Proactive Exploitation of the Spatial Context in LBS
- through Interoperable Integration of GIS-Services with an Multi Agent System

Alexander Zipf, Hidir Aras
<firstname>.<lastname>@eml.villa-bosch.de
European Media Laboratory, EML, Heidelberg, Germany

Abstract

LBS need to provide information that is user, location and time adaptive. Thus system intelligence must define a module that processes the contextual information to deliver useful and interesting information to the user which has not yet been provided by the system.

This paper focuses on two main aspects: One is the integration of agent technology and gis technology. Here we focus on interoperability issues and explain the necessary middleware and communication aspects briefly within the example of an server infrastructure for Location Based Services (LBS). One of the key aspects (or even “buzzwords”) of agent technology is “proactivity”, meaning that the agents can act not only in reaction to user requests but behave autonomous [Woolridge and Jennings 1995]. Within LBS the spatial context, in particular the location of the user, act as a means to parameterise the offered system services. This is the major new key quality that LBS offer in contrast to conventional systems. In this paper we explain how proactive location based services can be realized using the spatial context of a nomadic user. The realized example will be in the tourism and city guide domain. In order to exploit the spatial context an agent needs to extract, aggregate and process information. We call this agent providing location aware proactive tips to a user – Spatial Context Agent (SCA) and present within this paper strategies how the aggregation and processing can be achieved effectively.

In order to increase interoperability the integration of GIS components into a distributed agent platform like FIPA-OS [Poslad, Buckle & Hadingham 2000] is being proposed [Zipf & Aras 2000]. In a Multi Agent System (MAS) higher level intelligent services (like practive location aware tips) arise as a result of collaborating agents that use information and basic gis services from different kind of sources. This paper will discuss the architecture and some of the realized agents and their interaction with other, non-GIS agents, through a FIPA-based Agent Communication Language (ACL) and protocols based on speech acts theory. This involves the use of domain ontologies. These are represented using XML schema. Further the interaction of the GIS-agents with the underlying GIS infrastructure is explained. Finally, it is necessary to discuss how the modeling of GIS services as intelligent agents can be exploited to improve the development of value added personalized context sensitive services for LBS.

Keywords: LBS, spatial context, interoperability, OGC, FIPA proactive agents, agent technology

1. Introduction

Within the Deep Map Framework [Malaka & Zipf 2000] and projects like the EU Project Crumpet [Poslad *et al.* 2001], the European Media Laboratory (EML) is developing a user- and spatial-aware tourism information system about Heidelberg. Research prototypes include web-interfaces for pre-trip planning, language driven mobile prototypes on a portable

computer as well as Location Based Services (LBS) for smartphones using GPRS and UMTS [Zipf & Malaka 2001]. Examples for innovative services are intelligent position determination [Kray & Walther 2000], the dynamic generation of individualized proposals for sightseeing tours [Stille 2001], resource awareness [Yun *et al.* 2001] and adaptive maps [Zipf 2002].
The realized prototypes are being developed as so called Multi Agent Systems (MAS, see below). This means that also the GIS modules are being encapsulated as agents. The agents build on OpenGIS compliant gis modules. For example a geodataserver has been wrapped by a so-called Spatial Agent using the OGC SFS for CORBA [Zipf & Aras 2001]. Further GIS services that are necessary to develop location-based services (LBS) realized as agents include a Map Agent, a Tour Agent etc. In order to communicate, the agents within the system need to agree on a common data structure for representing the entities and concepts of their "discourse". These structures (sometimes being referred to as agent's "ontology") are defined using XML schema. Using data-binding tools it is possible to convert these into other representation formats (like into Java-objects using Castor (Castor 2001) or Enhydra (Enhydra 2001)). For geographic entities and geometric representations GML (Geographic Markup Language)(OGC 2001) is being used. But at the moment GML lacks richer semantics like the definition of routes and tours, or temporal aspects. For the latter issue a temporal extension to GML has been proposed [Zipf & Krüger 2001].

