



Adaptive on-line information system by means of a task model and spatial views

Serge Garlatti, Sébastien Iksal, Philippe Kervella

► To cite this version:

Serge Garlatti, Sébastien Iksal, Philippe Kervella. Adaptive on-line information system by means of a task model and spatial views. Second Workshop on Adaptive Systems and user Modeling on the World Wide Web, May 11-14, Toronto (WW8) & June 20-24, Banff (UM99), May 1999, Toronto, Canada. pp.59 - 66. hal-02142952

HAL Id: hal-02142952

<https://hal.science/hal-02142952>

Submitted on 29 May 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptive On-Line Information System by means of a Task Model and Spatial Views

S. Garlatti¹, S. Iksal¹, P. Kervella²

¹ [Laboratoire IASC](#), [ENST Bretagne](#), ZI de Kernevent, BP 832,
29285 BREST Cedex , France, Tel : 33 2 98 00 14 53, Fax : 33 2 98 00 10 30,

Email : Serge.Garlatti@enst-bretagne.fr

<http://www-iasc.enst-bretagne.fr/~garlatti>

² Atlantide, Technopôle Brest Iroise, Site du Vernis,
CP n°2, 29608 Brest Cedex, France

Abstract: SWAN project (Adaptive and Navigating Web Server) aims to design adaptive web servers for on-line multimedia information systems about nautical publications. It is a joined project between the IASC laboratory and a private company called Atlantide. The project is funded by the west region council and supported by the French naval hydrographic and oceanographic service. In on-line information systems, users used to only access a fragment of information space according to their current goal. User's goals can provide navigation help [1-8]. In our framework, user modelling is based on stereotypes and more precisely on prototypical user's tasks and on user's class. It also uses a domain model, an individual user's model and the navigation context features. The content and presentation adaptation is achieved by the user's class. The task model is used to design navigation processes, to define views of hyperspace. The task model structures the information access to make the navigation easier. A task determines the relevant domain concepts available in a particular geographical area - due to the structural organisation of nautical publications. The goals of adaptation, adaptation technologies, user features, strategy to define spatial views depend on the current task.

Keywords : on-line information system, adaptive web server, user modeling, task model, domain model.

1. Introduction

Due to new technologies in telecommunication, most of information systems are available through Internet or Intranet. They are based on one or more web servers. Web servers supply with hypermedia tools for user-driven access to information. Information retrieval mechanisms are based on browsing into database or information systems instead of writing queries. One of the main advantages of browsing for users, in general, is that they are better to recognise the information they search that to characterize it in advance [2]. Nevertheless, hypermedia systems have some drawbacks : a user may become hopelessly lost in hyperspace when browsing in a large information space [9]. Then it is necessary to assist user's navigation for information retrieval. Reducing information space to access relevant information needed by users is a well known method to prevent from getting lost in hyperspace. In on-line information systems, users used to only access a fragment of information space according to their current goal. User's goals can provide navigation help [1-8].

SWAN project aims to design an adaptive web server for on-line information systems about nautical publications by means of user modelling. First of all, we present the SWAN project and its requirements. Secondly, user modelling and task model are introduced. Thirdly, we go into details about two particular users's goals. In conclusion some discussions and perspectives are considered.

2. SWAN Project

SWAN project (Adaptive and Navigating Web Server) is a joined project between the IASC laboratory and a private company called Atlantide. The project is funded by the west region council and supported by the French naval hydrographic and oceanographic service. At present, sailors have to find out the relevant pieces of information in different categories of publications (sailing directions, lists of lights and fog signals, tide and streams publications and Radiosignals publications, ..). All these publications are geographically organised. For sailing directions, it is

organised around the notion of area which is recursively divided into sub-areas. There are approximately five levels of areas. At each level, there is a general information part which is relevant for all sub-areas. These multimedia publications are composed of texts, photos, tables, drawings, charts and plans.

