rich pervasive computing environments. This idea was adopted in the area of mobile context-sensitive services, where it was used to assist users in finding appropriate services for solving real-world tasks [8]. The notion of tasks was also used as a special situational parameter when using map-based mobile guides [10]. Also city portals for tourists start to integrate task-oriented functionality like arranging a personal tour through a city [4] or using a hierarchical system of categories to plan for leisure activities in a certain area [2].

The paper is organized as follows. Section 2 presents the idea and system concept of task-based travel guides. Section 3 discusses a data model combining the notions of user situations, task hierarchies, and content along with meta data to describe its semantics or spatial and temporal characteristics. The application of our approach is shown in two prototypes: a mobile travel guide, designed for the Apple iPhone and a task-based eBook travel guide (Sec. 4). Task- and map-based user interfaces help to select and access content of various types originating from existing travel guides, the Web, or from collaborative or personal sources. Finally, Sec. 5 draws our conclusion and discusses next steps.

2. System Concept

Current mobile services (e.g., tourist guides) support users with information and functionality in a tool-oriented fashion. User goals (e.g., going on a business trip to Athens) and the tasks necessary to reach a goal (e.g., go to the airport, find overnight accommodation) are not explicitly handled. The idea is to use a task computing approach to provide, select, access, and organize multimedia content in the context of mobile environments such as location-based services. The notion of tasks is used to enable identification of information (e.g., content from travel guides or services on the Web) supporting a certain task and to easily access that information.

2.1 Task-based Information Access

In order to elicit the general requirements on the functionality (from a user perspective) we considered the following general procedure.

1. *get user situation*: this request gathers contextual information about the user. Parameters encompass location, time of day, season, weather, profile, and so on.
2. *recommend tasks*: using context information in form of situations or situation sequence descriptions (see [7]) a set of suitable tasks is given to the user. Note that not only current context is taken into account. Also information about, e.g., appointments planned can be used to find appropriate recommendations.
3. *recommend content*: using the task (or task sequence) selected by the user and the contextual information a set of related content is presented to the user.
4. *access content*: this step comprises presenting a selected piece or subset of content to the user. The content presentation may be adapted according to the usage context (tasks, user situation).

2.2 Architecture

The system is structured into four major parts, (1) the *data* and content storage, (2) the *services*, (3) the *content portal*, and (4) the *mobile client* (see Fig. 1). The data storage and services parts belong to the system backend. The content portal and mobile client represent two different frontend subsystems supporting different application cases.

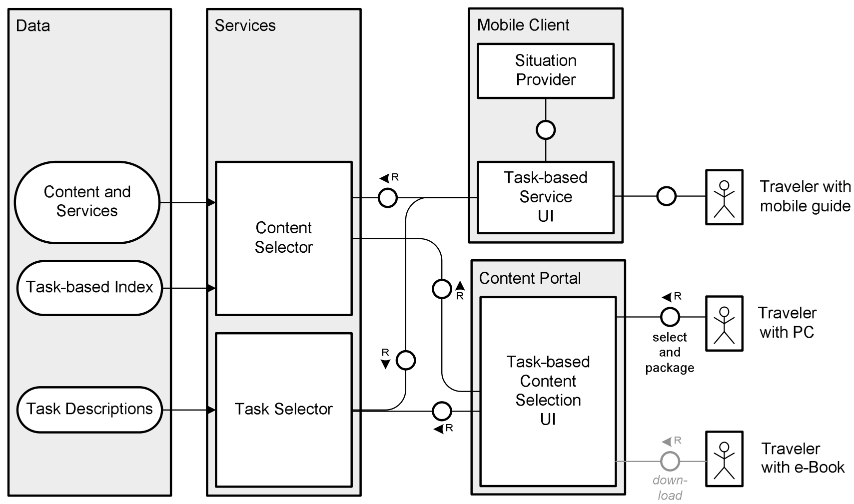

Figure 1: Concept of the TALOS System.

Mobile client. The mobile client provides a task-based interface where a mobile user can select tasks from a predefined or user-adapted task hierarchy and access relevant content. A situation provider component gathers context data, e.g., the user's current location, the time of day, or appointment information from the user's calendar. This information is used to parameterize the task recommendation and content selection processes.