The research topic of intelligent agents is now receiving some attention in the GIS research community. While mostly originating in an simulation context in raster or tuple worlds (like Cellular Automata etc. [Batty 2000; Openshaw 1991) it moved recently to more sophisticated individual-based simulations [Rodrigues & Raper 1999; O'Sullivan & Haklay 2000; Frank, Bittner & Raubal 2001]. Further approaches include for example applications in data seaarching [RAM *et al* 1998] and map generalization [Baeijs, Demazeau & Alvares 1996] is On the other hand there is research in software agent technologies in the area of software engineering that lead to standardization efforts within the intelligent software agent community to facilitate agent interoperability, mostly through the Foundation for Intelligent Physical Agents (FIPA).

In the next paragraph we want shortly to give some explanations and definitions about agent technology to avoid confusion with the interpretation of these terms in distinct communities.

2. Agent Benefits and Interoperability

Agents depict software entities that can run autonomous, proactive, can be personalized and support adaptivity. Agents allow encapsulation and distribution at component level (see Figure 1).

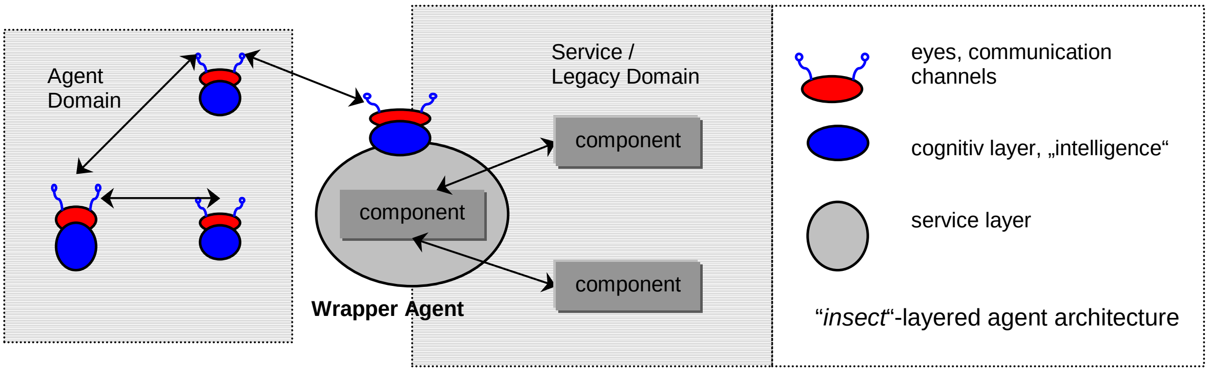

Figure 1: Integration of the Agent and Service/Legacy Domain through "wrapping"

Agents do not only encapsulate behavior but can decompose problems into subtasks that can be distributed among the agent community, where they are solved through agents that are specialized in solving a particular aspect of the domain problem (e.g. service location determination). A solution for a complex task (e.g. providing routing information in a form suitable for a PDA) can be achieved through communication and co-operation/collaboration among agents. Thus agents can be inserted at different levels of service aggregation. Agents sharing a common ontology can communicate with each other, reason, plan and execute tasks. The FIPA platform [FIPA 1999, 2000] supports inter-working with native software, interoperability between different agent platforms, agent human communication, middleware support and communication over an ACL, which can be used to encode ontology for working agent based systems. Using a layered "insect model" approach [Schlichter 2000] for programming intelligent agents the following procedure seems purposive:

- The agent's communication level uses industry-standard middleware platform to isolate communication from application details.
- The agent's body consists of interfaces for functionality in the conventional sense, where wrappers for legacy services can be used to integrate the needed basic services.
- The intelligence is located in the cognitive layer and can exploit different methods.