The on-line information system will provide nautical information available in different types of publications for different classes of sailors and vessels to prepare a maritime navigation or to navigate on oceans. In order to acquire users' goals and how they achieve them with paper publications, we made free and directed interviews of some different categories of sailors - military, shipping, commerce and yachtsman. This study showed that sailors have a common set of goals which are stables and are achieved in very similar way. Nevertheless, some of them have special goals related to their particular trade - for instance vessels which lay underwater cables for telecommunications or hydrographic vessels. We also investigated nautical publications for acquiring domain knowledge. In the first version of the web server, we decided to study these common goals, named services. Task analysis of services showed that it is quite natural to represent them by a hierarchical task model [10]. According to this study, a task model is used for user modelling.

Some aspects of the current version of this on-line information system is intentionally simple. Indeed, all future users don't make a habit of utilising computer-based systems to achieve their daily tasks. Consequently, it is important to firstly design a software which is not too far from the current software in nautical domain. The purpose of the first version is also to show the benefits of adaptive on-line information systems to sailors, to acquire new goals for the systems and to suggest sailor propositions and comments before going further. For the same reason, the task model is masked to the users. Now, we present user modelling enabling us to design the adaptive web server.

3. User modelling

Stereotype, introduced by Rich [11], is an important element of user modelling and it has been extensively used because it gives a simple but powerful way for adaptation [12, 13]. In our framework, the user's model is based on stereotypes and more precisely on prototypical user's tasks and on a user's class. (cf. Fig. 1).

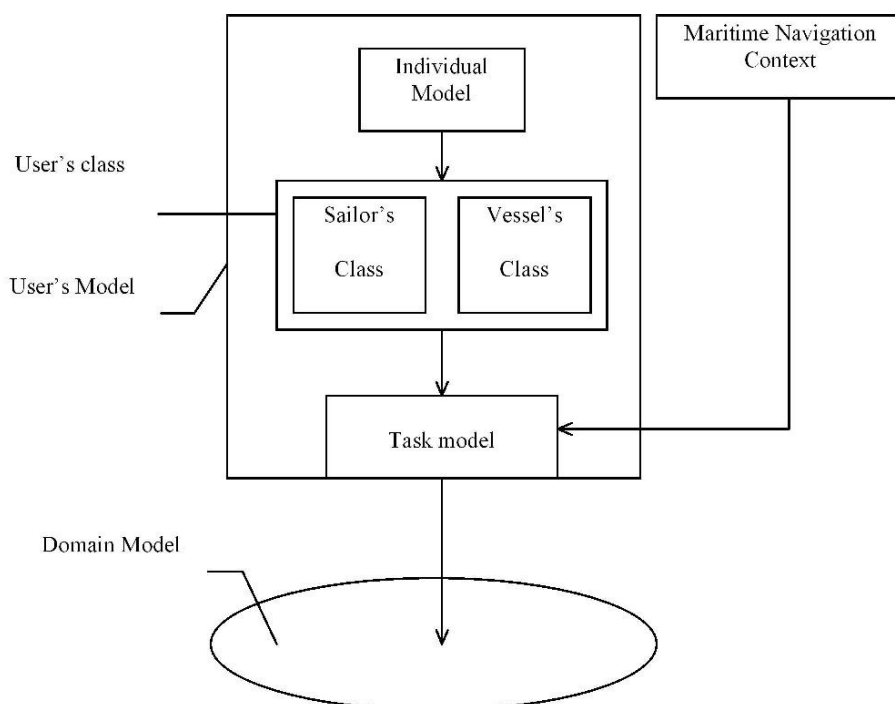


Fig. 1 User's Modelling ([full size](#))

The user's model is composed of a user's class, a task model and an individual model. Its structure is similar to the user's model of Hynecosum [2]. The user's class consists of a sailor's class and a vessel's class. The former has only one feature, the sailor category which can be professional or yachtsman. The vessel's class features are the following : length, breadth, height, tonnage, draught, navigation category which determines maximal distances from a shelter, vessel type (military, fishing, cargo, yacht, ..). The maritime navigation context consists of a set of navigation condition features : tide, time, weather forecast, general inference, GPS position (Lat/Long) or position chosen by the sailor. The user's individual model enables the sailor to choose an adaptation method for a particular task or to specify some parameters of an adaptation method and to choose the minimal depth of route. Some tasks have a default adaptation method which is annotation. It's possible for a user to choose hiding or partial hiding instead of annotation.

