A Location Aware Mobile Tourist Guide Selecting and Interpreting Sights and Services by Context Matching[1]

Klaus ten Hagen, Marko Modsching, Ronny Kramer
University of Applied Sciences
Brückenstr. 1, D-02826 Görlitz, Germany
K.tenHagen@HS-ZiGr.de

Abstract

Most tourists exploring a destination either join a guided tour or walk on their own using maps. Neither are their individual preferences nor the actual situation considered. These tours are strongly inflexible. The ideal is an intelligent guide taking care of the whole tour organisation and execution in time. This is the main objective of the Dynamic Tour Guide (DTG). The DTG is a mobile agent that selects attractions, plans an individual tour, provides navigational guidance and offers location based interpretation. This kind of ambient intelligence is based on the analysis of all available context information to support the tourist in any possible way with the help of a mobile device.

1. Introduction

Presently there's no way for a tourist to spontaneously explore a destination. But new technologies will make it possible in the near future. Tourists need an interest profile, a start- and end point and a given time period – all necessary data to compute a tour. This is described by their personal context which needs to be mapped with the environmental context at the destination. Context means all available information at a certain location for a certain time, what will be defined more precisely in one of the next sections. The challenge is to compute an optimal tour given the personal and local context. The tourist can modify the proposed tour.

During the execution of the tour the tourist will be guided to the next Tour Building Block (TBB) using standard navigation software, like MS Mappoint or Navigon [13]. When the tourist starts walking the DTG determines the actual walking speed of the tourist on this day given the conditions of the sidewalks and streets. This update of the personal context might make a recalculation of the remaining tour necessary. As soon as a tourist approaches a point where a TBB becomes visible the DTG will provide introductory information via a headset suitable to the direction from which the tourist is approaching the TBB. Otherwise the tourist might soon get disorientated when approaching from e.g. the opposite side. As long as she/he is in the proximity of the TBB, the tourist will receive audio information. Some tourists will decide to explore the TBB further by e.g. walking into the court yard. In this case additional information appropriate to current context is provided. As soon as she/he leaves the TBB, the DTG will stop the information provision and restart the navigational guidance towards the next TBB. In case the tourist stays much longer than initially assumed, the tour for the remaining amount of time will be recalculated.

On the way to the next TBB some tourists might get distracted by another attraction be it another sight or simply a shop. Then the DTG will interrupt the navigational hints and provide information for the current context if available. In case of a spontaneous visit to a local store the DTG will simply wait for the tourist to leave the location to continue on a tour recalculated for the remaining amount of time. Despite the navigational guidance through audio hints some tourists might get on the wrong path. The navigation software will try to get the tourist back on a path towards the next TBB. However beyond a certain deviation it becomes more meaningful to adapt the sequence of TBBs to the new location of the tourist.

[1] 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, July 17-21, 2005; San Diego, CA, USA; http://www.mobiquitous.org

2. Challenges

Tour Guides have always been an important research topic. The following projects summarize the current state of the art:

The Crumpet project [2], [15] enables a mobile agent to find certain sights, to present them on a map and to calculate a route to a selected one.

The software developed by eNarro [5] 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 loaded with 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.

In connection with the AgentCities [1] framework the "Fujitsu Laboratories of America" [6] have developed an event organizer. 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.

→ In contrast to existing tour guides the DTG computes an individual tour in real-time by considering available context information like personal interests and location based services. In order to build the DTG the following challenges have to be addressed:

- Acquisition of the interests of a tourist in a mobile context to seed the profile
- Ranking of TBBs by semantic matching
- Computation of a tour in less than 5 seconds
- Context aware interpretation of the environment
- Tour tracking and adaptation

3. Architecture

Each sight, as a possible component of the tour (TBB = Tour Building Block), is semantically modelled by a content provider using the DTG AuthoringTool. Each TBB will have its own web service (WS). A service provider like a restaurant will wrap the local restaurant management system by a WS. This WS will provide 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 published 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 of 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 to guide the tourist to the next TBB. The DTG provides information about a TBB as the tourist approaches it. Furthermore it adapts the higher-level plan for the remaining time to the actual walking speed and staying time at a TBB.