3. Multi-Agent Systems (MAS) and Agent Platforms (AP)

A Multi-Agent System (MAS) [Ferber 1999] is formed by a collection of agents with specific roles within a certain organizational structure. An agent platform (AP)(like FIPA-OS or JADE [CSELT 2000] rovides the infrastructure (communication middleware) in which agents can be deployed. In order to interact with agents on the local platform or on other platforms an agent must be registered to a Directory Facilitator (DF), that stores descriptions of the agents and the services they offer. An AP consists of the capability sets Agent Communication Channel (ACC), Agent Management System (AMS) and a default DF [FIPA 2000b]. As a result of the agent-distributed service communication the services and the service infrastructure can dynamically adapt to environmental aspects such as changes in the (available) network quality of service (QoS), to the insertion of new services into the system and therefore improve usability of services. Within the EU IST project CRUMPET such an open source FIPA-compliant AP has been ported to and optimized for small mobile devices [microFipaOS]. Some of our prototypes are being developed on a *resource-aware agent platform* (RAJA) [Ding *et al*. 2001], that sits on top of well known FIPA-compliant latforms like FIPA-OS, microFIPA-OS or JADE.

4. Interoperable GIS Agents within a Multi Agent System

Modern GIS need to be developed on open interfaces. GIS interoperability is now making good progress through the developments within OGC and got a boost through LBS (like the OpenLS initiative). The integration of such open GIS components into a Multi-Agent System (MAS) can be realized using an open-source agent platform like FIPA-OS, where the OGC services can be made accessible to other intelligent agents via specialized agents. This is a new approach to develop distributed GIS services that are being wrapped through FIPA conformant agents. Doing this, a two-way interoperability between different agent platforms and distributed GIS service can be achieved.

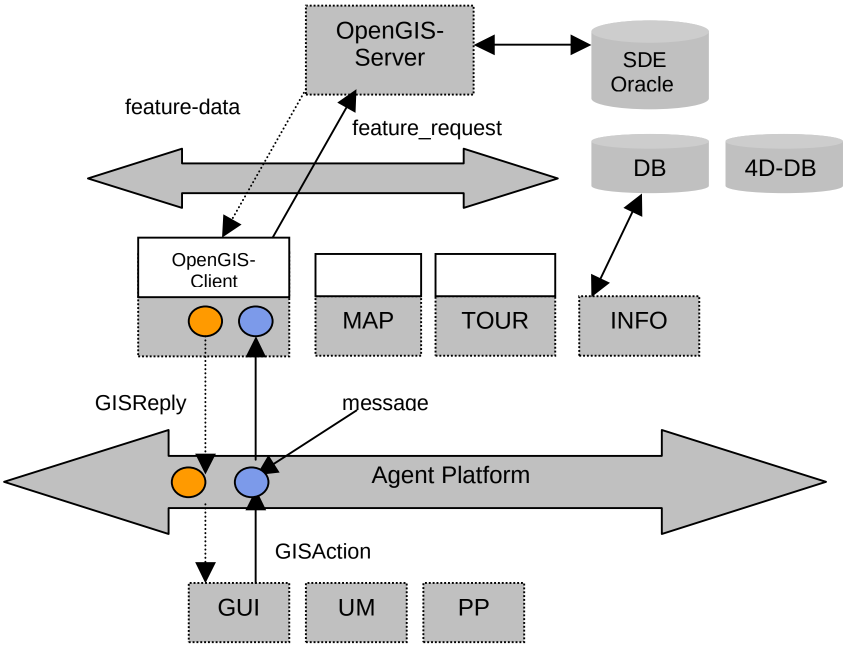

Figure 2: Agent communication infrastructure (via FIPA-ACL) in Deep Map with integrated OGC services using CORBA/IIOP

As an example, a “Spatial Agent” with integrated access to GIS services via the SFS for CORBA has been implemented wrapping a wide range of spatial functions from spatial queries and selections, distance measures, export of geometry data to visibility analysis [Zipf & Aras 2001]. Such an agent driven distributed GIS has a three-tier-architecture. Within the application tier the more complex functions are performed through collaborating agents, where the service layer is the middleware implementing the OGC interfaces.