According to Brusilosky, content-level and link-level, called respectively adaptive presentation and adaptive navigation support, are the two main classes of hypermedia adaptation [5]. The sailor's class is used for adaptive presentation and the task model for adaptive navigation. At present, the content and presentation adaptation is achieved in a simple way - for the first version : it depends on the sailor's category: professional or yachtsman ; sailing directions are different for these two user's classes. Adaptive presentation is processed in the same way whatever the task. Adaptive navigation support is achieved by means of a task model which uses the vessel's class, an individual model and the navigation context. Indeed, all tasks are available for each sailor's class. In a next version, we could design specific tasks for particular sailor's classes.

4. The task Model

The task model supplies with adaptive navigation support - it is generally the case. A task may have four roles :

- Determining a view of hyperspace - the relevant information space - according to a user's goal and then offering a small hyperspace to the user in which he can browse. A view, called a spatial view, is an information space which is composed of the data belonging to the domain of a particular geographical area.
- Communicating with the users to get some parameters or data,
- Providing the different " steps " of navigation in a certain context - current goal,
- Defining the adaptation method and its parameters.

According to interviews, we find out four common goals for sailors - named Services - that are sufficiently general and high-level to be stable : route retrieval or creation, route information retrieval, port / anchorage, general information retrieval. Route retrieval or creation helps the sailor to find a route from a port/anchorage to another one. Route information retrieval provides navigation information, regulations, aids to navigation, lights, dangers to navigation, local conditions, currents, ... according to the route chosen. Port / anchorage gives to the sailor information about port entry, anchorage, marinas, facilities, services, etc., and a port retrieval based on the available services in the port and around. General information retrieval provides history of weather, geography, oceanography, country and so on.

Task analysis of services showed that it is quite natural to represent them by a hierarchical task model [10]. In our model, we have two kinds of tasks : abstract and atomic.

- Firstly, *abstract tasks* are used to declare the navigation process, and to build the global and the local guides and orientation. The four more abstract tasks are the services. An abstract task is decomposed into sub-tasks which can be abstract or atomic. A control structure using standard operators determines the sub-tasks ordering. The operators are the following : sequence (and), selection (or) and a specific selection - which allows to have one task in abeyance, only for service's tasks.
- Secondly, *atomic tasks* are used for information retrieval and communication. They are not composed of sub-tasks, in fact they are the leaves of the hierarchical task model. An information retrieval task computes an hypermedia views allowing the user to browse in a small hyperspace. It determines the relevant domain concepts, an adaptive navigation method and a way to compute spatial views according to particular sub-domain spaces. A communication task gives some explanations to the user, specifies some user's needs and gathers data or information from users.

The hierarchical structure is based on a composition relationship between tasks. Tasks have four possible states: opened, closed, in abeyance or in use.

The domain model determines the domain concepts and their relationships which are well-known for advanced and expert sailors. The relevant domain concepts associated to a task specify a hyperspace view which is not sufficiently small. Indeed, the sailor may browse the relevant concepts of all geographical areas. We need to reduce the information space to that is useful according to the vessel's position by means of a spatial view. It is an information space which is composed of the data belonging to the relevant concepts of a particular geographical area. A geographical area can be represented by a polygon on a chart. It depends on the considered concepts and tasks. For some concepts, it corresponds to a sailing direction's area. For navigation aids, it corresponds to a smaller polygon defined a priori or be the results of a computation. It is always included into a sailing direction area. In other words, a spatial view selects the useful data according to the vessel's position and the user's task (cf. fig. 2).

