A design process for the development of an interactive and adaptive GIS

Mathieu Petit[1], Christophe Claramunt[1], Cyril Ray[1] and Gaëlle Calvary[2]

[1] Naval Academy Research Institute, 29200 Brest Naval BP600, France
{mathieu.petit, christophe.claramunt, cyril.ray}@ecole-navale.fr
[2] Laboratoire d'Informatique de Grenoble, Université Joseph Fourrier, 38041 Grenoble cedex 9 BP53, France gaelle.calvary@imag.fr

Abstract. Geographical Information Systems (GIS) have long bridged the gap between geo-information databases and applications. Although conceptual modelling approaches for GIS have been particularly successful in the representation of the specific properties of geographical information, there is still a need for a better integration of user intentions and usage. This paper introduces a conceptual framework applied to GIS and defined not only as a "*Geographic Information System*", but also as a "*Geographic Interactive System*". This approach extends the conceptual framework of a general purpose mobile interactive system to the geographical context. Beside a description of user tasks and the domain data layout, the proposed framework considers the geographical environment as an additional component to the design approach. The role of the spatial dimension in the design of such an interactive system is illustrated all along the conception of a real-time ship tracking system.

1 Introduction

Conceptual frameworks applied to the design of geographic information systems are oriented to the modelling and manipulation of spatial data. Conceptual frameworks such as MADS [14] or Perceptory [4] encompass conceptual modelling approaches that help represent the semantics of geographic data. The computing architecture implementation, usage context, and users characteristics restrict to the final steps of the design, or rely on existing client-side applications. In order to remain generic, domain concepts and their layouts are separated from the functional and architectural designs of the client side. This level is considered aside, as a front-end to access and manipulate data [10].

Coming along with the growth of geographical data available through the Internet, the range of GIS benefits and usages dramatically increases [11]: users may not be expert of the domain anymore ; platforms that interact with the system are rich and versatile ; usage contexts differ from one user to another and GISs might be accessed in mobile contexts, through the mediation of computing services.

Those emerging constraints make it necessary to adapt current conceptual GIS frameworks. This leads to a change of paradigm, where a GIS should evolve from an "*information system*" to an "*interactive system*". The design process is still oriented to geographic concepts, but another aspect to consider is the use that can be made of the geographical dimension to enrich user experience and interaction with the system.

A user-oriented and responsive GIS, considered as an interactive system [18], should take into account user goals and tasks. In a related work, a GIS has been considered as an interaction tool between multiple users, and that acts as a support to human dialog [16]. Implicit collaboration between several users favors recommendations of a spatial content for tourism applications [6], or allows cartographic layer selection applied to archaeology [12]. Automated adaptation to hardware resources has also been addressed for cartographic display [9], and for mobile navigation systems [3]. These systems consider the spatial dimension as the core of a user interactive process. They might be considered as *Geographic Interactive systems*, as they take into account users needs regarding their respective environments. Thereby, the design of an interactive GIS is a special case of an interactive system design. It appears that these spatial concepts impact the system conception at the functional and architectural levels.

This paper introduces a framework for the development of a contextual interactive system. It is experimented in section 2 to design a sailing race documentation system. Geographic concepts are part of the framework and provide an additional input to the design process (Section 3). This environment is based on several regions of significance at execution time, whose spatial relationships are likely to generate alternative system behaviors (Section 3.1). The contextual setup of the system refines user tasks and goals (Section 3.2). Section 4 illustrates the benefits of the proposed geographical approach. Finally, section 5 summarizes the contribution and discusses future work.

2 Interactive system design principles

At the initial stage of a design process, the user objective should be identified and categorized into a set of *tasks* [1]. These tasks rely on the application domain *concepts*, and qualify the system data and resources.

The user objectives and operative tasks are informally described in a so-called *nominal scenario* (Fig. 1(a)). This documentation is textual based, and summarizes the user needs in the light of the knowledge of the domain experts and system designers. Given a nominal scenario, an interactive system design process follows several conceptual stages:

1. identification and organization of tasks and concepts (Fig. 1(b&c)),
2. software distribution design (Fig. 1(d)), processes and data management implementation (Fig. 1(e)),
3. tasks mapping towards the presentation of functionalities, and user interface development (Fig. 1(h, i and j)).