Content portal. The content portal provides a user interface that is similar in functionality to that of the mobile client and is also based on the task concept. Tasks are used to identify and retrieve content relevant to a trip the user plans to do. It offers facilities to package the selected content and to create an additional task-based index to also support task-oriented navigation through the content of the package. The package is exported in EPUB format [1], which is an evolving standard for eBook publishing based on xHTML.

Services and data. The *task selector* uses *task descriptions* defined in a task model in order to recommend tasks to the user based on his or her situation. Tasks are defined along a task hierarchy from top-level tasks (e.g., 'visit city' or 'eat') down to fine granular tasks like 'buying a metro ticket at a ticket booth'. Further relationships between tasks are used to support task sequences or alternatives (this is further elaborated on in Sec. 3). The *content selector* selects content relevant to a given task. The selection process may be further parameterized using situational information from the user. The content selector uses a *task-based index* that links the tasks from the task model to the available *content and services*.

Alternative deployment. The presentation of the architecture in Fig. 1 suggests that the mobile client always uses a connection to the TALOS servers to retrieve the content the user is interested in. Especially, in tourist applications it is sensible to also provide *offline* access to content and even services, either because the user bought the content or does not want to connect because of roaming fees. Therefore, in an offline version, the components of the backend (i.e., the task and content selectors as well as content, descriptions, and indexes) reside on the mobile client, too.

3. Data Model

Requirements. We focus on the mobile client part of the system to specify the requirements for the data model. One basic requirement is to enable the selection and organization of content around tasks. Available content originates from existing travel guides, the Web, as well as from personal or collaborative sources. Tasks can be simple or complex and are structured based on hierarchical relationships. Furthermore, the application should support mobile users and offer the possibility to capture their context, which consists of time, location, weather, and preferences in order to suggest accurate tasks and provide personalized content.

Concerning the basic functionality of the mobile client, the representation of available content in a map-, a content-, and a task-based view is of significant importance. The application should also support transitions among the various views to allow for consistent recommendations to the user. Moreover, the provision of planning, search, filtering, and bookmarking mechanisms is required. The adopted data model, which is described in the following paragraphs, addresses all these requirements.

3.1 Primary Entities

Figure 2 shows the data model with its primary entities and relationships. The entities shown allow for managing different types of multimedia *content*, spatially-referenced objects (*POI*), *tasks*, and *actions* (as task instances). Additional *semantic categories* are used to add meaning to objects.

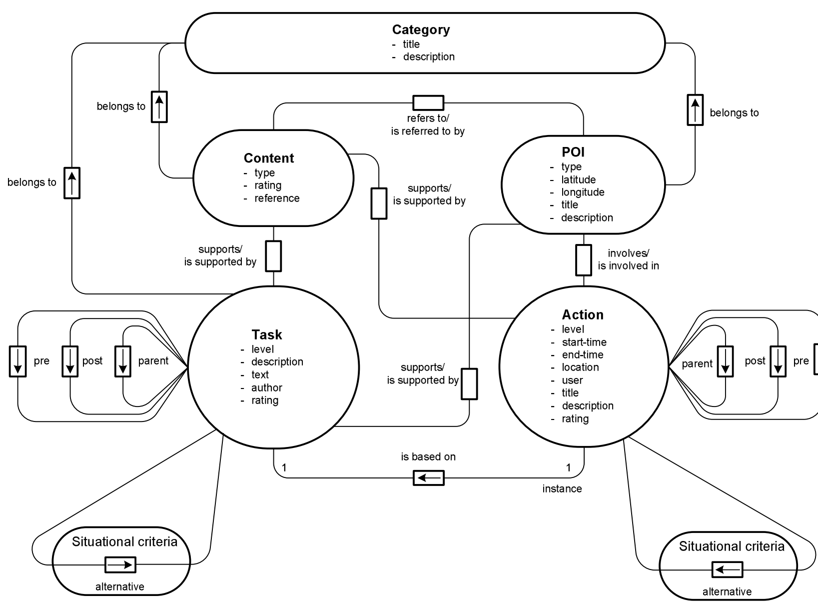

Figure 2: Data Model.

Content. The content entity represents all possible content types offered by the application. Categorization of content includes the type of book-based content found in existing travel guides; web-based content such as up-to-date information about events, exhibitions, and opening hours; collaborative content such as reviews and suggested tours; and personal content which can be photos or videos captured by the user. All types of content share common properties, such as the rating property and the same structure. In particular, content is composed of various components, namely book, section, paragraph, and resource, which store the textual and multimedia parts of it. The components serve as constituent units of content and are associated with one another in a hierarchical structure. To support this structure each component has a reference to its parent component.