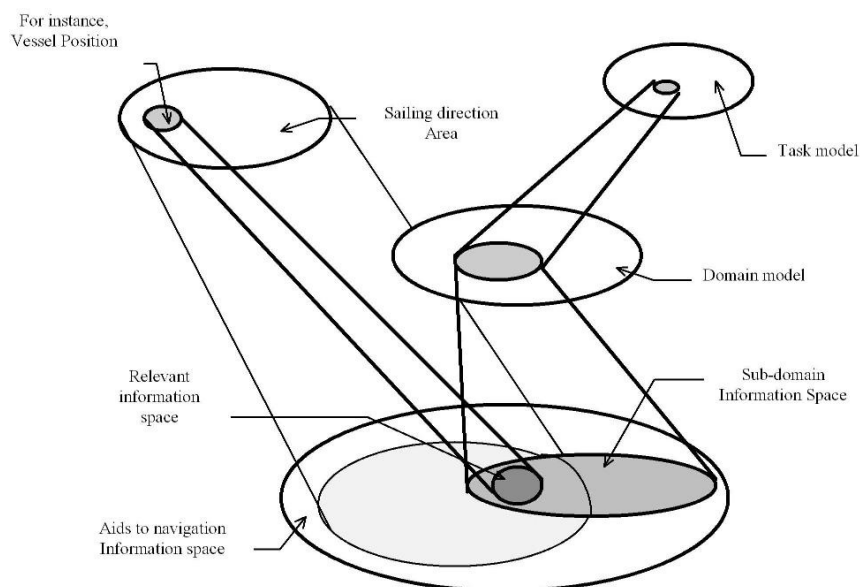


Fig 2. Hyperspace views providing by different parts of the user model ([full size](#)).

5. Task Examples

In this paragraph, we highlight some system features about adaptation : the uses of user's model and navigation context, adaptive navigation and spatial views. Two particular services are partially analysed: the " route creation/retrieval " service and the "route information retrieval " service. The former enables us to mainly show adaptive navigation and the latter adaptive navigation and spatial views.

5.1 The " Route Creation/Retrieval " Service

The aim of this service is to aid the sailor to retrieve or create a route from a port/anchorage to another one (cf. fig. 3).

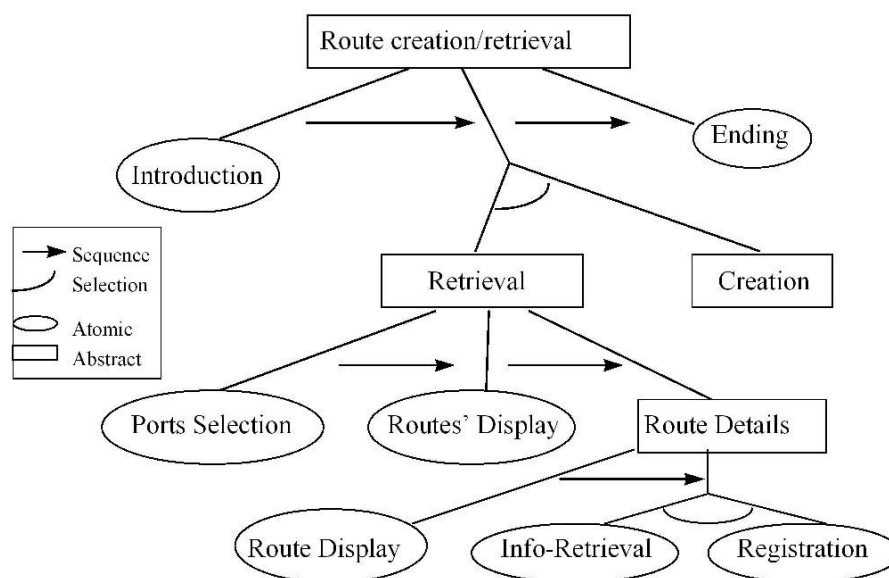


Fig. 3. Graph of the " Route Creation/Retrieval " service ([full size](#))

Each service begins with an introduction task to explain the service's goals and a ending task to close it, mainly for tutorial aspects of the first version. Then the users may choose between the retrieval task or the creation. Whether retrieval task has been chosen and no route has been selected, the user can select the creation task to design a new route. The " Ports Selection " task enables the user to choose two ports/anchorage, " Routes' display " find out the corresponding routes and display the result with an default adaptive method. The user may choose a route and get details about it and maybe want to get route information before validating his choice. The system does not place restriction on port/anchorage choices about maritime regulations or sailor's skills. Whether the route is non-relevant or forbidden, an explanation is provided to the sailor. Annotation as default adaptive method was chosen because the sailors prefer it and it enable us to explain the reasons to sailors.