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 [4]. This enables the following features and interactions:

- Localisation:
  The mobile device is aware of its current position, either in a city via e.g. the Global Positioning System (GPS-WAAS) or inside buildings like museums via WLAN, Infrared grids or RFIDs.
- Service discovery:
  After arrival at a destination the DTG will determine the next DTG server in a UDDI registry. Based on the personal context like the maintained interest profile and the time period set by the tourist, the DTG will discover the local context like sights and services at this destination by scanning another UDDI registry, interrogate the corresponding web services to update the current information and then compute potential tours.
- Navigation and tour adaptation:

After the tourist has selected and optionally modified a tour, the local navigation software will visualize the tour on a map and guide the tourist via audio information. In the background the DTG will consistently track the execution of the ongoing tour for contextual changes, e.g. 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.

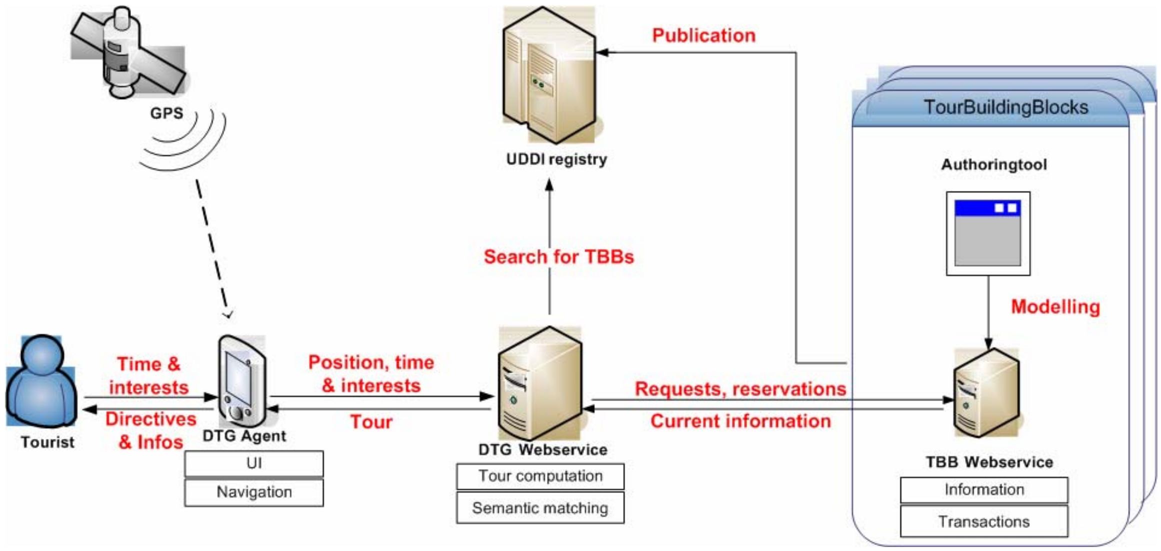

Figure 1: System architecture

4. Tour composition

4.1. Basic ontology

Semantic matching is applied to select TBBs according to the personal interests of a tourist. 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 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 region. 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 destination in a new region, which at best will lead to very shallow interests profiles.

All existing sights of a city (here: Goerlitz 2003, [8]) are grouped into main categories of interests, which have little in common. These are: art & culture, buildings, history, nature, religion and science & technology. The ontology for that has been created by an XML based language called DAML+OIL [3]. It's hierarchical view is shown in Figure 2.

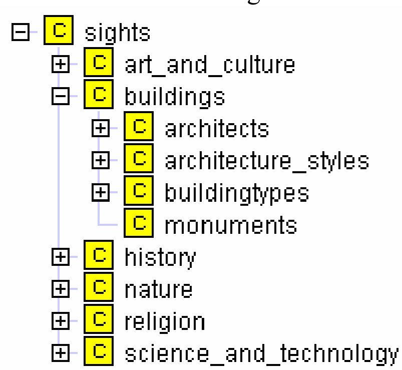

Figure 2: Hierarchical subset

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 sights also being described as landscape will satisfy her/his desires in some measure.

