Context driven, adaptive tour computation and information presentation

Ronny Kramer, Marko Modsching, Joerg Schulze, Marcel Hermkes, Klaus ten Hagen

Department of Computer Science
University of Applied Sciences Zittau / Görlitz, Germany
K.tenHagen@HS-ZiGr.de

Abstract. Standard tours combined with lack of information make many tourists miss some important experiences a destination has to offer. The target of the Dynamic Tour Guide (DTG) is to discover Tour Building Blocks (TBB), e.g. attractions like sights or restaurants, dependent on the actual context which is defined by personal interests, location and schedule of a tourist. It determines current information, e.g. opening hours or availability. It plans an individual tour, provides navigational guidance during the execution and offers information depending among others on the direction the tourist is approaching the attraction. All together the DTG is a mobile agent enabling a personalized, spontaneous and guided tour.

1 Motivation

Foreign tourists sometimes stand in front of closed facilities, since e.g. museums have different opening hours or might offer special expositions. On a summer weekend restaurants might be fully booked, whereas in November many restaurants will be closed. Because of this lack of information (availability), many tourists are unprepared following signs, studying maps or attending a guided tour on the spot. As human tour guides generally serve groups of tourists they follow predetermined routes to the major sights. Therefore the majority of the tourists end-up on the beaten tracks. Interesting sights just a couple of hundred yards off the main tourist arteries are rarely visited.

The ideal is to have a local guide, who understands the individual interests and timeframe, knows the local situation and gives a personal tour, and which additionally fits into a pocket. This is the objective of the Dynamic Tour Guide (DTG). The purpose is to devise a tour, just like an expert guidance would do after getting to know a tourist's preferences, by means of new technologies like mobile applications and context aware computing.

The first section will examine related projects in comparison to the concept of the DTG which will be presented by a scenario in the following section. The main part of this article will cover the development of the DTG by describing the architecture and discussing context awareness and semantic matching in detail. Finally the discoveries will be summarized

2 Related work

Tour Guides have always been an important topic for research activities. In contrast to the DTG the following important projects only consider a part of it's criteria to generate a tour:

- The Crumpet project (Schmidt-Belz, 2003b; Crumpet, 2004) enables a mobile agent to find certain sights, to present them on a map and to calculate a route to a selected one.
  → The sights are found because of their locality. The user has to decide for her-/himself whether they are interesting for her/him and if she/he has enough time to visit them.
- The software developed by enorro (Enarro, 2004) provides predetermined tours presenting the most important sights in many big cities all over the world. The tourist needs a PDA with a special player and the content for the particular tour. She/he also has to have navigation software which will lead her/him to the different places. The attractions are then presented using audiovisual information.
  → The user can only select an existing tour for available cities. The sights being shown to the tourists are pre-selected to suit the interests of a broad demographic.
- Among others the AgentCities-project (AgentCities, 2004) deals with finding restaurants. The system contains a restaurant-ontology and a restaurant guide web service. However all information, taken from well known published restaurant guides, is static.
  → With the use of static data there's no way to react on crucial changes in the context, e.g. opening hours or table reservations.
- In connection with the AgentCities framework the "Fujitsu Laboratories of America" (Fujitsu, 2004) has developed an event organizer. Also based on an ontology, it selects a restaurant according to the guest's preferences and makes a reservation when planning an evening.
  → This is a step towards context-awareness, because the search for a restaurant is dynamic due to the user's preferences.

➔ Predetermined tours is not the objective of the DTG, it intends to generate an individual tour in real-time. Additionally it pays attention to the local situation like opening hours by always having up to date information via web services. Hence the DTG is more flexible by considering a much broader set of contextual information and by reacting on external influences.