In "Route Creation/Retrieval" service the relevant domain concepts are defined by the sub-graph root, named "route object" which possesses the following sub-concepts "way-points", "route section" and "route". A route is composed of several route sections which are defined by two way-points, a compass course, a route section type (inshore traffic area, offshore traffic area, landfall, port entry), a minimal depth, a length, a sailing direction area, danger conditions. A route possesses some other attributes like: a departure and an arrival port/anchorage, a route category, a minimal depth, a length and advisable or not.

By means of the departure and arrival port/anchorage, the task retrieves the routes between the two locations and associates to each route a state. There are four possible states: advisable, secure, non secure and forbidden. The route states are computed from the vessel's class features (draught, navigation category, vessel type), the individual model (minimal depth) and navigation context features (tide, tidal streams, time). The default adaptive navigation method is the annotation based on these states. But the user will be able to choose between annotation and hiding - one or more route states - by means of the individual model or could be suggested by the system.

The four annotation states are computed as follows: i) forbidden route: wrong vessel's type, wrong route's category, minimal depth less than the vessel's draught or the user's minimal depth ; ii) non secure route: permitted route, port/anchorage non-allowed for the vessel, due to draught, tonnage or size, forecast an tide conditions leading to dangerous route ; iii) secure route : minimal depth equal or greater than that required by the sailor, permitted route ; iv) advisable route, a secure route which is advised in sailing directions.

5.2 The "Route Information Retrieval" Service

In our framework, two main information categories are provided to sailors: aids to navigation (buoys, lights, seamarks and alignments) and sailing direction content (texts, charts, images, drawings) which come from the sailing directions for professional or yachtsmen and lists of lights and fog signals. The corresponding information space is defined by the sub-graph roots "geographical objects" and "aids to navigation" (cf. fig. 4). Annotation of concepts is used as default adaptive method for the two categories. Concept names of the considered domain sub-graph are annotated. Two states are used relevant and non-relevant - two different colors. For instance, lights are relevant on night in the "Aids to Navigation" category and the others concepts are non-relevant, but accessible. Moreover, another adaptive method is used for elements of the different concept: sorting according to the course section.

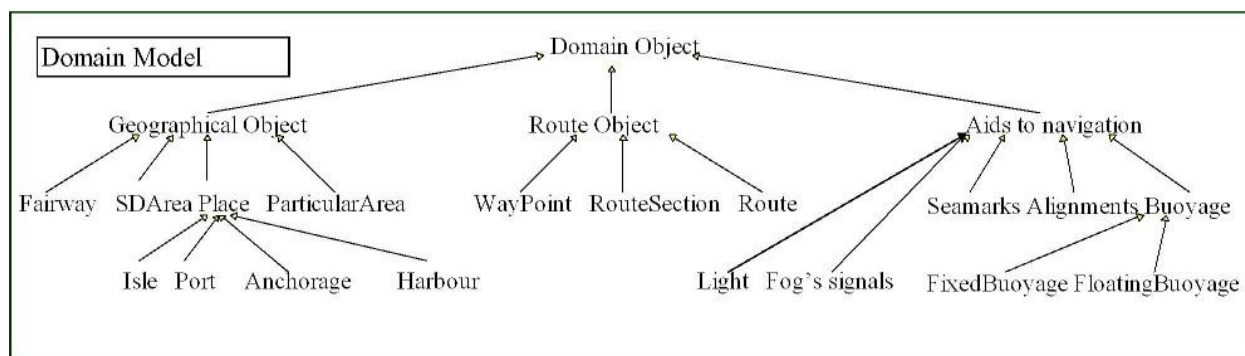


Fig. 4. Domain model ([full size](#))

A spatial view - an information space - depends on the categories and the route section type. It is composed of the data belonging to the relevant concepts of a particular geographical area. A geographical area can be represented by a polygon on a chart. This area depends on route section type for a given category. For a given concept - lights, buoyage, seamarks, fog signals or alignments -, the relevant data are obtained by a filtering process defined by means of the geographical area which reduces the information space to the useful data. We mainly focus on the "aids to navigation" category to highlight the different way of defining spatial views.