Nowadays, the wide variety of user platforms available, as well as new ubiquitous usages, stress the need for the integration of *usage contexts* into interactive

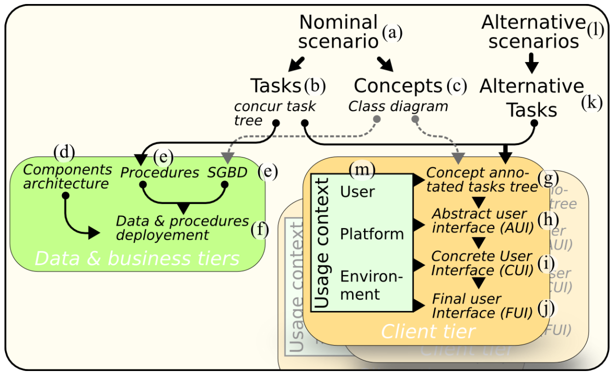


Fig. 1. Design framwork of a context-aware interactive system

system designs [17]. When designing a system, every usage context considered is likely to impact the client-side design (Fig. 1(m)). A usage context is usually described along three orthogonal contexts : user, client platform, and usage environment [8]. Prior to a consideration of this notion of context to a system-wide definition, the next section introduces the early conceptual stage of our case study, from a nominal scenario to the task and concept correlations.

2.1 Nominal scenario and sailing race documentation system

Our approach is experimented in the context of a sailing championship held once a year in Britany in North-West France. This event allways gathers a large audience, from sail fans to newcomers. The race innings often occurs a long way from the shore, and the audience may only notice and follow closest ships. This entails the need to offer a wireless accessible regatta documentation system to a wider audience.

Table 1. Nominal scenario

"The race documentation system runs on a user's PDA and allows her/him to ***follow the regatta*** *in real-time. The PDA provides manipulation tools, and a map of the race area where the racing ships are regularly re-located. The user may* ***be interested in several ships****, or alternaltively* ***by other user interests****, to* ***set her/his own area of interest****. If she/he* ***is interested in a specific ship****, information (year, name, crew and pictures) and real-time data (location, speed and heading) on this ship are provided. When being close enough to the race area, the user* ***takes and shares ships pictures*** *with other users."*

In the proposed case study, members from the audience and the organizing committee are part of the drafting team. Together, they specify the system using a user-centred design approach and summarize their collaboration within the nominal scenario (Tab. 1). When starting a system design from scratch, sequences of tasks and subtasks to meet the user intentions are suggested and integrated into the nominal scenario by designers and HCI experts. In order to to fully comprehend the designed software functionalities, and to complete the task model, the overall design process should be repeated several time. Each iteration adjusts and enriches the prototype, to eventually meet the user needs and intentions [5].

2.2 Domain tasks and concepts

A nominal scenario supports the identification of the main concepts and user tasks of the domain. An object modelling formalism lays out the concepts underlined in the nominal scenario. We retained a UML class decomposition as an initial concept structuring template. For example, the concept of "ship" is defined by data like "name", "type" or "year" and is implemented within the class "Boat".

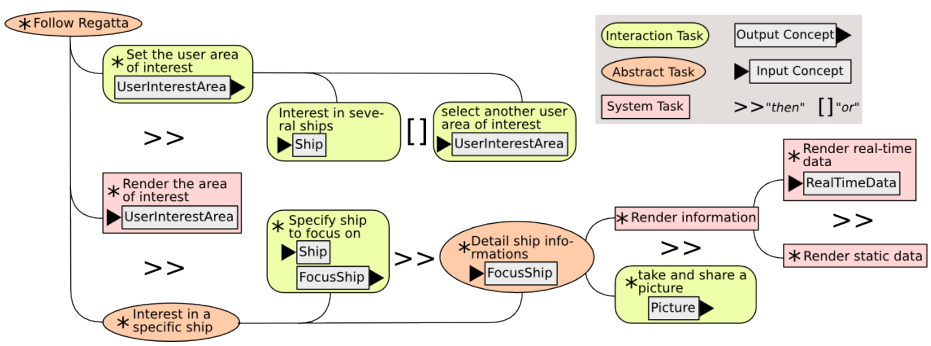


Fig. 2. Task tree of the race documentation system

From the scenario, designers emphasize user tasks. For example, the main task "Follow the regatta" incorporates subtasks such as "Set the area of interest", "Interest in a specific ship" or "take and share pictures". The CTT notation (Concur Task Tree) defines the relationships that organize the tasks, from the user intention level, to the interaction and system levels (Fig. 2)[13]. From a software perspective, each concrete task should be implemented by appropriate procedures and methods, using a semi-automated process [7]. The task tree nodes might be associated with input and output concepts handled by the considered task or sub-task. For example, the interactive task "Set the user area of interest" defines the "UserInterestArea" concept from an input concept, being either a set of ships or another user area of interest.