The TBBs are sorted into this hierarchy by the content providers using an AuthoringTool thereby creating the TBB models. Most TBBs will be listed in multiple 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 all relevant 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 expects that this investment will be reused at the next.

4.2. Semantic match algorithm

The ontology, the interest profile and the TBB models are used by the semantic match algorithm to compute the degree of similarity. 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. These are the TBBs with a high amount of points (IMPs = interest matching points). 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 this interest 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:

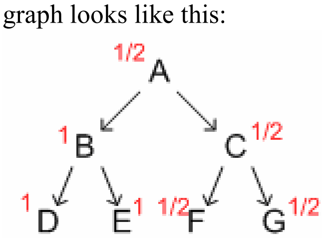

Figure 3: Evaluated graph

For node 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 $f^d(x)$ starting in A, and then starting in C rates F and G with ½.

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 the Figure 4 is the ontological hierarchy including the TBBs. 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 TBBs (rectangles marked with TBB 1 to TBB 5 in Figure 4) receive points twice because of two different interest fields. TBB 4 meets the first interest exactly and receives 100 points. 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. In total it gets 150 points, as only the maximal amount of points of each rating process is relevant.

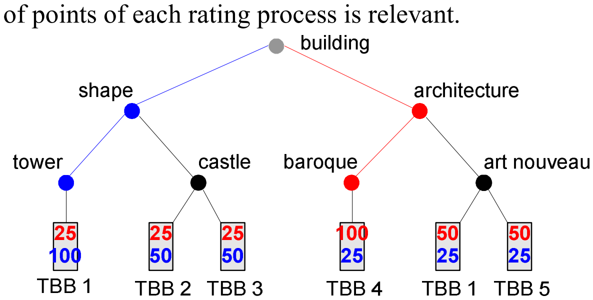

Figure 4: Hierarchy with TBBs

5. 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 little 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 x-axis is scaled by a complexity metric. The first factor #availTBB gives the number of TBBs the algorithm can choose from. The second factor #TBBinTour is number of TBBs in a tour or the depth of recursion. The y-axis is scaled by the reduction of IMPs compared to the optimal solution. Heuristics are applied to select a TBB for the candidate list, sort the candidate list and to determine the insertion point for a new TBB.

The pruning algorithm removes a TBB 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. Given the insertion heuristics this removal avoids the computation of identical tours. The bucket algorithm divides the candidate list into chunks and processes them sequentially. The outcome of processing a partition of the candidate list is an approximation of the optimal tour, which is improved further using the candidates from the next bucket. As Figure 5 clearly shows, the effect of working with a candidate list of finite length is extremely effective for a larger set of available TBBs. 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 still less than 7%.

The algorithm and its heuristics are discussed in more detail in (ten Hagen et. al, 2004) [16].

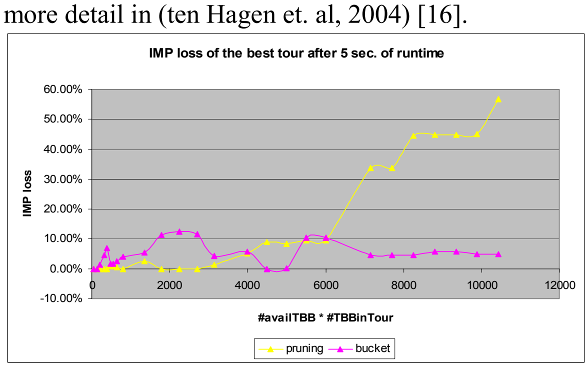

Figure 5: IMP loss of the best tour after 5 sec of runtime versus the optimum

6. Tour execution

6.1. Context driven interpretation

Context spans the situational information. Any feature characterizing an entity and its environment determines its context [10]. 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 local weather.
3. Service context:
   The services context describes the available services. Static elements are information about a close-by sight, whereas data about a current exhibition or availability of a table in a restaurant are dynamic.

Based on the tourist's personal context a tour starting at the current position, including sights according to the interests and ending in the given time frame is generated.

Shown in Figure 6 is an ongoing tour. The first stop is attraction A. Along the way a pre-loader will be called at regular intervals, e.g. 5 minutes, to download all contextual information for the area the tourist can reach within a certain amount of time, like the next 10 minutes. As a side effect this forward looking caching makes the DTG more robust in situation with spotty mobile coverage.