In figure 5, the "Route Information Retrieval" service is described. The "Route selection" task is available to the sailor whether he has not previously chosen a route in the "Route Retrieval/Creation" service. Then, he can access the "Retrieval" sub-task to select a route. The "Route Information Retrieval" service is also accessed from the "Route Creation/Retrieval" service, when the sailor has a look for routes' details.

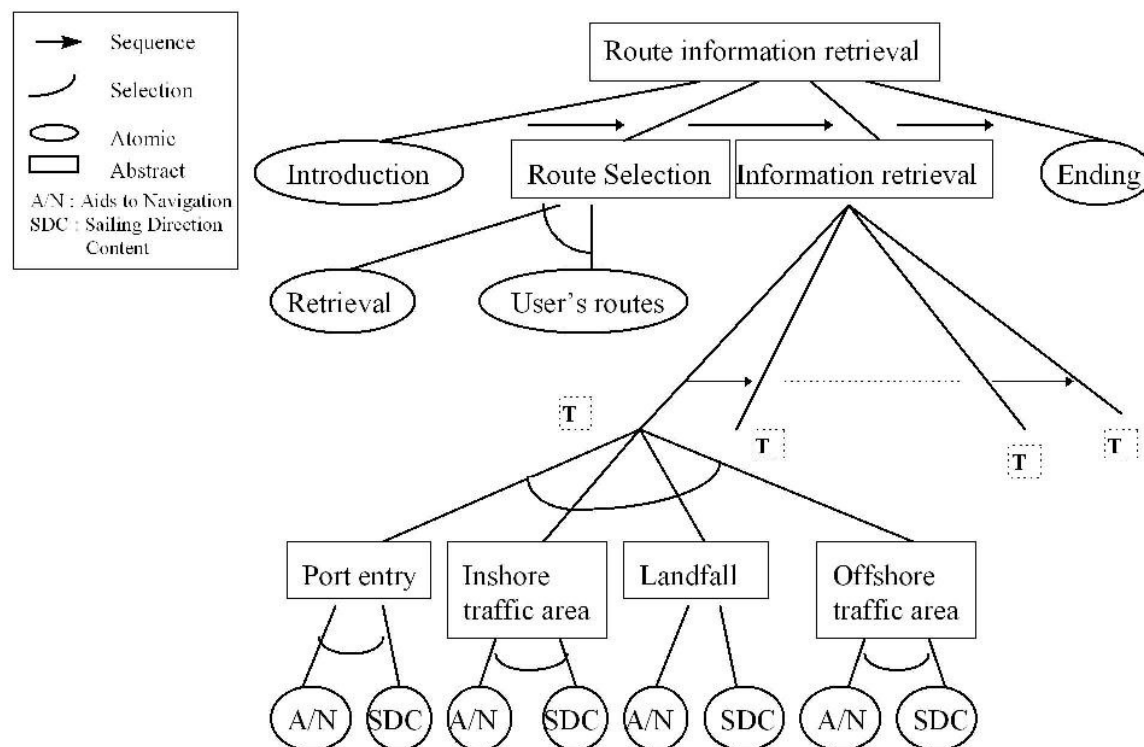


Fig. 5 Graph of the " Route Information Retrieval " service ([full size](#))

Now, we go into details about the " Information Retrieval " task. This task is composed of a sequence of sub-tasks, one task per route section type. A particular task class is associated with each route section type. Each class is composed of two sub-tasks, one per category " aids to navigation " or " sailing direction content " to define the corresponding strategy for spatial views and adaptive method.

In a task, a sailor can access the sub-graph corresponding to navigation's aids. Each relevant leaf is annotated with a specific color and non-relevant one with another color. Concept annotation is based on the weather and time features : fog or not -, day or night. Relevant concepts are computed as follows :

- *Fog* : Fog's signals are relevant
- *Night* : Lights and lights' alignments are relevant.
- *Night and fog* : Lights, fog's signals and lights' alignments are relevant.
- *Day and no fog* : These are buoys, seamarks and alignments are relevant.

