Metaphor-Aware 3D Navigation

C. Russo Dos Santos, P. Gros, P. Abel,
D. Loisel, N. Trichaud
Eurécom Institute, B.P. 193
Sophia Antipolis, France
{russo,gros,abel,loisel,trichaud}@eurecom.fr

J. P. Paris
CNET France Télécom
Sophia Antipolis
France
jeanpierre.paris@cnet.francetelecom.fr

Abstract

Anyone who has ever experienced three-dimensional (3D) interfaces will agree that navigating in a 3D world is not a trivial task. The user interface of traditional 3D browsers provides simple navigation tools that allow the user to modify the camera parameters such as orientation, position and focal. Using these tools, it is frequent that, after some movements, the user is lost in the virtual 3D space and usually tries to restart from the beginning.

This paper presents how the 3D navigation problem is addressed in the context of the CyberNet project [2]. Our underlying principle is to help the user navigate by adapting the navigation tool to the virtual world. We feel that the navigation schemes provided by the 3D browsers are too generic for some specific 3D tools and we have developed adaptive navigation features that are dependent on the 3D metaphor used for visualizing the information and on the user's task.

1. Introduction

The primary aim of the CyberNet project is to develop a tool for network monitoring and management that takes advantage of 3D visualization and 3D virtual worlds. The network data is depicted metaphorically using the visual parameters of the virtual world. According to the type of task that the user wants to accomplish, different metaphors are used to visualize the information. The choice depends on the type of information that is relevant for that task, on the best mapping between the data values, and on the metaphor's visual parameters and user preferences.

Nonetheless, the CyberNet project is a research-oriented project and along with the development of a network management and monitoring tool more general concerns are aimed. Namely, our research interests are focused on the problem of visualizing in real-time large amounts of dynamic data; on how to construct virtual worlds automatically from a set of initial data and how to change these worlds on the fly according to the data dynamics; on how to choose the metaphor that is more adequate to represent a given network service and how to best map data values onto the 3D visual parameters available for each metaphor; and how to easily navigate in the 3D worlds. In this paper we only address the navigation problem.

When interacting with a 3D virtual world, one of the first requirements is being able to navigate in the world in order to easily access and explore information to allow for judicious decision making for solving eventual problems. Basic navigation requires being able to modify the viewpoint parameters (position, orientation and focal). For the user's movements to be efficient, it is important for the user to have a spatial knowledge of the environment and a clear understanding of his location. In order to enhance the user's navigation, navigation tools have to take into account the user goals and provide tools that help the user accomplish specific tasks.

We believe that the built-in navigation schemes that are available in most current 3D browsers are too generic. Navigation can be improved by adapting the navigation schemes to the virtual world and to the user's tasks. This belief led us to the concept of *metaphor-aware navigation*, that is, the navigation is tightly bound to the visual metaphor used and the way the user moves in the virtual world is determined by the metaphor that the same world is based upon. We also believe that the way a user navigates in a 3D world is intimately related to the task that he pretends to accomplish. This paper exploits the concept of metaphor-awareness related to 3D navigation and presents the CyberNet metaphor wizard. We focus on the 3D navigation using standard user interface tools (i.e., a mouse and a keyboard). In other words, we only address desktop navigation; we do not address specific problems and solutions related to immersive navigation.

The paper is organized as follows: in Section 2 we refer previous work already done in the field of 3D navigation. The next section, Section 3, introduces the concept of task-

Proceedings of the IEEE Symposium on Information Visualization 2000 (InfoVis'00)
0-7695-0804-9/00 $10.00 @ 2000 IEEE

dependent navigation. Section 4 presents some metaphor-aware navigation principles and Section 5 the corresponding metaphor-aware navigation control mechanisms. In Section 6 we discuss the implementation of the navigation wizard and in Section 7 we present an example of metaphor-aware navigation that is a part of the CyberNet project. Finally, in Section 8, some conclusions are taken and perspectives for further work are outlined.

2. Previous work

Several research works have already addressed the subject of 3D navigation, focusing different navigation issues and using different approaches. Some of this research work deals mainly with viewpoint manipulation. There is also research addressing the specific subject of spatial knowledge. Other research focuses primarily on constrained navigation. This sections presents some previous work regarding these three questions.

[6] reports most of the work already done on viewpoint manipulation. Navigation tools can be classified as being egocentric (moving a viewpoint through the world) or exocentric (moving the world in front of a viewpoint). They are also classified in terms of general movement (exploratory), targeted movement, specified coordinate movement and specified trajectory movement.

