An. Şt. Univ. Ovidius Constanţa Vol. 12(2), 2004, 147–156

MODELING NAVIGATION SYSTEMS BY MEAN OF MULTIMEDIA AND COMMUNICATION TECHNOLOGIES

Elena Popescu and Gheorghe Popescu

To Professor Dan Pascali, at his 70's anniversary

Abstract

It is well known that the navigation simulators are quite expensiv and in many cases the number of students that may attend a training session in a simulator is too small by comparison with what it should be in order to offer the training conditions for a heavy naval trafic. The purpose of this research has been to develop a handy multimedia support for the simulation of the navigation systems, that my be installed from a CD and used in a reqular information laboratory, without diminishing its main destination. This paper deals with the physical model of the navigation systems simulation. In spite of the fact that the users of such a software product represent a rather limited group, the approach of the navigation processes simulation presented in this paper is an exemple of how the spectacular impact of the multimedia and communication technologies may be applied not only in the simulation of the navigation systems but in many other problems where the modeling of a piece of reality requires the use of these technologies.

Before getting started, it is worth to mention that the modeling of the navigation systems takes place in two steps. The first one is to find a mathematical model for the simulation of the navigation process, i.e. to develop the conceptual model of the radar simulation (this must generate more fictitious ships, which interact on each other according to the environmental conditions that the trainer pretends to have in the navigation area and to the wish of their commander) and the second step means to develop an user interface

Key Words: modeling,simulation,e-learning, multimedia, navigational process.

which takes into account the conceptual model developed in the first step but more than that, it looks like and behaves like a navigation system.

The requirements imposed by the second step on the user interface have been presented in another paper published in *Analele Ştiinţifice ale Universităţii "Ovidius" Constanţa, seria matematica, volume XI (2003), fascicola 1.*

This paper deals with the solutions we have used to satisfy the requirements imposed by the second step on the user interface. From the programming point of view this solution is a Visual Basic Project, but from TI point of view, it is a multimedia support for the simulation of the navigation systems. We have named it SINAV (**SI**mulator for **NAV**igation) and its block diagram is presented below.

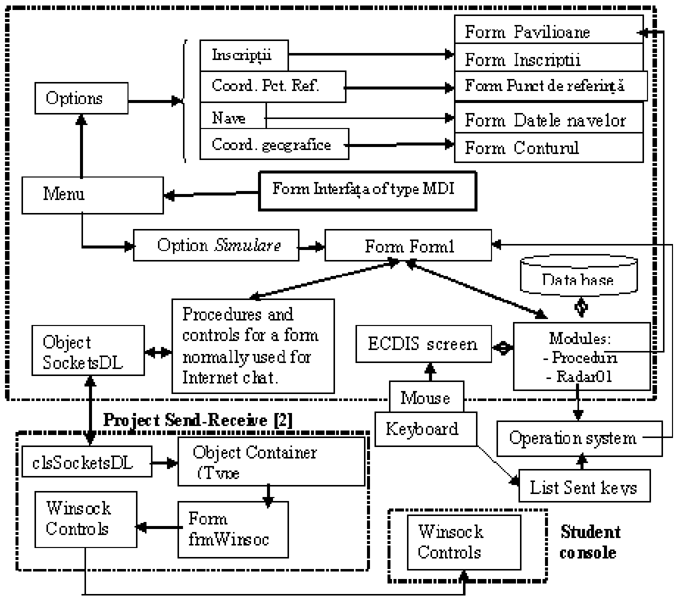

Figure 1: Project SINAV

1. The menu of the multimedia support

This product may be used in training sessions, each session based on a training project, i.e. a certain screen play which when it is developed on a computer network, may generate a framework for a practical exercise on board of ships.

The multimedia support is provided with a meniu that includes 2 submenus and 4 buttons, as it is shown in the table below.

LITORAL	NAVE	INSCRIPTII	Help_comenzi	Playback
Coord.geografice	Date nave			
Coord_metri	Coord.nave_metri			
Coord.pct.de ref.				
Val. unui minut				

The submenus can be used by the trainer before the beginning of the exercise. The option Coord. geografice from submenu LITORAL allow the trainer to update the geographical coordinates of the points that define the coast line.

The option **Coord_metri** runs a program that converts the geographical coordinates into rectangular coordinates, expressed in meters in respect with a point of reference located inside of the area where is pretended that the exercise takes place. This point is used to reduce the time necessary for updating the radar image on the screen, because the geographical coordinates take a lot more time to be updated.