3 Environment modelling

To the best of our knowledge, interactive system design methodologies limit the impact of a dynamic context to user-related concepts. The main principle is that a context is worth measurement if it influences the execution of system at the user level. In the described generic framework (Fig. 1), only the final design stages integrate such contextual constraints. Besides, when distributed on a multi-components architecture, an interactive system is related to a changing spatial environment for each of its components.

Therefore, a readily adaptable system strongly relies on accurate environmental conditions, and regularly adapts its functional level at running time. Our extended design framework integrates the system distribution as a foundation for functional behaviour derivation. The *impact of the system spatial environment on the course of the nominal scenario* is grasped at every conceptual stage (Fig. 3).

3.1 Extended design framework

In the context of mobility, a runtime environment evolves in an almost continuous mode. These changes are characterized at the design level using a geographical approach to model the system components and their evolution (Fig. 3(a)). This description allows the designer to derive location-based assertions such as "the task T_x can be done within the region R_y" or "the concept C_x is available within the region R_y, and comes from the region R_x"(Fig. 3(b)).

This extended framework derives the set of possible spatial configurations of the environment, according to the spatio-temporal mobility of the system components (Fig. 3(c)). The course of the nominal scenario depends on the validity of these configurations. A description of these environmental states reports the scenario restrictions at the task level (Fig. 3(d)).

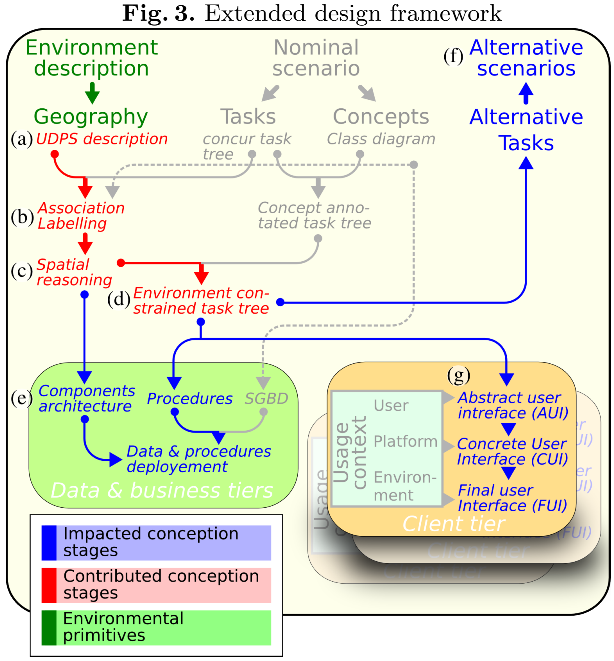
Some of the spatial configurations derived do not constraint any part of the nominal scenario. These spatial configurations characterize alternative execution environments. Initially, they were not stated in the nominal scenario, but they might be of interest and integrated into the design process. In order to support these alternative executions, the task tree is provided with new tasks. This implies the presence of inherent *alternatives* to the nominal scenario (Fig.3(f)).

The environment characterization influences the architecture design at the system level, and the interaction design at the client level. This architecture fits the environment when designed to comprehend contextual changes at the components level (Fig. 3(e)). On the other hand, the tasks tree is the primitive of the client side design, where nominal and alternative tasks are integrated within an integrated and consistant user-interface (Fig. 3(g))[2].

3.2 Runtime environment

The runtime environment is derived from the spatial distribution of the GIS components. This allows to distinguish several system states, and to update the

Fig. 3. Extended design framework



functionality and data available to the user. In a previous work, several regions of significance have been defined to characterize the system at runtime [15]:

- processing region(s) P_x, where the procedures for the completion of a given task are available to the user,
- broadcasting region(s) D_x, where the concepts are available to the system,
- user(s) region(s) U_x, where the user is located,
- source region(s) S_x, where the concepts come from.