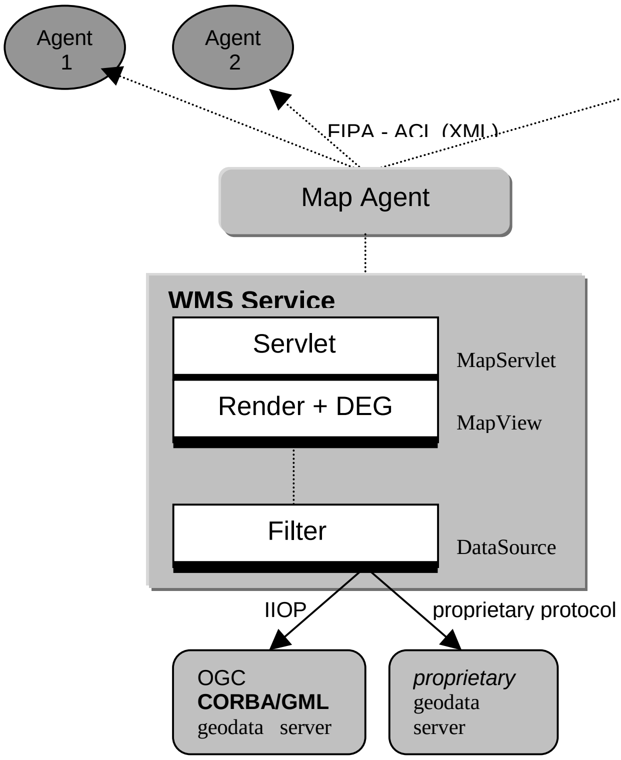

Figure 3 : Exploiting adaptive maps through a Map Agent

Additionally a WMS (Web Map Server) is being developed according to the OGC specifications and has been wrapped by a map agent in a similar way (figure 3).

But the OGC WMS specification [OGC 2001] defines only a limited set of parameters for requesting maps. In order to be able to generate more sophisticated maps adapted to location, context or user interest the WMS specification must be extended by a range of parameters. These enhancements include the possibility not only to specify predefined styles for each feature type, but to specify a whole XML sub tree. An example is the possibility to specify the orientation of the map, in order to be possible to align the map to the walking direction of the user, what seems to be a useful feature for LBS when one considers how some people in real live sometimes turn around the paper maps to match and align their mental map with the current location and the map representation. The heterogeneous user groups using LBS make more sophisticated map adaptation necessary. A step towards building a model of user centric map generation for LBS is explained in [Zipf 2002] and further possibilities to exploit the spatial context proactively are given in section 4 (Future Work).

5. Exploiting the Spatial Context

In the example of our digital tour guide a “Spatial Context Agent” shall give tips in an proactive - but non-intrusive - manner, if a user is close to a sight or object he might be interested in. In order to do this, the Spatial Context needs to take the user interests into account as well as his position and the structure of the surrounding environment. Buildings suitable for acting as landmarks should be priorised, i.e. if the user is reasonable close to a church or a restaurant it might be more suitable to return the address and name of this building instead of explaining to the user that she is closest to “Church Street 218”, which is a not easily to recognize building. The SCA therefore would need to solve some problems regarding location, time and user interests (in the current implementation only a subset of these are realized, but we want to discuss here the applicability and relevance of these):

Location	What is meant to be *near* objects ? Are they visible to the user? What kind of variables determine near objects (user condition, trafic mode …)?
User interests	The relevant objects should be ordered according to user interest, distance etc How shall these different parameters be weighted against each other?.
Time	The processing might take some time – so the result set of the relevant objects must still be in the determined region.

To gain the relevant contextual information the Spatial Context Agent (SCA) (as modelled in the UML diagram below) has to query the following agents:

- *User Model Agent*: information on the current users interests.
- *Spatial Agent*: objects in a region near the actual user location.
- *Sights Agent*: the taxonomic information classifying the identified object.

In order to explain this in more detail, we now sketch the SCA’s communication:
After the activation the SCA will subscribe or location based triggers respectively by the positioning system. As input data the positioning system reveals the current position of the user or it delivers a region-trigger, when the user leaves or enters a specified location. So the SCA can start processing the relevant information with reference to the user position. Every time a new region is entered by the user the SCA will be activated. On (re-)activation the SCA will gather the needed information by sending messages to the involved agents. Hereby the SCA can follow different strategies to find out the most user- relevant sight. The User

Model Agent and the Sights Agent are queried in parallel. After the reply of the User Model Agent, the objects specified as being of interest for the user are extracted from the sights database. After that distance of the objects to the user is calculated. Now the SCA can start processing and weighting the data:

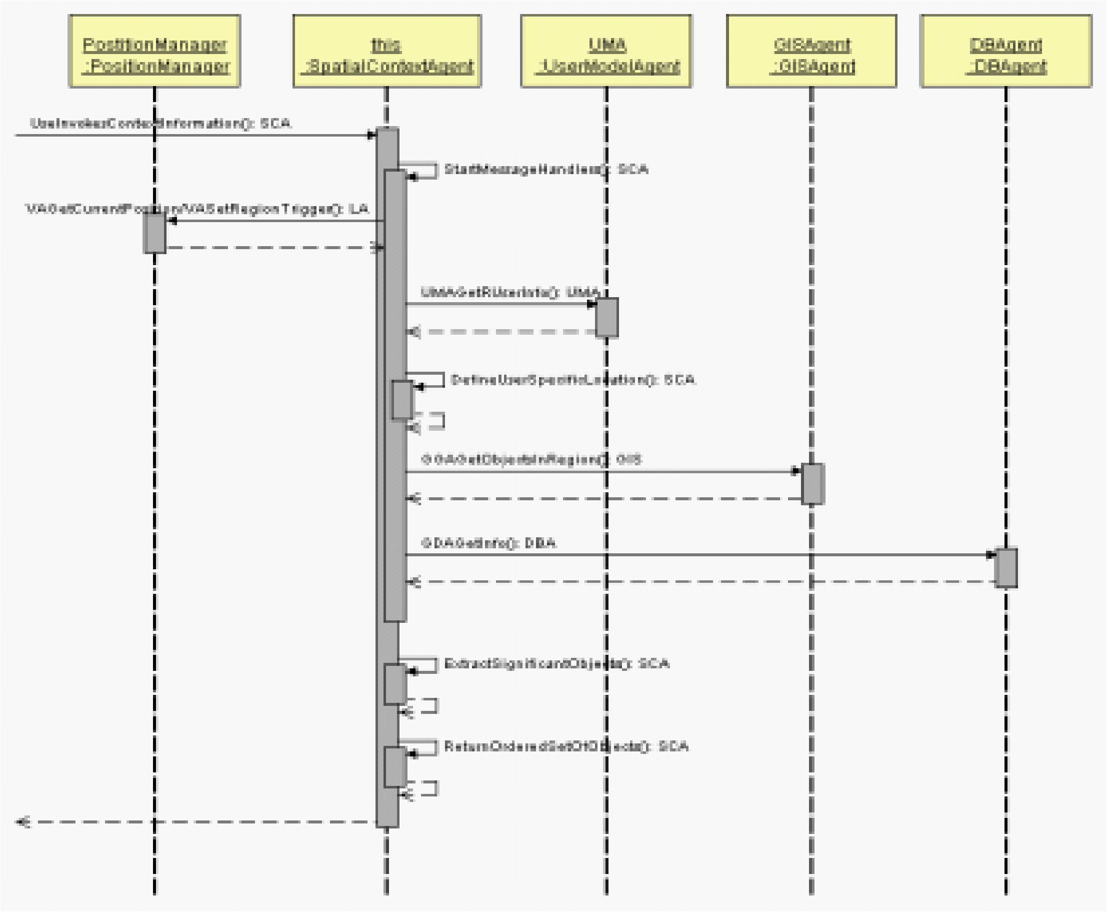

Figure 3: UML-interaction diagram for Spatial Context Agent

One of the endogenous variables the SCA has to determine is the area which should be defined as being “close” to the user. In the first prototype this region is a polygon directed towards the user’s walking direction. Some parameters that might determine what the user means with "near" are these:

- physical condition of the user (dependent from: age, fitness, handicapped, walking duration) transportation (foot, bus, bike, car, wheelchair,…).
- weather (when it is raining it should probably be closer.)
- task (near might mean something different to the user when he is asking for a closet or for a good outlook or famous sight).
- how good do I know the region (how good is my mental map? Research on mental maps has shown that perceived distances shrink when the user learns to know the region better).
- structure of the region (flat area with only few houses or complicated setting with confusing number and placement of structures).
- steepness / height difference (going upwards or downwards).