When the point of reference remains too far from the area where the ships operate, it has to be removed closer to the ships. This action may be carried out by mean of option Coord. pct. de ref. If the point of reference has been changed, then the trainer must use the option Val. unui minut in order to update the value of a minute of meridian and that of a minute of parallel because these are the conversion factors used in conversion of distances between two points (given in geographical coordinates), in meters.

The option Date nave from submenu Nave, is used to update the starting position of each ship. These points may be converted in meters, for the same reason for which the points from the coast line have to be converted in meters.

When this button is pressed the user has to provide the ship number (the trainer has the ship no. 0), the Internet address or local addres, and the trainer has to mention how many ships (ship-computers) will atend the exercise. After he mentioned this number should not press the Enter key until all students have finished introducing their address, because each ship-computer after address is provided, remains in receive mode, waiting for the trainer's computer to ask for initial data, which actually are not necessary, but this is a test just to make sure that all computers may communicate with the trainer's computer.

Because there are many commands which may be introduced in computer (this is done from a text box which belongs to the form where the ECDIS image is displayed), the menu is provided with a button called Help, which can be pressed even during the exercise development, in order to remember the symbols used for commands to be typed in the command box. The last button on the menu is called Playback and it is used to review the evolution of each student during the exercise development. ECDIS comes from Electronic Chart Display Information System.

2. The development of the exercise

When the simulation starts the screen of the monitor suddenly becomes an ECDIS console. The image the trainer will see on this screen is presented in image on page 4. The most impressive part of the screen is the radar image combined with ECDIS image which is the core of the multimedia technologies used in this application, but for the sake of accuracy we will make a short presentation of the controls existing on this console. There are three modes of representing the radar image on this console: North-Up, Course-Up and Head-Up.

In North-Up operation mode the radar image is an absolute one, i.e. the land or the sea is fixed while the ship moves over it, but in Course-Up and Head-Up mode, if the ship is turning around, the lanscape or the other ships will rotate in oposite direction, which means that the course of the ship stays on the upper side of the circle of the gyrocompass and the lanscape, as well as the circle of the gyrocompass, rotates as one can see while flying in a helicopter. In this mode the image is built relative to the course of the ship.

Starting from the left-upper corner of the screen one can see a combobox where the user may choose the mode of radar operation. Regardless of the selected mode of representing the radar image, the radar antenna-signal (cursor) rotates continuously, without taking into account any user action such as typing commands, or the moment when the image on the screen is updated.

Next, the second object on the screen is a text box called **Nava la rdl**, that is the ship for which the ECDIS console builts the radar image. That is

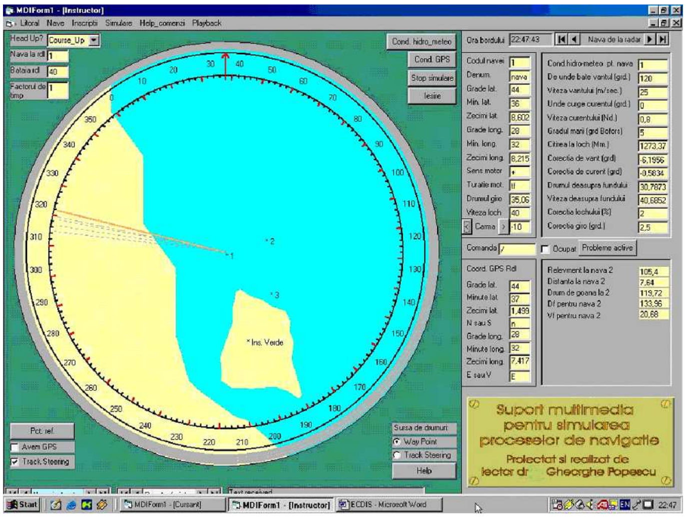

Figure 2: A typical image of the ECDIS console

the ship which is always in the middle of the screen and all the positions are related to this ship. That is why when the ship moves in straight line, the landscape or other fixed objects, if they are worth to be seen on the screen, will flow backwards. This option is valid only for the trainer because he is the only one who can afford to choose on which ship is on board at a given moment. The students have no choice. The image will always be built for their ship as it is normal, because it is supposed that the radar is on board of their ship.

The third object is **Bataia rdl** which means the range of the radar, but in fact this may not be changed for a real radar. What we chage here is the scale of representation on the screen. Of course when the scale is increased, more images will be represented inside the circle, but they will became smaller.

The fourth object is **Factorul de timp**. This is the factor of time compression which can be used to step faster on sequences which do not offer too much action for the students.

The next object in the left side, bellow the circle, is a button **Pct. ref.** which may be used to see the current coordinates of the reference point.

The sixth object is a check box called **Avem GPS**, which can be used only by the trainer. The students may read the message in this check box, but if this box is not checked it means that the GPS equipment is out of order and they have to compare their estimated point with points obtained by other methods.