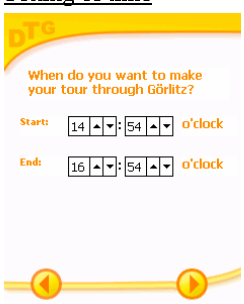
3 Scenario

Expectedly most people will own a mobile device in the next couple of years, cities will be covered with WLAN access points and DGPS will provide localization with a precision of at least 1 m. These are the preconditions to develop a Dynamic Tour Guide. The following scenario will describe its functionality best:

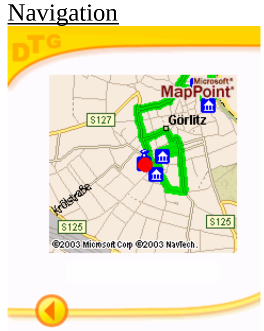
"A businessman has an appointment in a foreign city in the evening at 2pm. After arrival at his destination in the morning at 10 o'clock, he has some time left and would like to get to know the city. He starts the DTG which is installed on his mobile device. Furthermore, the mobile device is aware of its position via e.g. the Global Positioning System (GPS-WAAS) and it also maintains a personal interest profile. Setting the available time period to 4 hours will start a tour request. The DTG automatically discovers the sights and services at this destination, interrogates the corresponding web services to update the current information and then computes potential tours by selecting attractions according to the personal interests. As the tour will include noon, a lunch-break is integrated. A table in a suitable restaurant is booked for 12 o'clock. After selection of the tour, the DTG will visualize it on a map, giving the tourist the option to modify it. Then the tourist starts the tour. The DTG guides him via audio information to the first attraction which is a church. Noticing the tourists approach, the DTG draws his attention to it by giving audio information about the architecture style and history of it. The tourist listens and watches carefully. As he spends more time at the attractions as the DTG had planned, the next attraction is left and he is directly guided to the restaurant where he must be at 12 o'clock. After lunch and on his way to his appointment location, the DTG realizes that there is some time left and leads him to another monument, again presenting audiovisual information. After some minutes it gives a warning signal to let the tourist know it's time to leave. It leads him straight to the office building so he makes it to his appointment in time."

Please see Table 1 presenting screenshots of that scenario:

Setting of time



Navigation



Information

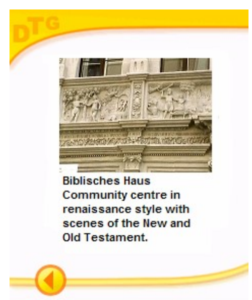


Table 1: Screenshots

4 Architecture

Each tourist has a mobile device using e.g. GPS to determine its location. The mobile device is connected to the internet either via GPRS or UMTS. Each sight, as a possible component of the tour (TBB = Tour Building Block), is semantically modeled by a content provider using an AuthoringTool. This model contains address, interest coverage, picture and audio files and general information. Each TBB will have its own web service (WS) to store and provide these data. A service provider like a restaurant will wrap the local restaurant management system by a WS to grant public access to the semantic model, current information, e.g. opening hours, and a transactional interface to e.g. reserve a table. The WSs of the TBBs are registered at a UDDI registry.

The DTG server is executing a semantic match algorithm to rank the sights for a specific tourist. A computationally more demanding task for the DTG server is the computation of a tour as a sequence of TBBs.

Audio hints and a map for navigation are provided by standard navigation software installed on the mobile device to guide the tourist to the next TBB. The DTG provides information about a TBB as the tourist approaches it depending on the direction. Furthermore it adapts the higher-level plan for the remaining time to the actual walking speed and the time spent at each TBB. The standard navigation software will try to get a tourist back on-track to the next TBB. After some time limit the DTG will interpret the continued movement of the tourist as a decision and adapt by computing a new tour starting from the current position.