Moving closer to attraction A its data will be downloaded as well, so that it is ready for being presented on the mobile device when the tourist reaches the attraction. The presentation continues as long as the tourist keeps standing in front of it. As he leaves the presentation stops and the navigation to attraction B begins. On the way to attraction C the tourist leaves the path to go to another attraction (E) which seems interesting to him. As soon as possible the sight is searched in a registry by its address and the right information, having been downloaded in advance, is determined. As the tourist spends a lot of time listening to this information, he gets behind the schedule. The tour has to be recalculated, as C and D are not reachable both in the time left. The DTG alerts the tourist that it's time to go on with the tour in order to reach at least attraction C. After C the tourist is led to the desired end point.

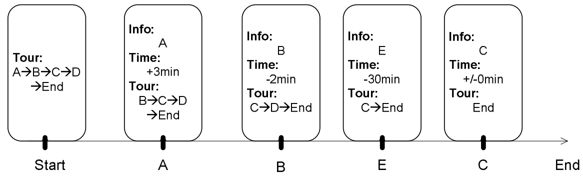

Figure 6: Tour adaptation

As mentioned above, the contexts are first used for the tour computation. The tourist's personal context, especially considering his actual position, interests and time frame, affects the selection of the sights. The contexts are secondly applied for supervising the tour execution. The DTG is always aware of context changes which appear most clearly when approaching a TBB. The following figure shall demonstrate the interactions resulting from that scenario:

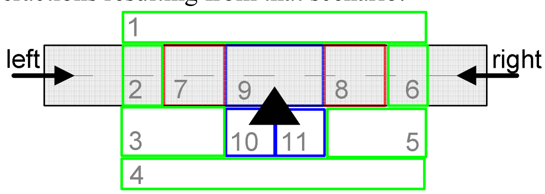

Figure 7: TBB model

The relevant TBB is located near a certain street. It is modelled by storing all necessary data in an XML file. This also includes a separation of the area around the TBB into virtual geometric forms. As circles cause overlaps the most suitable ones are rectangles. Furthermore rectangles can be evaluated computationally efficient. For every rectangle the upper left and the lower right coordinates are known. The coordinates of the tourist's current position are transmitted by a DGPS receiver. To determine if the tourist is situated in one of the rectangles it needs to be checked whether her/his coordinates are smaller than the upper left and bigger than the lower right ones:

$$P^{up\_left}(x, y) \leq P^{tourist}(x, y) \leq P^{low\_right}(x, y)$$

Entering a rectangle, the following events are triggered:

Table 1: Actions when approaching or leaving a TBB

Entering	from	Action
7	2	Alert about the forth-coming TBB, e.g. "On your **right** you see …."
8	6	Alert about the forth-coming TBB, e.g. "On your **left** you see …."
7, 8	9	Alert about the departure from the current TBB
9,10,11	-	Information about the **front side** or **back side** of the TBB
2	7	Navigation towards the next TBB
6	8	Navigation towards the next TBB
2,6	-	Navigation towards the current TBB
1,3,4,5	any	Alert of potential deviation and navigation back to the TBB or the route

Example:

The navigation software installed on the mobile device is performing the task of leading the tourist straight to the desired TBB. The tourist is approaching it coming from the left side. The required information derived from his personal context is his current position and his moving direction, which can be determined from her/his recent walking path. While walking, her/his position steadily changes. The DTG permanently checks the position to detect the intrusion into a rectangle:

1. The tourist enters rectangle 2. As she/he hasn't been in nr 1, 3, 4 or 7 before, nothing happens.
2. The tourist enters rectangle 7. This is the position the sight can be seen from. As he comes from nr 2, he is alerted that she/he will soon be able to see the TBB on her/his right side.
3. The tourist enters rectangle 9. The DTG starts giving information about the sight like its architecture style, history and so on by playing an audio file.
4. The tourist moves around the TBB and enters rectangle 10. The DTG changes the audio file to provide information about the backside of the TBB.
5. The tourist departs from the sight and enters rectangle 4, maybe to take a photo. The DTG now stops giving information but alerts the tourist that he is leaving the right path and is recommended to go back.
6. The tourist goes back into rectangle 10. The DTG continues giving information to the tourist.
7. The tourist enters rectangle 11. The DTG keeps on giving information.
8. The tourist enters rectangle 9. As he comes from nr 11, the DTG still gives further information if available.
9. The tourist leaves entering rectangle 8. As he has been in nr 9 before, the DTG just stops giving information and restarts the standard navigator do guide the tourist to the next station of the tour. In the background it prepares the next stop by downloading the necessary files for providing information at the following TBB or any other TBB in the immediate vicinity.

The action of the DTG in rectangles 7 and 8 strongly depends on the direction the tourist enters them from: If she/he comes from the right side and reaches rectangle 8, she/he is alerted that she/he can see the TBB on her/his left hand side. And if he/she then reaches nr 7 coming from 9, the information presentation will stop and switch to the navigation for the next TBB.

Figure 8 displays the complex interactions during the tour execution which is driven by the different contexts and the navigator. They all result in appropriate information like multimedia content, audible user guidance and maps. The context driven interpretation was described above and the tour adaptation and navigation will be covered in the following sections.

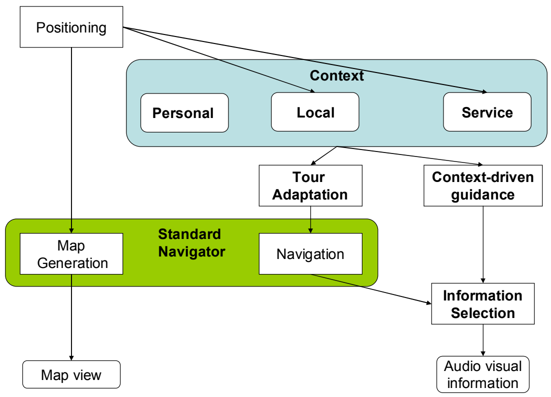

Figure 8: Context-driven tour execution and interpretation

6.2. Context driven interpretation at a complex sight

The method of context-driven interpretation described above is satisfactory for a small building like a tower or a house. But for larger attractions like a castle, a park or a big church, this scenario has to be extended by additional user guidance within the TBB itself. Having arrived, for instance at a big church, the tourist might stand in front of it quite helplessly, not knowing where to go and where any information is presented. The tour inside the whole tour might start at the front door of the church, showing e.g. a picture of its digital reconstruction on the screen of the mobile device. The screen will further present a map showing the single stations around the church with different information sources as sketched in the left picture of Figure 9. The audio hints are using cross references, e.g. "Please enter the church to receive information about the organ." Inside the church, the tourist might hear an mp3-file of a concert. When moving out again the tourist is informed that he will receive historical information about the tower at the backside. With audible instructions supported by arrows on the screen the tourist is guided to the next station, as depicted on the right side of Figure 9.

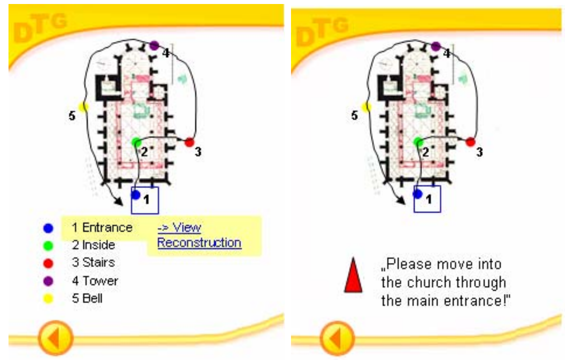

Figure 9: User guidance at a TBB

This ensures that the tourist doesn't miss any important information about a TBB. These navigation instructions given in the vicinity of a TBB, aren't generated by the navigator but are given by the DTG itself, since they are highly TBB specific. They are static for each tourist, use environmental descriptions like "Move around the right corner straight towards the stairs." and depend on the actual position and the positions the tourist has already been at before.