Now that the SCA has determined all the necessary parameters we need to draw the user's attention to these near-by objects of interest in a non-intrusive way, as experience with proactive approaches have shown that acceptance is otherways quite low. There a range of possibilities there and these have to be teted and evaluated in turn. This ranges from presenting a new icon (e.g. a (blinking?) bulb) to the user in order to indicate that the system wants to give a proposal, to starting to present pages with information on the sights immidally. We are in particular interested in a map based approach, meaning that for a example the map changes its appearance in a way that the objects of interest are highlighted in a particular manner or the route to that object is presented. The evaluation of such ways to exploit the spatial context in LBS will be part of future user trials scheduled for summer 2002.

6. Summary and Future Work

This paper presented an overview about some of the spatially enabled modules and services necessary to build personalised location based services for mobile users (within a tourism scenario). In order to achieve a high degree of interoperability and reusability it is necessary that these components follow open standards. Within projects like Deep Map and CRUMPET [Poslad et al 2001] we evaluate for the first time how such *standard-based geo-components* can be integrated into an *agent-based software architecture* to develop personal digital mobile tourism guides based both on the standards of *FIPA* and *OGC*.

Further work on exploitation of the current location in our agent based LBS - next to the mentioned trial and evaluation - will be on the issue of map adaptation including culture-, task- and interest-specific map design [see Zipf 2002]. Maps clearly have great a value for LBS. But in order to facilitate the correct understanding of a map a personalized and intuitive dynamic map service needs be designed. Dynamic map generation and adaptivity is still a challenge for smart systems, as it is a complex task involving cognitive and psychological aspects. An example application where personal interests of the user together with his positional information influences dynamically parameters of the map (like the map extend) is outlined here and will be a future sample application of our system. Influencing the map extend means in this example that the map extend may differ from the originally requested one, if the systems thinks that there are some reasons to show a slightly different map than requested to the user, e.g. because close to the map border there are some interesting sights that wouldn't be on the map in spite of them being interesting to the user (as the system knows from the user model). This would be another example of proactiveness as the user didn't request to show him these sights originally. A similar scenario has already been implemented in a simulation environment in a previous prototype). The quality of the positioning can for example also influence the extend of the map. If the system is not very sure about the actual correct position of the user, the area displayed on the map should be larger to ensure that the actual position is captured. The map should further focus on the area the user is actually most likely to be located, while the broader surroundings can be presented in a generalized manner.

References

Baeijs, C., Demazeau, Y. and Alvares, L. (1996): SIGMA: Application of Multi-Agent Systems to Cartographic Generalization. 7th European Workshop on Modeling Autonomous Agents in a Multi-Agents-World. Eidhoven. January 1996.

Batty, M. (2000) Geocomputation Using Cellular Automata, in Geocomputation. (Eds. S. Openshaw and B. Abrahart, Taylor and Francis, London, 2000), 95-126.

CSELT (2000): JADE, http://sharon.cselt.it/projects/jade/.

CASTOR 2001: CASTOR open source data binding framework for Java; http://castor.exolab.org/

Ding, Y, C. Kray, R. Malaka, and M. Schillo (2001): RAJA - A Ressource-Adaptive Java Agent Infrastructure. Proceedings of Autonomous Agents'01, Montreal, Canada, 2001

ENHYDRA 200: Open Source XML/Java Data binding. http://zeus.enhydra.org/index.html