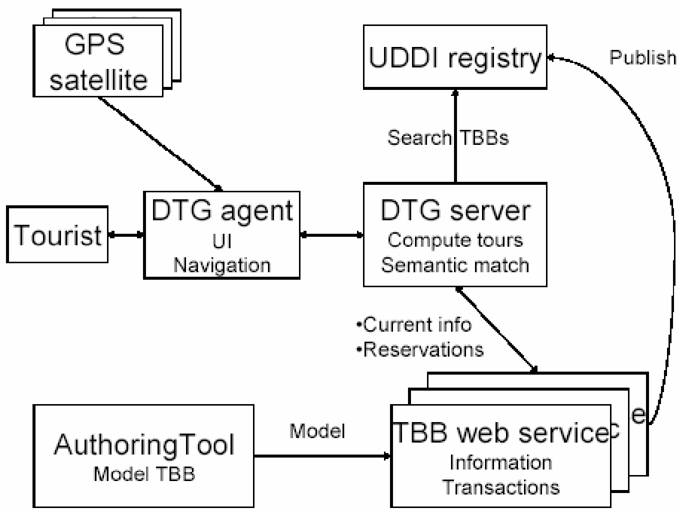


Figure 1: System architecture

5 Context awareness

Context spans the situational information. Any feature characterizing an entity and its environment determines its context. This context can be divided into different areas:

1. Personal context: The personal context includes ones personal information. It is defined by static elements like name or interests and dynamic elements like walking speed and current position
2. Local context: The local context consists of ones environmental information. These are for instance street and number of the actual position or the weather.
3. Service context: The services context describes the available services. Static elements are historic information about a sight, whereas a current exhibition or availability of a table in a restaurant is dynamic.

A context aware system is able to adapt its functionality because of filtered out contextual information (Korkea-aho, 2000). This is called ambient intelligence; the personal context is mapped with the services context and the local one. The DTG does so, using the following information to create a tour according to the actual context:

- Personal interests to rate (and select) the available sights
- The available time to limit the tour duration
- Opening hours of e.g. museums or restaurants to ensure availability
- The current position to determine nearby sights

Then it can plan an optimal, user specific tour.

Additionally, it will consistently supervise the ongoing tour and react on any deviations like changing walking speeds or additional breaks by recalculating the tour to make sure that the tourist arrives at the desired endpoint in time. Hence it has to react on changes concerning the context by constantly observing:

- The walking speed and tour duration to notice time problems
- The position to realize a tourist's approach to a sight or to get aware of distractions
- The walking direction to be able to call the tourist's attention to visible sights and start giving suitable information

6 Semantic matching

The central problem of this project is the selection of the right attractions. This task is different for any tourist as the contexts always differ. The personal context of the tourist has to be mapped with the local one. The interests, the available time period and the position of the tourist are most important. Based on this information a human expert can decide which tour would possibly fit best, but the challenge is to let the decision be made by a program. Therefore the computer needs to understand the meaning of certain data. The solution is to define a common knowledge base, containing all possible terms, arranging relations like synonyms and defining attributes – an ontology. It's a model of a specific area of reality. Every concept, existing in the real world, is displayed as a class. Relations between classes result in a hierarchical structure of all concepts, where each class can have parent classes and child classes. Attributes serve to define properties in order to describe classes more precisely.

The ontology is used to semantically model the interests of the tourist and the category of the TBB. At the beginning an ontology will be defined for a single destination. This ontology will have to be extended slightly in order to be used for other destinations in the same area. As the system is being applied to other regions it is important to maintain a hierarchical ontological system in order to enable reuse of the interest profiles. Otherwise a tourist would have to describe his interests from scratch whenever he enters a new region, which at best will lead to very shallow interests profiles.