The last object is a check box, called **Track Steering** which when checked, means that all changes of course and speed in the current voyage are stored for another voyage to be use by the automatic pilot.

In the right hand side, on top, there is a button called **Cond. Hidro_meteo** used to display the environmental conditions provided by the student, and the result of this display may be seen in image 1, in a table located in the right corner of the screen. This is an **On/Off** buton;

In the right side of this button is a text box called **Ora bordului** and one called **Nava de la radar**, which allows the program to display data in almost real time, from a table located in the database. These data refers to the ship which is in the center of the radar image and they can be seen in a table located below this data control. In the left side of this table, below the **Cond. Hidro_meteo** button there are three buttons:
- **Cond. GPS**, which is available only on the trainer's console and it may be

used to display the environmental condition provided by the trainer. These conditions are considerred real conditions and their efect is taken into account when the GPS point is computed.

- **Stop Simulare** stops the simulation;

- **Iesire** is the exit from the simulation realm.

Below the table displaying data about the ship located in the center of radar image is an embosed text box provided with label **Comanda** . This is the command text box and here is where the user may type all the commands regarding the ship moving parameters, the environmental conditions, and many others. Above this text box on the display is a field called Carma (the helm). The user may drive the rudder from this text box, but most convenient way is to press the buttons located in the left side of this text box, one for each sense. One press on such a button means 5 degrees on the rudder.

Below the text box for commands there is another table for display of the GPS coordinates of the ship. When the **Avem GPS** check box is deselected, this table reamains empty.

In the left side of the **Comanda** text box there is a button called **Probleme Active**. When this button is pressed a picture box is displayed containing a column of check boxes which in fact is a list of problems that may be solved continuously, in real time, until they are stopped. When a problem is checked on this picture box, another picture box shows up to take the input data for that problem and when the problem is ready to start, the picture boxes disapear and in the right hand side of the screen shows up a display for the results of active problems. The results of active problems (one may select more problems to be active simultaneously) are displayed continously until the user stops one or more of them. This way of solving problems make sense in navigation because for instance, if one wants to take a course to get on board of another ship, this course value is changing continuously depending on the relative postion of those two ships and on their speed, until one of these courses is taken by the ship that is to approach the other one.

3. The mechanism of the simulation

The simulated ECDIS console has a graphic interface capable to transform the display of a computer in a console like those included in a real bridge system used to control the ship operation. This means that it must allow the navigator to control the ship by supplying the course and the speed of the ship as well as many other user options and, at the same time it offers various displaing possibilities of all informations necessary to control the ship and to evaluate the situation in neighborhood of the ship. Among these displaing possibilities the most important place is reserved to the radar image combined with the ECDIS image, i.e. with the electronic map.

Each workstation has to compute, in real time (at about each 5 to 8 seconds), the real and the estimated point of the ship and when the master computer (located in the trainer's cabinet) asks for this position, the workstation must supply the real point to the master. The master computer collects the real point along with the ship movement parameters for each ship-computer (crew) and if necessary generates the position of other fictitious ships. Next, the master delivers the points of all ships to each ship (workstation), so that the monitors representing ARPA equipment will have enough data to compute and to build the radar image of the area surrounding the ship. The communication between workstations is accomplished by mean of a program used for chat on Internet [2].

ARPA not only computes the radar image, it also checks each neighbor ship if there is any danger of collision, (for this it has to solve the collision route for each neighbor ship). It also stores the route of the ship on magnetic supports, so that all the exercise can be played back for criticism and for grading.

Next the ECDIS will rebuilt the image computed by ARPA equipment, but on a background consisting of the electronic chart of the area. This background has to be placed of such a manner as the point of the ship to be located in the middle of the screen. It means that the background has to be rebuilt in real time. As one can easily realize, the displays of a workstation have to represent multimedia elements, not simple bitmaps but computed vector-based graphics, and in case of the ECDIS these images have to be placed on different layers.

This interface alows the user to mark some actions which he can decide to perform in certain situations. For instance the student my rise some signaling flags on the mast, or he can check if any other ship has flags risen on the mast, so that he can get their message. This feature is based on a set of files containing bitmaps of flags and a table from database that contains the name of each flag and the path to its file. It is quite clear that this application requires databases, multimedia, real time, computer network and many others.

The program that does the jobs specified above has been conceived under the pressure of three tough constraints.

The first one comes from the fact that the user or any other observer should not asist at the construction of the new image on the screen of the ECDIS console. They should only suddenly see a new image. This is a tough constraint because the reconstruction of this image may take a few seconds and therefore one must use a special technology to rebuilt this image underground and then to show it at once. The solution for this constraint is to built the image underground, in a hidden picture box, and to show it instantly when the image is ready to be seen.