Ferber, J. (1999): Multi-Agent Systems. Addison-Wesley.
FIPA (1999): FIPA Foundation for Intelligent Physical Agents home page. FIPA, http://www.fipa.org
FIPA (2000a): FIPA Nomadic Application Support Specification. http://www.fipa.org/specs/fipa00014/
FIPA (2000b) FIPA Agent Management Specification. Available at http://www.fipa.org/specs/fipa00023/.
Frank, A.U., Bittner, S., & Raubal, M. (2001): Spatial and cognitive simulation with multi-agent systems. In Proceedings of COSIT'01, Lecture Notes in Computer Science, Berlin. Heidelberg, Springer-Verlag
Kray, C. and Walther, U. (2000): Intelligent Handling of Positional Information.
Malaka, R. and Zipf, A. (2000): DEEP MAP - Challenging IT research in the framework of a tourist information system. In: Fesenmaier, D. Klein, S. and Buhalis, D. (Eds.): Information and Communication Technologies in Tourism 2000. Proceedings of ENTER 2000, 7th. International Congress on Tourism and Communications Technologies in Tourism. Barcelona. Spain. Springer Computer Science, Wien, New York. p. 15-27.
Open GIS Consortium (OGC)(2000): Web Map Server Specification 1.0. www.opengis.org/specs
Openshaw, S. (1991): Some suggestions for the development of AI tools for spatial analysis and modelling in GIS. In: Annales of Regional Science 26. 35-51.
Open Location Services Inititative (OpenLS)(2000): www.openls.org
O'Sullivan, D., and Haklay, M., (2000): Agent-based models and individualism: is the world agent-based?, Environment and Planning A, 32(8), 1409-1425.
Poslad S, Buckle P and Hadingham RG, (2000): "The FIPA-OS agent platform: Open Source for Open Standards", Proceedings of PAAM 2000, Manchester, UK, 355-368.
Poslad, S., Laamanen, H., Malaka, R., Nick, A., Buckle, P. and Zipf, A. (2001): CRUMPET: Creation of User-friendly Mobile services Personalised for Tourism. In: 3G 2001. Second International Conference on 3G Mobile Communication Technologie. March 2001. London UK.
Rodrigues, A. and Raper, J. (1999): Defining Spatial Agents. In: Camara, A. S. and Raper, J (Eds.): Spatial Multimedia and Virtual Reality. Taylor and Francis. London. 111-129.
Schlichter, J (2001): Verteiltes Problemlösen, http http://www11.informatik.tu-muenchen.de/lehre/skripten/ vp-student-ws00.
Stille, W. (2001): Lösungsverfahren für Prize-Collecting Traveling Salesman Subtour Probleme und ihre Anwendung auf die Tourenplanung in dem Touristeninformationssystem Deep Map. Diplomarbeit. Ruprecht-Karls-Universität Heidelberg. Fakultät für Mathematik, Institut für Informatik. Heidelberg.
Woolridge, M. and N. R. Jennings (1995): Intelligent agents: theory and practice. In: The Knowledge Engineering Review 10(2).
Zipf, A. and Malaka, R. (2001): Developing "Location Based Services" (LBS) for tourism - The service providers view. ENTER 2001, 8th. International Congress on Tourism and Communications Technologies in Tourism. Montreal, Canada. April 24-27, 2001. Springer Computer Science.
Zipf, A. und Aras, H. (2001): Realisierung verteilter Geodatenserver mit der OpenGIS SFS für CORBA. In: GIS. 3/2001. GIS - Geo-Informations-Systeme. Zeitschrift für raumbezogene Information und Entscheidungen. Heidelberg. pp 36-41.
Zipf, A. und Krüger, S. (2001): TGML - Extending GML by Temporal Constructs - A Proposal for a Spatiotemporal Framework in XML. Submitted for ACM-GIS 2001. The Ninth ACM International Symposium on Advances in Geographic Information Systems. Atlanta.
Ram, S., Venkatsubramanyan, S., Yool, S., Marsh, S. and Park, J. (1998): SAGE: An intelligent agent based toolkit for discovering and accessing distributed earth observation data. In: Strobl, J. and Best, C.: Proceedings of the Earth Observation & Geo-Spatial Web and Internet Workshop 98. Salzbuger Geographische Materialien. V. 27. Salzburg.
Zipf, A. (2002 accepted): User-Adaptive Maps for Location-Based Services (LBS) for Tourism. Proceedings of ENTER 2002. International Congress on Tourism and Communications Technologies in Tourism Innsbruck Austria. Springer Computer Science. Heidelberg, Berlin.

Acknowledgments

This work has been undertaken in the context of the project Deep Map at the EML, supported by Klaus Tschira Foundation (KTS) and the EU funded project CRUMPET (IST-1999-20147).