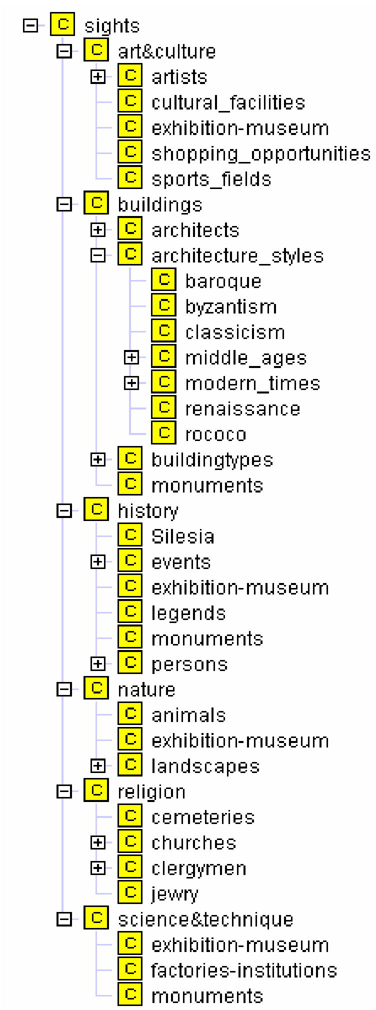


Figure 2: Interest ontology

All existing sights of a city (here: Goerlitz, 2003) are grouped into main categories of interests, which don't have anything in common. These are art & culture, buildings, history, nature, religion and science & technique.

Each category is subdivided. This allows a more precise modelling of interests. For example if a tourist is interested in buildings, he can either select a certain type of building like bridges, castles and so on, or he can opt architectural styles like baroque, art nouveau or others. The idea is that if a tourist's preference doesn't comply with a feature of a sight itself, but with a neighbour class (subclass or parent class) in the ontology, the tourist will probably be interested in that sight as well. That means that close-by classes are expected to be semantically similar so that relationships become visible easily. One example shall illustrate the way similarities are identified. If somebody is interested in animals (a subclass of nature), he's likely to be interested in nature in general. Thus also sights being described as landscape will satisfy his desires in some measure.

The content providers use an AuthoringTool to sort the TBBs into this hierarchy as a first step to create the TBB model. Most TBBs will be listed in different branches of the hierarchy, e.g. a church might be listed under *Religion/Churches* and *Architecture styles/Middle ages/Gothic*. The sorting process results in the creation of an XML-profile that contains all chosen categories with their accompanying superclasses.

The tourist is expressing her/his interests using the branches of the hierarchy. She/he will go through the exercise at one destination, and then rightfully expect that this investment will be reused at the next.

One tourists' interest profile and the TBB models are used by the semantic match algorithm to compute the degree of similarity based on the ontology. The degree of similarity is measured by the amount of Interest Matching Points (IMPs) for each TBB given a concrete interest profile. As mentioned above, the ability to deal with several degrees of similarity is important, since if there aren't any sights available that cover the tourist's interests exactly, ones that meet related interests should be considered as well. Therefore the semantic match algorithm evaluates the hierarchical part of the ontology, which is a directed graph, with the given interest profile. The node presenting that field of interest chosen by the tourist is evaluated with 1. There are two functions the rest of the nodes can be evaluated with, whereas each node is restricted to have exactly one parent-node. Going up, the IMPs of the nodes are divided by two:

$$y \leftarrow f^{u}(x) = \frac{1}{2}x$$

Going down, the subnodes receive the same IMPs as their parent node:

$$y \leftarrow f^{d}(x) = x$$

Presumed node B was chosen as the starting point, an evaluated graph looks like this:

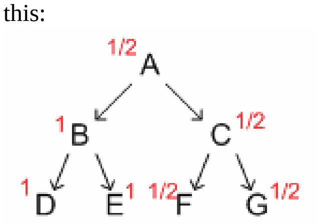


Figure 3: Evaluated graph

For nodes D and E function $f^d(x)$ fits, so they also receive the IMPs of 1. Node A receives ½ because of function $f^u(x)$. C is rated with ½ by $f^d(x)$ starting in A, and then starting in C $f^d(x)$ rates F and G with ½ too.

Depending on the amount of interest fields in the profile the whole hierarchical structure is rated several times. Each time the instances receive points. At the end the points are summed up. The following example shall demonstrate that:

Shown in Figure 4 is the ontological hierarchy including the TBB's. The interest profile of a tourist contains the following paths:

```
<interest>
         <class>baroque</class>
         <superclass>architecture</superclass>
         <superclass>building</superclass>
</interest>
<interest>
         <class>tower</class>
```

```
            <superclass>shape</superclass>
            <superclass>building</superclass>
</interest>
```

All TBB's (rectangles marked with TBB 1 to TBB 5) receive points twice because of two different interest fields contained in the profile.