Point of interest (POI). The POI entity models specific points of interest or entire locations. The properties of a POI include specific geographical coordinates, a description, and a rating. A location can be stored as a POI entity which represents the POI placed in the center of the location.

Task. The task entity corresponds to tasks which are necessary to achieve certain goals. Tasks resemble the steps followed by people in order to accomplish their assignments. For example, planning a trip, a tourist must take care of the travel to and from the destination, the accommodation, and the preferred activities. In order to arrange the journeys to and from the destination, the tourist has to select the preferred means of transport, then check timetables and availability, and finally book the tickets. Nevertheless, the steps that lead to determining the preferred activities during the trip are much more complicated and depend on the tourist's preferences and interests.

All the above mentioned steps are mapped to tasks in our application. Task authors undertake the work of identifying human-followed procedures (in the context of tourism) and decomposing them to distinct tasks. Some tasks are simple such as booking a flight and some are complex such as finding preferred activities. Furthermore, there exist *independent tasks*, for example, find out about the destination and *time-dependent* tasks, for instance, book a hotel only after the flight is booked or attend an open-air concert only if the weather conditions are good then. To support these restrictions and conditions in the synthesis of tasks, task entities are organized as a hierarchy. Each task has a level property, which reflects its position in the task hierarchy, and references to other tasks with which it is associated in composition or sequential relationships. Moreover, tasks take part in alternate (disjunctive) relationships which include various situational properties as parameters (e.g., weather conditions). Additional properties of tasks are the author and rating information.

Action. The action entity models concrete instances of tasks specified by the user of the application. For example, an instance of the task 'book a flight' could be 'book a flight to Berlin at 11:00 on July 1st, 2009'. An action can be created by attaching location, time and other situational information to a specific task. In our application actions are created by the user during the planning phase. Finally, the set of created actions forms the plan of the user. Given that an action is constructed based on a task, it shares the same properties with it. The action inherits the position in the task hierarchy, the composition, sequence, and alternative relationships of its parent task. Furthermore, the action entities have a spatial and a temporal dimension which describe the situation of the user when this action takes place.

Semantic category. This entity represents categories that can classify semantically information and in particular the content, POI, and task entities. For instance, a semantic category can be “hotel”, “museum”, “park”, “education”, or “amusement”. By assigning semantic categories to content, POI, and task entities we add meaning to them.

3.2 Primary Relationships

Content, POI, and task entities are associated in many-to-many relationships with the semantic category entity. These relationships describe the subject of each content unit and task, issues related to them, as well as possible functionality and usage of each POI. Semantic categories can be easily understood and selected by the user in order to access information or filter the data presented in each view.

Additionally, many-to-many relationships among POI, content, and task entities are established. The relationship between content and task entities enables the task-based selection and organization of content, as each unit of content (book, section, paragraph, resource) is associated with one or more tasks. Moreover, relating the content and task entities to POI entities allows for making transitions from the content- and task-based views to the map-based view and vice-versa. For example, geo-referenced content and task entities lead to points of interest relative to the selected content or task. Similarly, action entities are associated with the content and POI entities. Two additional characteristics of these relationships are that the user is able to attach personal content to an action and that the action's spatial dimension can be mapped to a known POI. Apart from these cases an action entity inherits all the relationships of its parent task entity to which it is also connected.

3.3 Secondary Entities and Relationships

Secondary entities and relationships ensure the support of the system's remaining functionality. The bookmark entity is used to store bookmarked information in all views of the application. The entity is categorized in three types, namely the content, task, and POI bookmarks. Additionally, the recent content and recent task entities model content and tasks that have been read and selected recently. These can be used to make recommendations to the user. Finally, the map tiles cache entity stores the necessary map tiles in order to provide offline map views.

4. Application Prototypes

As an application of our approach we develop a prototype system with two output channels; the first consisting of a mobile travel guide for the Apple iPhone platform and the second of a travel guide in an eBook form.