Most of the navigation tools implemented by VRML (Virtual Reality Modeling Language) browsers fall in the egocentric category [9] and the movements allowed have names such as fly, pan, walk or examine. General movements require to fix all the parameters but one, and to let the user modify the value of that specific parameter using the mouse or the keyboard. Some targeted movements (such as "fly to" with direct selection of the target or "jump to" with selection in a list of view points) are already supported. Although they may exploit the 3D world to simulate gravity or collision these navigation mechanisms are completely independent of the virtual environment itself.

On the subject of providing spatial knowledge to the user, [14] has classified spatial knowledge in three classes: landmark knowledge (being able to identify positions using visual cues), route knowledge (having a knowledge of spatial relationship between visual cues) and survey knowledge (having a global spatial understanding of the environment).

For position awareness, different kinds of solutions have been investigated. The main idea is to provide visual feedback of the user position. The simplest feedback scheme is to permanently display the 3D coordinate position of the user. This solution is not of great help especially because this position only has a meaning if the user already has an in-depth knowledge of the world geography. More elaborated solutions are based on the display of a global, simplified view of the world added in the user's field of view. [13] proposes the concepts of "World In Miniature". [12] studied how 2D maps could help users to navigate in virtual buildings. [4] presents the concept of "map view", a tool that allow the user to monitor its position (viewed from some "satellite" position) on a small virtual screen embedded in the 3D world.

Although there are differences among these methods, the basic idea is to include - in front of the user - a small overall view of the world and a marker showing the position of the user in that world. For orientation awareness, [10] has pointed out the importance of the knowledge of the vertical direction and presented some "upward" cues such as ground planes, backdrops and directional illumination.

[4] has proposed the concept of "trailblazing". The basic idea is to allow the user to leave graphical markers (that act as user defined landmarks) in the 3D world. The use of landmarks can be compared to the use of hyperlinks in HTML (HyperText Markup Language) documents, trailblazers being some kind of equivalent to the user's bookmarks proposed by HTML browsers.

Allowing a user to navigate freely in the environment is important, but most of us have experienced that being "as free as a bird" is not that easy. Research has been done in order to enhance the navigation activity by taking into account the goals of the user. The solution is generally referenced by the term "constrained navigation". Although it is true that these methods generally put constraints on the user movements, we prefer to use the term "helped navigation".

[6] reports some early work toward that direction. [8] presents the "tracking viewpoint"; the idea is to modify the user's direction of view in order to allow the user to track a specific object (potentially moving) in the scene. In other words, the system provides an automatic cameraman that follows an object in the scene. This idea was also presented in [4] for tracking the user position in order to control the "map view". [7] presents a tool that constrains camera movements so that the position is limited to a surface and the orientation is dependent upon the surrounding objects. This study has been done mainly in the context of terrain navigation. The authors conclude that this kind of help tools should be context-dependent, state-dependent and history sensitive.

3. Task-dependent navigation

For the user's movements to be efficient, it is important for the user to have a spatial knowledge of the environment and a clear understanding of his location. Much effort as been put in the "wayfinding" tasks [3]. The interest on this topic is mainly related to virtual reality immersed interfaces. Wayfinding is obviously not the only task a user may want to do when navigating in virtual spaces [5]. In complement to "be as free as a bird", which should be always possible,

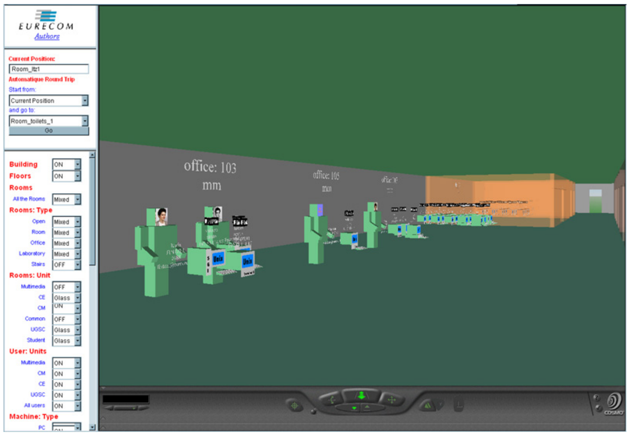

Figure 1. CyberNet Project: Inside the building metaphor. Network devices, machines, offices and staff information visualization tool.