TBB 4, a baroque building meets the first interest of 'baroque' exactly and is rated with 1 which means it receives the maximum of 100 points. For the second choice 'tower' the rate is ¼. All together it reaches 125 points.

TBB 1 is a tower built in art nouveau style, therefore it belongs to two branches of the hierarchy. For the first valuation of the interest 'baroque' the node 'art nouveau' is rated with ½. Hence TBB 1 gets 50 points. The second valuation of the interest 'tower' results in a rate with value 1 for the node 'tower'. TBB 1 belonging to that node gets 100 points in addition. In total TBB 1 receives 150 points, as only the maximal amount of points of each rating process is relevant:

```
foreach interest in profile
       foreach TBB
            TBB.IMP += MaxPoints(TBB, interest)
```

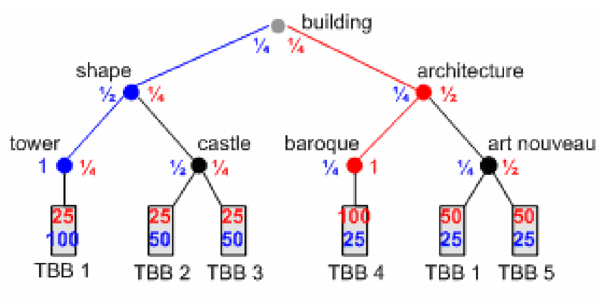


Figure 4: Hierarchy with TBB's

7 Tour computation

After the semantic match algorithm has assigned IMPs to each TBB a tour can be computed. A valid tour is a sequence of TBBs that can be visited within the time allocated by the tourist. Each TBB has an average duration of visit. Since 20 TBBs with the same start and end point lead to $(20-1)!/2 = 6*10^{16}$ possible tours, valid tours can't be cached in advance and thus need to be computed online. The challenge is to compute a valid tour that maximises the IMP. When the tourist asks her/his mobile device to compute a tour she/he is most likely standing with the mobile device in her/his hands somewhere within the destination. Given that situation the tourist won't care too much if the tour presented to him after e.g. 5 seconds has a few less

IMPs than the optimal tour. For most tourists the optimal tour is irrelevant – actually any tour – if the computation takes more than 5 seconds.

The used approximation algorithm is based on a depth first search. Figure 5 compares two variants of the search algorithm for a suit of benchmarks. The y-axis is scaled by the product #availTBB * #TBBinTour. #availTBB gives the number of TBBs the algorithm can choose from. #TBBinTour is number of TBBs in a tour or the depth of recursion. This product is a measure of complexity. The y-axis is scaled by the reduction of IMPs compared to the optimal solution. The heuristic to select a TBB for the candidate list, sort the candidate list and to insert a new TBB are discussed in (ten Hagen et al, 2004).

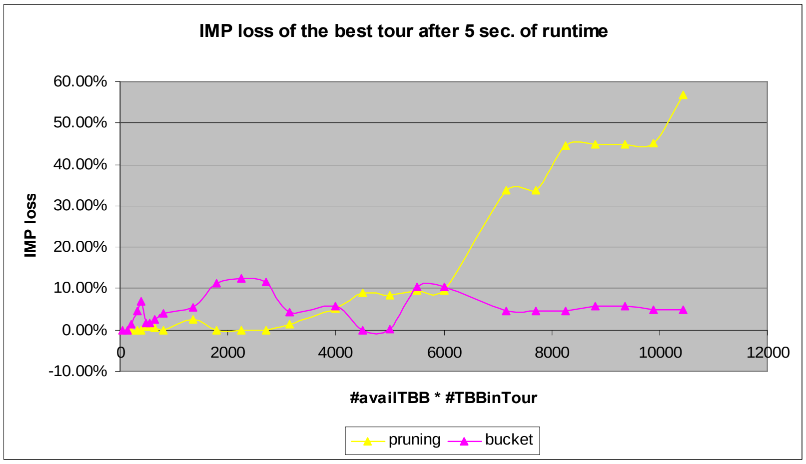


Figure 5: IMP loss of the best tour after 5 sec of runtime versus the optimum.