Mobile guide. The mobile travel guide is an iPhone application which offers touristic content employing the task computing paradigm. It features a “cube-shaped” user interface, the facets of which correspond to the book-, the map-, and the task-based view of the content. This interface allows for easy transitions among the different views, offering consistent content recommendations to the user. Available content involves existing travel guides for Dresden and geo-referenced content created by Michael Müller Verlag, as well as map tiles data provided by the OpenStreetMap project. Data storage on the mobile client is handled by an SQLite system and the Apple Core Data Framework is used as a persistence application programming interface. Finally, a combination of the Apple Core Location Framework and Wireless Positioning Techniques (developed by IMIS ATHENA as explained in [5]) are utilized in order to capture the user's location (indoors and outdoors) and to offer context-aware content.

Task-based eBook. The task-based eBook constitutes the static alternative, however, user-customized output of the system. The user is able to create an eBook tailored to his or her needs by selecting the appropriate content through a task hierarchy. The EPUB format is employed for the creation of the eBook ensuring compatibility with as many eBook readers as possible. Also here, the idea is to provide different ways to find relevant content when traveling. Apart from a table of contents, a keyword index, or full text search, a task-based index as well as a hierarchical map interface is used to navigate through the content of the eBook.

5. Conclusion

We have discussed the utilization of tasks in the process of organizing, selecting, and providing information. Similarly to the steps followed by people to fulfill their goals in real life, tasks can be used in order to identify accurate information for every situation in an efficient manner. Porting this idea to the field of travel guides and combining it with the notion of context-aware services we propose a highly accessible personalized travel guide, straightforward even for non-technical users. In this way, the situation of the user is used by the system in order to recommend relevant tasks and the selection of the appropriate tasks lead to the content of interest. To support these ideas we have presented a system architecture that supports two output channels – one static and one dynamic – and a data model which combines user situations, task hierarchies, and content along with meta data for their semantic interpretation.

Our ongoing work evolves around modeling the temporal and spatial granularity of actions and situations so as to finalize the data model. Future steps also include the complete implementation of the data model, dealing with all the restrictions imposed by a mobile environment ensuring high performance. As far as the user interface is concerned, the first design phase is to be followed by evaluation and in the end by the implementation phase. As a last step, complete testing of the functionality is planned as well as the conduct of user acceptance tests in order to evaluate our prototypes.

6. Acknowledgements

The work described in this paper was carried out in the framework of the TALOS project (EC grant 222292). We would like to thank the TALOS team for their valuable input and interesting discussions.

7. References

[1] EPUB standards specification. International Digital Publishing Forum (IDPF), http://www.idpf.org/specs.htm, last accessed October 6, 2009.

[2] planjam. planjam.com, http://www.planjam.com/, last accessed October 6, 2009.

[3] TALOS project web site. TALOS consortium, http://www.talos.cti.gr/, last accessed October 6, 2009.

[4] unlike.net. Unlike Media Ltd., London/Berlin, http://unlike.net/, last accessed October 6, 2009.

[5] S. Athanasiou, P. Georgantas, G. Gerakakis, and D. Pfoser. Utilizing wireless positioning as a tracking data source. In SSTD '09: Proceedings of the 11th International Symposium on Advances in Spatial and Temporal Databases, pages 171–188, Berlin, Heidelberg, 2009. Springer-Verlag.

[6] R. Masuoka, B. Parsia, and Y. Labrou. Task computing – the semantic web meets pervasive computing. In D. Fensel, K. Sycara, and J. Mylopoulos, editors, The SemanticWeb – ISWC 2003, volume 2870 of LNCS, pages 866–881, Berlin/Heidelberg/New York, 2003. Springer-Verlag.

[7] U. Meissen, S. Pfennigschmidt, A. Voisard, and T. Wahnfried. Resolving knowledge discrepancies in situation-aware systems. International Journal of Pervasive Computing and Communication JPCC, 1(4):327–336, December 2005.

[8] T. Naganuma and S. Kurukake. Task knowledge base retrieval for service relevant to mobile user's activity the semantic web – iswc 2005. In Y. Gil, E. Motta, V. R. Benjamins, and M. A. Musen, editors, The Semantic Web – ISWC 2005, volume 3729 of LNCS, pages 959–973, Berlin/Heidelberg/New York, 2005. Springer-Verlag.

[9] J. Schiller and A. Voisard, editors. Location-based Services. Morgan Kaufmann, San Francisco, 2004.

[10] A. Zipf and S. von Hunolstein. Towards task oriented map-based mobile tourist guides. In Proceedings of the International Workshop HCI in Mobile Guides (held in conjunction with Mobile HCI 2003), 2003.