6.3. Navigation and tour adaptation

The DTG applies a standard navigator installed on the mobile device to navigate from one sight to the next. This navigator is a separate program using offline available geographical data for navigation and a stored map data for showing instructions and routes on a map. Furthermore the navigator gives instructions via audio to avoid the user holding the PDA in field of view all the time.

One of the main targets of the DTG is to ensure that the tourist gets back at the time he has established at the beginning. Therefore the tour has to be tracked constantly. The initial tour for example is calculated for a walking speed of 5 km/h. If the tourist walks slower through the streets, the tour has to be adapted by removing selected TBBs from the tour. Another important issue is a sight situated along the way that distracts the tourist. In this case the standard navigator permanently tries to bring the tourist back onto the regular route. After a couple of seconds the DTG recognizes that the tourist is interested in this sight and interrupts the navigator to provide context-driven interpretation for this sight. After that the rest of the tour has to be rearranged because the tourist hasn't the time to attend all other previously scheduled TBBs.

This kind of context-aware navigation can be compared with seafaring, where the standard navigator is the coxswain and the DTG is the captain who directs the whole trip. A schema of the navigation interaction is shown in Figure 10:

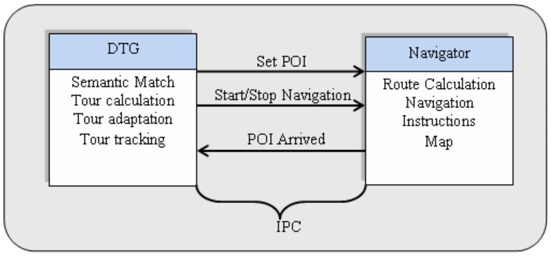

Figure 10: Interaction with navigation software

7. Conclusion

Conceptual the DTG uses context-driven technologies in order to create highly personalized adaptive tours. Independent of location and time it determines the necessary information by detecting and interrogating available web services. It provides complex user guidance by providing navigation instructions and by offering the right information at the right time and place. Permanent tracking of the tour progress considers external influences to adapt the tour. Diverse prototype implementations demonstrate the discussed technologies impressively.

Semantic technology and an approximate heuristic tour computation 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.

8. Future work

In order to check the usability of this semantic matching technology as well as the relevance of the determined results, empirical experiments are currently designed:

All important sights of Goerlitz will be modelled as TBBs. Crucial is the initial seeding of the personal interest profile. A broad selection of tourists will be asked to specify their interests based on the common ontology with the help of a simple application running on a mobile device. The TBB profiles are semantically matched with the interest profile to derive a list of TBB with IMP > 0. Different interest profiles should lead to different TBB lists. Thus a tourist who's interested in Religion will see completely different sights than a tourist who likes the architecture of the Wilhelminian style. Depending on were the sights are situated both might walk completely different routes, leading them to completely different areas of a city. The differences might be calculated as a percentage.

To judge whether the algorithm ranks the TBBs correctly, the users, having specified their interests before, have to rate some representative sights from different interest categories by text and picture. The results of the user are compared to those of the algorithm. The objective is to find the means of eliciting the interests of the tourist with the highest correlation.

A second investigation will determine the precision of common GPS receivers in a few cities. Then the methods of context determination will be refined. If the precision isn't good enough, additional means to find out where a tourist really is have to be introduced.

The upcoming field studies shall also analyze the behaviour and habits of tourists in general, so that the whole system can be adapted according to the results.

9. Acknowledgement

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).

10. References

[1] 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

[2] Crumpet (2004): http://www.ist-crumpet.org/
DGPS (2005): http://www.wsrcc.com/wolfgang/gps/dgps-ip.html; Wolfgang S. Rupprecht.

[3] DAML+OIL (2001): http://www.daml.org/2001/03/daml+oil-index.html, Joint Committee.

[4] EGNOS/SISNeT (2005): http://esamultimedia.esa.int/docs/egnos/estb/sisnet/over6.htm; European Space Agency.

[5] Enarro (2004): http://www.enarro.com

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

[7] 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

[8] Goerlitz (2003): http://www.goerlitz.de/de/tourimus/kunst/kulturdenkmale2003.pdf

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

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

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

[12] 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

[13] Navigon (2004): http://www.navigon.de

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

[15] 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

[16] 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