The pruning algorithm removes a TBB removes a candidate from the list after it has been added to a tour. This is a deviation from the standard depth first, where a candidate is only removed from the candidate list for all nodes below the node of insertion. The bucket algorithm divides the candidate list into buckets and processes the buckets sequentially. With a larger set of TBB the effect of working with a candidate list of finite length is extremely effective. For a destination with 1000 available TBBs and 10 TBBs in a tour the complexity product would be 10,000 and the reduction of IMP less than 7%.

8 Conclusion

The DTG uses innovative technologies in order to create individual, context-aware tours. Independent of location and time it determines the necessary information by detecting and interrogating available web services. It provides user guidance by giving navigation instructions and by offering the right information at the right time and place. A permanent supervision of the tour progress continuously adapts the tour to external influences or spontaneous decisions of the tourist.

Semantic matching is used to select TBB's according to the personal interests of a tourist. The tours are computed by a Directed Depth First algorithm. Benchmarks with different complexity indicate that a tour found after 5 seconds never has more than 15% less IMP than the optimal tour. Semantic technology and an innovative approximate algorithm enable tourists to enjoy a destination according to different contexts, which includes their interests, available time, actual position and environmental conditions. Also important is the fact that the DTG will help to spread the tourists more evenly across the destination and give exposure to a much wider set of services.

Acknowledgements

This project is part of VESUV in cooperation with Siemens AG, Microsoft's European Innovation Center (EMIC) and Fraunhofer Institute IGD. VESUV is supported by the Federal Ministry for Economics and Labour (BMWA).

References

AgentCities (2004): http://www.agentcities.org

Bussler, Christoph; Cardoso, Jorge; Fensel, Dieter; Sheth, Amit (2002): Semantic web services and processes: Semantic composition and quality of service, Irvine CA

Crumpet (2004): http://www.ist-crumpet.org/

Fuijtsu (2004): http://sf.us.agentcities.net/sf/

Godart, Jean-Marc (2003); Beyond the Trip Planning Problem for Effective Computer-Assisted Customization, *Information and Communication Technologies in Tourism 2003*, Andrew Frew et al. (eds.), Springer Computer Science

Enarro (2004): http://www.enarro.com

Goerlitz (2003): http://www.goerlitz.de/de/tourimus/kunst/kultur-denkmale2003.pdf

Horrocks, Ian; Li, Lei (2003): A software framework for matchmaking based on semantic web technology

Korkea-aho, Mari (2000): "Context aware applications survey", Helsinki University of Technology. http://www.hut.fi/~mkorkeaa/doc/context-aware.html#chap3.2

Lopez, Beatriz (2003): Holiday Scheduling for City Visitors, *Information and Communication Technologies in Tourism 2003*, Andrew Frew et al. (eds.), Springer Computer Science

Maedche, Alexander; (2003), Staab, Steffen; Services on the Move: Towards a P2P-Enabled Semantic Web Services, *Information and Communication Technologies in Tourism 2003*, Andrew Frew et al. (eds.), Springer Computer Science

Schmidt-Belz, Barbara; Posland, Stefan (2003a): "User Validation of a mobile Tourism Service"; Workshop "HCI mobile Guides", Udine (Italy)

Schmidt-Belz, Barbara; Laamanen, Heimo; Poslad, Stefan; Zipf, Alexander (2003b): Location-based Mobile Tourist Services – First User, *Information and Communication Technologies in Tourism 2003*, Andrew Frew et al. (eds.), Springer Computer Science

Ten Hagen, Klaus; Kramer, Ronny; Müller, Patrick; Schumann, Bjoern; Hermkes, Marcel (2004): Semantic Matching and Heuristic Search for a Dynamic Tour Guide, *Information and Communication Technologies in Tourism 2005*, Andrew Frew et al. (eds.), Springer Computer Science