Significance regions The set of regions located in space at a given time defines the *runtime environment* of the interactive system. In the case of a sailing race documentation system, data providing heading, speed, and coordinates come from the regatta area $\big(\text{Fig. } 4(S_1)\big)$. The region where the user is located is defined by its immediate environment, so called interaction space $\big(\text{Fig. } 4(U_1)\big)$.

The processing and broadcasting regions are constrained by the capabilities to physically implement the system components. Real-time positioning data are collected through wireless communications $\big(\text{Fig. } 4(\ast)\big)$. During a regatta, sailors are not allowed to access the system concepts. They are broadcasted in a limited region, far from the race area $\big(\text{Fig. } 4(D_1)\big)$. The processing procedures on location-based concepts are accessible to the audience, close to the arrival of the regatta $\big(\text{Fig. } 4(P_1)\big)$.

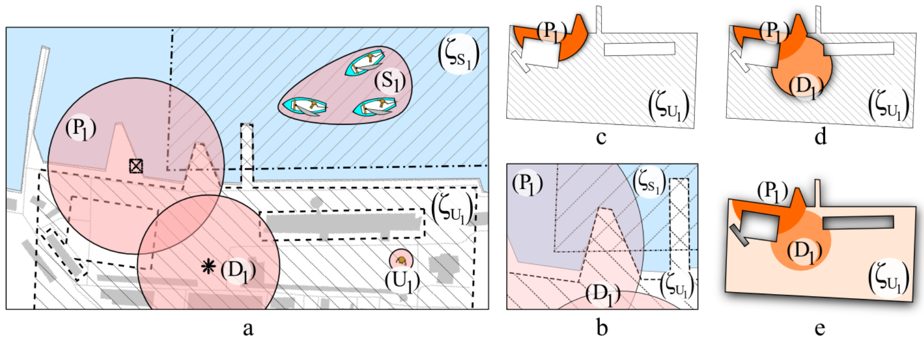


Fig. 4. Significance regions example

Tasks and concept labelling In order to identify the region in the environment that supports a given task execution, tasks and concepts are *labeled* by their respective regions of influence. Let us consider an interactive task $Task_x$, that processes the concept $Concept_x$ in and out. Assuming that, $Task_x$ and $Concept_x$ are up for usage in regions P_x et D_x, respectively, then, region P_x labels $Task_x$ and region D_x labels $Concept_x$.

Procedures that code a given task perform accurately when their required concepts are accessible. In the proposed example, the task $Task_x$ is *runnable*, provided an access to $Concept_x$. Regarding space, this situation occurs when the tasks and concepts labeling regions intersect. In that case, where $P_x \cap D_x \neq \emptyset$, a user standing in P_x might complete the runnable task $Task_x$.

In the sailing race documentation system, only one processing region and concept broadcasting region are defined. All the task procedures are available in P_1, and every concept of the domain is accessible in D_1. As $P_1 \cap D_1 \neq \emptyset$, the concepts are available inside P_1 whatever the considered task is, and the whole nominal scenario is executable in P_1.

Dynamic environment and spatial reasoning At the conceptual level, the spatial properties of significance regions characterize the environment variability at execution time. Several alternative system behaviours are derived when the user cannot perform the tasks from the nominal scenario. This gives rise to unhandled environmental configurations, and provide guidelines towards alternative usages recognition [15].

In a changing environment, the regions of significance evolve. Given a region R_x, let the *mobility area* ζ_{R_x} denotes the set of possible locations of R_x during the system uptime. When $R_x \subsetneq \zeta_{R_x}$, the region R_x is *mobile*. Conversely, this region is *fixed* when $\zeta_{R_x} = R_x$. At a given time of the execution, the *system state* is characterized by the set of intersecting regions of significance. For example, the initial system state for the regatta case study (Fig. 4(a)) is as follows:

$$\{P_1 \cap D_1 \neq \emptyset, P_1 \cap U_1 = \emptyset, P_1 \cap S_1 = \emptyset, U_1 \cap D_1 = \emptyset, U_1 \cap S_1 = \emptyset, D_1 \cap S_1 = \emptyset\}$$

A tabular notation summarizes this formalism $\big($Fig. 5(a-*first matrix*)$\big)$. Per convention, a black cell represents an intersection between the regions of significance.