Now, we go into details about geographical areas which determine spatial views. For a given concept, a computed or defined - a priori - geographical area is associated to a task class - corresponding to port entry, landfall or inshore traffic area. An adaptive method is also associated with a task class for a given concept - sorting can be process in a different way.

The computation of the geographical area uses the notion of visibility. It takes into account the range and the angular sector of the navigation's aid - angular geographical area where the element is visible regardless of its range -. According to the task class, all visible elements are selected or only those which are visible and available in a particular polygon. The system must consider the three following case :

- *Port entry* : All elements visible - range + angular sector - from the vessel are selected. All the elements are ordered according to the course section : the nearest is in first, the furthest in last. The sailor can choose a buoy and gets its features (text, images, ...) and its orientation according to the course section - North, East, South, West, North-East, ...). Seamarks are not relevant in this case.
- *Landfall* : A sailing direction has a specific chapter for landfall, it describes how to join the inshore area from some particular zones of the offshore area which contained seamarks, alignments and relevant lights for the corresponding area. All elements of the corresponding zone are selected. They are sorted by direction according to the course section: North, North-West, North-East, South, South-East, West, East
- *Inshore traffic area* : The filtering method process in a different way the concepts - buoyage, lights, seamarks, alignments and fog's signals. These concepts are clustered into two classes, aids close to the route section and those far from the route section. The first class consists of buoyage, lights on buoys and fog's signals. For them,

a relevant area is considered around the route section - a rectangle with the length of the section and a user defined width (by default 3 nautical miles) - (cf. Fig.6).

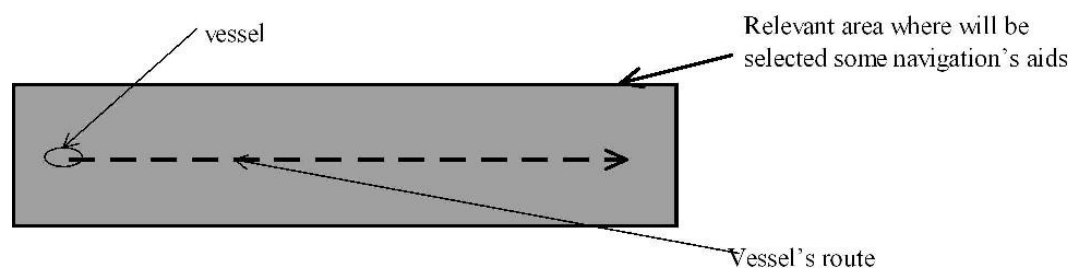


Fig. 6. Relevant area ([full size](#)).

All elements which are in the relevant area and are visible from the vessel are selected. All the elements are ordered according to the course section: the nearest is in first, the furthest in last. The sailor can choose an element and gets its features (text, images, ...) and its orientation according to the route section - North, East, South, West, North-East, ...).

The second class is composed of seamarks and alignments (with or without lights). All elements, visible from the vessel, are selected. They are sorted by direction according to the course section: North, North-West, North-East, South, South-East, West, East.

6. Current Development State

The web server is based on a client/server architecture which is composed of four components : a HTTP server, a dynamic web server called WebObjects [14], a data base system and a knowledge-based system with LOOM [15] - a description logic. The dynamic web server manages users' identity and recognition, builds dynamic web pages and gets all the user's needs in interactions. They are sending to the knowledge-based system which consists of the domain model and the user model. According to these models, it computes the following web pages and its content which are sending to the dynamic web server. At present, the user and the domain's model are ready for the first prototype, the communication protocol between Loom and WebObjects is validated. Now, we are focusing our development on the WebObjects part - design of web pages - and the data recovery.