The second constraint has to do with the fact that images moving on the screen, should not be interrupted to allow the user to type commands as usually happens when a dialog box is open on top of a form. Also, the request of getting data from the ship-computers has to be satisfyed without desturbing the ECDIS screen images.

The third constraint comes from the fact that the whole cycle of the program, including a complete automatic data exchange among these ECDIS consoles, should not be longer than the duration of a complete rotation of the radar antenna. Ussually this duration is somewhere between 7 to 10 seconds. For this reason all along of development of these programs the time was a permanent constraint.

Another problem to be solved in order to obtain a simulator from a bunch of ECDIS consoles is to simulate the GPS device. If somebody may ask why GPS, the answer is that the navigator may use estimated navigation, but when his ship has covered long distances, he must use and other algorithms (like those provided in coastal navigation, radio navigation, astronomic navigation) because if he has errors in estimating the effect of environmental conditions, then the effect of errors cumulates and after a longer period of time the point of the ship calculated with these errors may be far away from the real one. The simplest and safest way to get a correct point is to use GPS. That is why we have to simulate the presence of a GPS device on board.
If the GPS is so good why do we use other means for calculation of the point of the ship ?

The answer is "from the same reasons for which we don't abandon the magnetic compass for the sake of gyrocompass, i. e. if for some reasons the gyrocompass or GPS comes out of order and we don't have or we don't know to do something to replace the nonoperational equipment, then on the sea, we will get in a big trouble"

The solution to this problem comes from the fact that in an ECDIS console instead of the ship engine, we use a computer program which is infinitely repeated. This program calculates two points regarding the position of the ship: one based on the environmental data (currents, wind, vawes, tide) supplied by the trainer and one based on the enviromental data supplied by the commander of the ship. Both points take into account, beside the environmental data and the course and the speed of the ship supplied by the commander of the ship.

Because the environmental data supplied by the trainer are out of the control of the student, (he doesn't know the values of the corrections to be applied for the value of the environmental data because they are not displayed), these data behave like real weather conditions or factors wich carry the ship where they can. It is the navigator duty to determine the correct value of the corrections to be applied in order to get a corect estimated point of the ship. To be able to do that he knows the environmental data supplied by the trainer. What he doesn't know is the corrections that emerge from these data. That is why the point of the ship calculated with environmental data supplied by the trainer may be considered as GPS point. The other point may be considered the estimated point of the ship. When the estimated point of the ship is close enough to the GPS point, it means that the ship commander or the navigator on duty has obtained a correct estimated point.

In order to give to the trainer the possibility to watch the operation mode of each student, he has the possibility to built on his ECDIS console images according to data specific to any student ECDIS console, so that he can realize what course and speed has the student's ship, what environmental data uses the student and how the situation in the operation area looks like from the student's cabine. At the end of the exercise, the trainer may playback the development of the exercise for grading and criticism.

Conclusions

Modeling is an abstract representation of the reality and therefore only the aspects of interest have to be taken into account. The development of new informational and communication technologies on one side and the development of multimedia on the other side make the simulation of the integrated navigational systems an easier task. It is easier because the panels and the main board of these systems come closer to the appearance of the computers, and the computers have more efficient multimedia technologies to simulate any phenomenon or appearance.

As a consequence, many equipments used in the field to be modeled should be presented in simulator by their data suply and not by their panel. Based on these considerations simulators of various destination have been developed in recent years, some of them simply taking the form of software applications, without any hardware accesories like in the past. Such accesories are simulated by the software application, while the operation of the simulated equipment is simulated by algorithms developed on conceptual modelling.

SINAV is an example of how this trend operates. It is quite clear that such an application requires databases, multimedia, real time processing, computer network, activeX, sometime threads and multithreading, concurrent programming and many others but it comes out with a beautifull smart solution.

The methodology described in this report may be applied and in many other simulations or in e-learning products.

References

[1] http://home.clic2learn.com: Multimedia ToolBook, ToolBook II Assistant si ToolBook II Instructor.

[2] Cristopher J. Backmann, Lars Klander, Lingyan Tang, "Visual Basic Biblioteca programatorului ", Editura Teora, Bucureşti, 2001.

[1] "Ovidius" University of Constanta
Department of Mathematics and Informatics,
900527 Constanta, Bd. Mamaia 124, Romania
e-mail: epopescu@univ-ovidius.ro

[2] Faculty of Accounting and Finance
"Spiru Haret " University, Constanţa, Romania