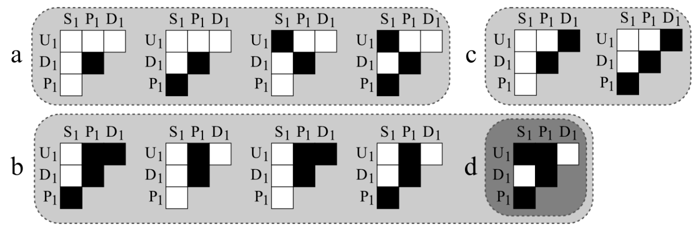


Fig. 5. Spatial configurations of the sailing race system

Any system described by evolving regions generates a countable set of system states. This gives the boundaries of the whole range of spatial configurations. The system is *highly constrained* when every mobility area ζ_{R_x} with $R_x \subsetneq \zeta_{R_x}$ stands apart, and when the intersecting regions are fixed. In that case, the system is characterized by a spatial configuration and a system state. When all mobility areas ζ_{R_x} intersect with every other, the system is *unconstrained*. In that case, and given the number of regions $|R|$, $2^{|R|}$ spatial configuration and system states are identified. Consequently, a *partly constrained* system is characterized by a range of 1 to $2^{|R|}$ system states.

In the documentation system, where the user and the race area are mobile in their respective areas $\big($Fig. 4(ζ_{U_1}) and (ζ_{S_1})$\big)$, several configurations are identified (Fig. 5). These configurations take into account the P_1 and D_1 regions bounded intersection, and the impossible simultaneous intersection of U_1 with D_1 and S_1.

3.3 Spatial configurations and task constraints

In the regatta documentation system, the entire scenario might be executed by a user located in the region P_1. However, as the user remains in the area ζ_{U_1}, only a part of the processing region is accessed $\big($Fig. 4(c)$\big)$. Moreover, only five out of eleven spatial configurations characterize the situation of a user running a task in P_1 $\big($Fig. 5(b)$\big)$.

In an adaptive prospect, every configuration should fall into a system state. In the documentation system, the nominal scenario and tasks are enriched with alternative system states built on top of three spatial configurations groups. When the user is in D_1 without being in P_1 $\big($Fig. 5(c)$\big)$, the system provides a summary of the concepts ships and crews. The useful area covers both regions $\big($Fig. 4(d)$\big)$. When the user is outside the component supported regions D_1 and P_1 $\big($Fig. 5(c)$\big)$, she/he is provided with a minimap of the system coverage. At last, the user proximity to the racing area (*"when being close enough to the race area"* in the scenario) occurs when she/he stands in the race area, that is when $U_1 \cap S_1 \neq \emptyset$. $\big($Fig. 6(c)$\big)$.

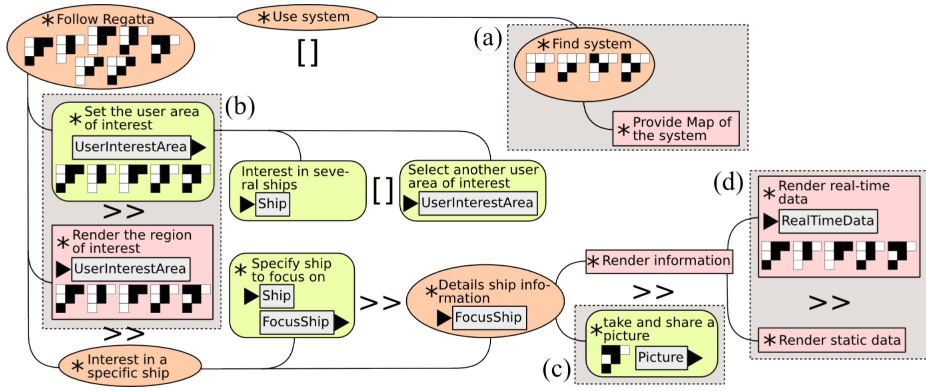


Fig. 6. Environment constrained tasks tree

These spatial configurations become pre-requisites for tasks execution. They annotate tree nodes using their environmental requirements. For example, registering and rendering the user focus can be done when the system components spatial configuration belong to the group *(a)* in figure 5 (Fig. 6(b)). When the user stands in the broadcasting region D_1, only the path leading to the subtask "Render static data" is executable (Fig. 6(d)). In order to enable the task "take and share a picture", the user region has to intersect with the source region An unforeseen task completes the tree and is available when the user is outside all the regions (Fig. 6(a)). The abstract task, "Find system", is iteratively operated on the user platform. Reported at the top design level, this new task outlines an implicit alternative scenario to the nominal case.