7. Discussions and Perspectives

Our approach assumes some hypotheses about domain model and task model. Indeed, we assumed that the domain model extracted from sailing directions is well understood by the sailors. We use the domain model to give local guide to the user because it is an abstract and stable view of the domain. For example, when a task has determined a hyperspace view, the corresponding concept names are displayed to the users to select information. Then it is necessary that the user does not misunderstand these names and their interpretation. We need to check this point because the domain structure of the sailing directions may not be the one of the sailors in general or some sailor's classes. We plan to add a mechanism able to only select the well understood concepts of a user or a user's class by hiding some sub-concepts. For instance, naive sailors could have some difficulties to cope with the entire domain model. Then, naive sailor could have a wrong representation of the domain.

The task model has been designed from interviews. But, the system is a new tool and consequently the task model is not based on a real user's task. It was necessary to design new tasks which are extended from the current user's behaviours with paper publications. We need to check and improve the task model. At present the task model is masked to the users because we assume that the user's background is insufficient to deal with the complexity of the task model, its modifications and its design. But maybe, it is important to enable some users to manage the task model and to create new tasks.

In the next stage, we plan to add some features in our adaptive web server : extending user's preferences for adapting dynamically the task model, adapting the relevant zone of a route section which gives us the visible buoys, the strategy to choose the seamarks, finer user's classes - more user's classes -, We plan to adapt the server to the user's knowledge as well as his domain knowledge than his hypermedia's knowledge. But we need to evaluate the current version before modifying in order to take into account the user's comments. Indeed, when a new tool is designed it is difficult to get the user's preferences or needs. Then, it is necessary to design a first version with which

it is possible to get a new tool, closer to the user's needs.

8. Bibliographies

1. Waern, Y., *Cognitive Aspects of Computer Supported tasks*. 1986, New York: Wiley.
2. Vassileva, J., *A Task-Centered Approach for user Modeling in a hypermedia Office Documentation system*. User Models and User Adapted Interaction, 1996(6): p. 185-223.
3. Höök, K., et al., *A glass box approach to adaptive hypermedia*. User Models and User Adapted Interaction, 1996 (6).
4. Encarnação, L.M., *Adaptivity in graphical user interface: an experimental framework*. Computers & graphics, 1995. 19(6): p. 873-884.
5. Brusilovsky, P., *Methods and techniques of adaptive hypermedia*. User Modeling and User-Adapted Interaction, 1996. 6(2-3): p. 87-129.
6. Grunst, G., *Adaptive hypermedia for support systems, in Adaptive user interfaces: Principles and Practice*, M. Schneider-Hufschmidt, T. Kùme, and U. Malinowski, Editors. 1993, North-Holland: Amsterdam. p. 269-283.
7. Tyler, S.W. and S. Treu, *An interface architecture to provide Adaptive Task-Specific Context for the user*. International Journal of Man, Machine and Studies, 1989(30): p. 303-327.
8. Kaplan, C., J. Fenwick, and J. Chen, *Adaptive Hypertext Navigation Based on User Goals and Context*. User modelling and user adapted interaction, 1993. 3(2): p. 193-220.
9. Conklin, J., *Hypertext: An introduction and Survey*. IEEE Computer, 1987.
10. Rasmussen, J., A. Pejtersen, and L. Goodstein, *Cognitive system Engineering*. 1994, New York: John Wiley & Sons.
11. Rich, E., *Stereotypes and user modeling, in user models in dialog systems*, A. Kobsa and W. Wahlster, Editors. 1989, Springer verlag: berlin. p. 35-51.
12. Kobsa, A., *User modeling: Recent Work, Prospects and Hazards, in Adaptive User Interfaces: Principles and Practice*, M. Schneider-Hufschmidt, T. Kühme, and U. Malinowski, Editors. 1993, North-Holland: Amsterdam.
13. Kay, J. *Lies, Damned Lies and Stereotypes: Pragmatic approximations of users*. in Conference on user modeling. 1994: Hyannis, MA: p. 175-184.
14. Apple, *WebObjects*, . 1997, Apple Computer.
15. MacGregor, R.M. *A Description Classifier for the Predicate Calculus*. in Proceedings of the Twelfth National Conference on Artificial Intelligence. 1994: p. 213-220,.