4 Prototype implementation

The task tree allows every system state to be part of a common user interface at the client level. Besides, the environmental conditions at the tree nodes level identify the procedure distribution. For example, the task "Provide map of the system" is enabled when the user stands outside of other regions. In that case, the task implementing procedures runs on the client platform.

The implementation of the proposed sailing race documentation system is still in progress. In order to give a brief overview of the intended results on the user platform, an illustrative walk of a user in the area of the system shows the relation between the system states and the functionality offered at the user-interface level (Fig. 7).

When being outside of any other region; the user is provided with a map of the system coverage(Fig. 7(a) - enabled task : "Provide a map of the system"). When the user comes in the broadcasting region at t_2, a sequential-access list of the ships shows-up (Fig. 7(b) - enabled task path : "Interest in a specificship" → "Render static data"). In the processing region, boats are mapped and their

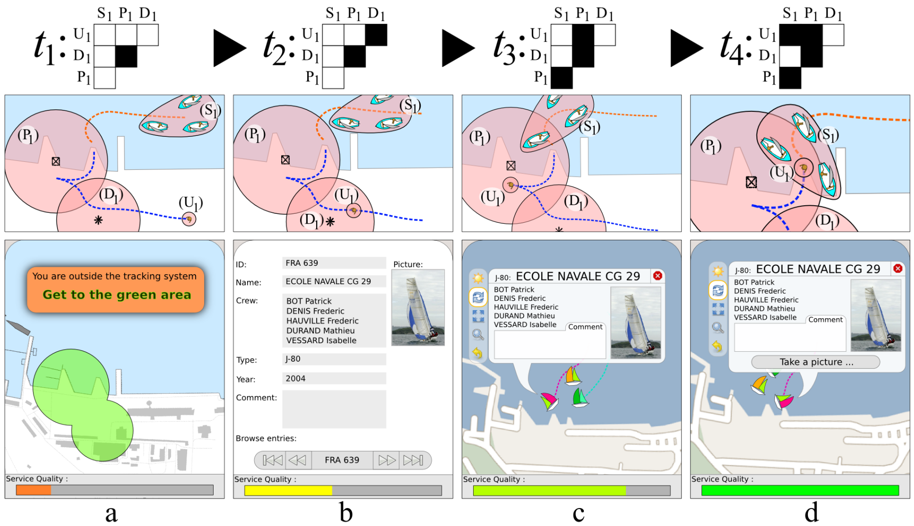


Fig. 7. Evolution of the system behavior in a dynamic environment

position are regularly derived. The user has an access to the detailed information by clicking on the displayed boats (Fig. 7(c) - this enables every tasks except "Take and share a picture"). Finally, when the sailing ships come near the shoreline, the user gets into the source region, and the task "take and share picture" is made accessible trough an enriched user interface (Fig. 7(d)). Given a small region available for services (Fig. 4(c)), the environment characterization has led to alternative system usages. Consequently, at the implementation stage, the useful area has been extended to cover entirely the user mobility area (Fig. 4(e)).

5 Conclusion

Coming along with growing interest in information technologies, progress made in ubiquitous computing and data management led GIS to the edge of interactive and information systems. While new usages of geographic information are emerging, novel design methodologies should be explored in order to move forward *geographic interactive systems*.

The research presented in this paper introduces a geographical extension to an interactive system design framework. From early conceptual stages, the spatial dimension of the environment is integrated within a user-centred design approach. The design process takes into account the runtime environment of the system. This enriches the nominal scenario by the generation of additional usages. The approach has been illustrated by the design of a distributed GIS, and work is still in progress regarding the validation in a real experimental context. Future work concerns the integration of the temporal dimension within the framework, and the design of the usage context at the user level.

References

1. R. Bastide, D. Navarre, and P. Palanque. A tool-supported design framework for safety critical interactive systems. *Interacting with Computers*, 15(3):309–328, jun 2003.
2. J.M.C. Bastien and D.L. Scapin. Evaluating a user interface with ergonomic criteria. *International Journal of Human-Computer Interaction*, 7(2):105–121, 1995.
3. J. Baus, A. Krüger, and W. Wahlster. A resource-adaptive mobile navigation system. In *Proceedings of the 7th international conference on Intelligent user interfaces: IUI'02*, pages 15–22, 2002.
4. Y. Bédard. *Geographical Information Systems: Principles, Techniques, Application and Managements*, chapter Principles of Spatial Database Analysis and Design, pages 413–424. Wiley, 1999.
5. B Boehm. A spiral model of software development and enhancement. *SIGSOFT Softw. Eng. Notes*, 11(4):14–24, 1986.
6. B. Brown, M Chalmers, and I. MacColl. Exploring tourism as a collaborative activity. Technical report, Equator-02-018, September 2002. http://citeseer.ist.psu.edu/brown02exploring.html.
7. S. Caffiau, P. Girard, D. Scapin, and L. Guittet. Generating Interactive Applications from Task Models: A Hard Challenge. In *Task Models and Diagrams for User Interface Design*, volume 4849 of *Lecture Notes in Computer Science*, pages 267–272. Springer Verlag, 2007.
8. G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, L. Bouillon, and J. Vanderdonckt. A unifying reference framework for multi-target user interfaces. *Interacting with Computers*, 15(3):289–308, 2003.
9. M. Hampe and V. Paelke. Adaptive maps for mobile applications. In *Proceedings of the Mobile Maps Workshop at MobileHCI'05*, 2005. http://www.ikg.uni-hannover.de/publikationen/publikationen/.
10. P. A. Longley, M. F. Goodchild, D. J. Maguire, and D. W. Rhind. *Geographic Information Systems and Sciences*. Wiley, 2nd edition, 2005.
11. M.R. Luaces, N.R. Brisaboa, J.R. Paramá, and J.R. Viqueira. A generic framework for gis applications. In Y-J. Kwon, A. Bouju, and C. Claramunt, editors, *Proceedings of the 4th International Symposium on Web and Wireless Geographical Information Systems*, pages 94–109. Springer Verlag, 2004.
12. E. Mac Aoidh, E. Koinis, and M. Bertolotto. Improving archaeological heritage information access trough a personalized GIS interface. In J. Carswell and T. Tekuza, editors, *Proceedings of the 6th International Symposium on Web and Wireless Geographical Information Systems*, pages 135–145. Springer Verlag, 2006.
13. G. Mori, F. Paterno, and C. Santoro. CTTE: support for developing and analyzing task models for interactive system design. *IEEE Transactions on Software Engineering*, 28(8):797–813, aug 2002.
14. C. Parents, S. Spaccapietra, E. Zimànyi, P. Domini, C. Plazent, and C. Vangenot. Modeling Spatial Data in the MADS Conceptial Model. In *Proceedings of the 8th international symposium on Spatial Data Handling: SDH'98*, pages 130–150, 1998.
15. M. Petit, C. Ray, and C. Claramunt. A contextual approach for the development of GIS: Application to maritime navigation. In J. Carswell and T. Tekuza, editors, *Proceedings of the 6th International Symposium on Web and Wireless Geographical Information Systems*, pages 158–169. Springer Verlag, 2006.

16. I. Rauschert, P. Agrawal, R. Sharma, S. Fuhrmann, I. Brewer, and A. MacEachren. Designing a human-centered, multimodal gis interface to support emergency management. In *Proceedings of the 10th ACM International Symposium on Advances in Geographic Information Systems*, pages 119–124. ACM Press, 2002.
17. M. Satyanarayanan. Fundamental challenges in mobile computing. In *Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing*, pages 1–7, 1996.
18. Y. Yupin and L.J. Shrum. What is interactivity and is it always such a good thing? implications of definition, person, and situation for the influence of interactivity on advertising effectiveness. *Journal of Advertising*, 31(4):53–64, 2002.