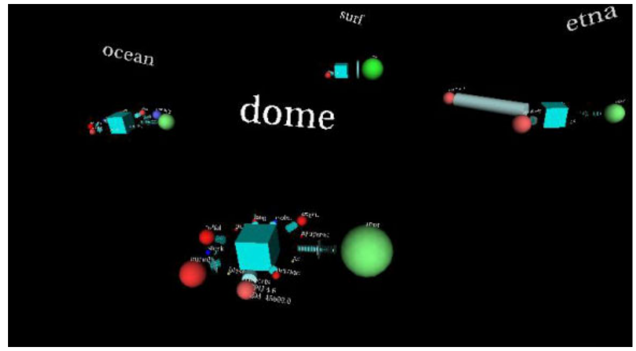

Figure 2. CyberNet Project: The solar system metaphor. Machine data visualization tool.

following are several user tasks related to our field of interest (we do not pretend to be exhaustive):

- Inspect an object either by jumping to a predefined viewpoint attached to that object, or by looking in the direction of that object from the user's current position. In the latter, a tracking mechanism should handle moving objects.
- Travel to an object following a logical path according to the metaphor in order to be aware of the objects relative positions or to monitor other objects along that path.
- Scan, traverse or visit several objects according to some criteria:
  - Scan all the objects that are children of a given object.
  - Traverse all the objects that have a common ancestor in the underlying hierarchy (depth first).
  - Visit all the objects from a user defined list or that the user has already inspected. This scanning requires jumping, traveling or looking at the successive objects.
- Have a global view of a set of objects. What is viewed from a parent is dependent upon the metaphor itself; the system may give an overview of all the children.
- Navigate in the 3D world relatively to the underlying hierarchy. The user can take advantage of his knowledge of the underlying hierarchical structure of the metaphoric virtual world.

4. Metaphor-aware navigation principles

Our work in the context of the CyberNet project pertains to the domain of information visualization. We use 3D information visualization to depict and exploit network data. Nonetheless, network management and monitoring is just an application domain to validate our research. Our research interests are mainly related to the three-dimensional information visualization of large quantities of dynamic data.

We chose to use 3D information visualization because of the added value of the third dimension. Not only it allows for larger volumes of information to be displayed but it also allows for better perception of the information and for new paradigms of visualization and interaction with the data.

4.1. Metaphoric worlds

The CyberNet project uses 3D metaphoric worlds to visualize the network data. Two factors are subjacent to the use of metaphors: most of the network information is abstract (e.g., what is the actual representation of a process?) and metaphors provide a means for making the information understanding easier since they utilize concepts that the user is already familiarized with.

CyberNet virtual worlds use mostly real-world based metaphors since their underlying structure is familiar to the user. We have designed real-world metaphors such as a building (see Figure 1), a city (see Figure 6 and Figure 7) or a solar system (see Figure 2). Nevertheless, it is sometimes useful to use more abstract metaphors, such as a cone tree [11], specially for depicting hierarchical information. The CyberNet system also uses abstract metaphors as the already cited cone tree (see Figure 3) or an information landscape (see Figure 4). The choice of the metaphor to be used depends on several factors, such as, the information to be

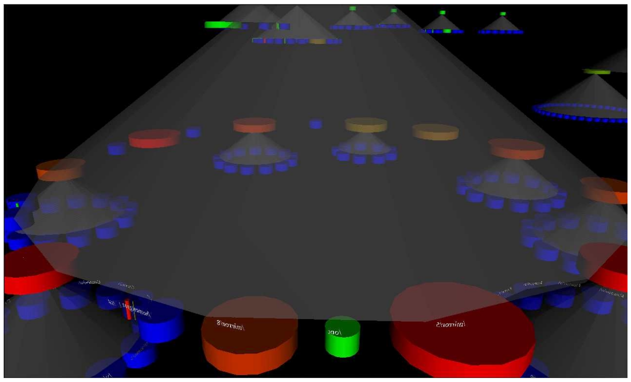

Figure 3. CyberNet Project: The cone tree metaphor. NFS data visualization tool.

displayed, the user's task and and the user's preferences.

Metaphoric worlds are built from predefined graphical components arranged hierarchically (e.g. cities are made of districts, districts are made of streets and buildings, buildings are made of floors that are made of corridors and offices, etc.). In the CyberNet project we use two different kinds of graphical objects: *3D glyphs* and *layout managers*.

3D glyphs are the atomic building blocks of the virtual worlds. They are used to construct the world and their visual parameters are used to visually map the information. Layout managers are fundamentally used to position their children, either 3D glyphs or other layout managers, in the 3D space (e.g., in a row, in a grid, in an orbit). The layout managers can also play the role of information mappers by using the position as a mapping parameter.

4.2. Metaphor-aware navigation

When the user navigates from office to office in a virtual building he does not use the same navigation mechanisms as when he is exploring a landscape of data or studying the topological structure of a cone tree. Our belief is that navigation mechanisms should be dependent on the metaphor and embedded in the graphical components of the 3D world. The goal is that the user navigates in the world with the mechanism most suited to that particular metaphor. We call this principle *"metaphor-aware navigation"*.

One of the easiest ways to understand the idea behind metaphor aware navigation is to consider path-based navigation in a building. When a user wants to go from his current position to another office, he generally follows some logical path through the stairs and corridors (see Figure 5). Following this path (as opposed to instantaneous jumping) is important since it gives to the user the knowledge of the relative location of the objects in the virtual space. Using

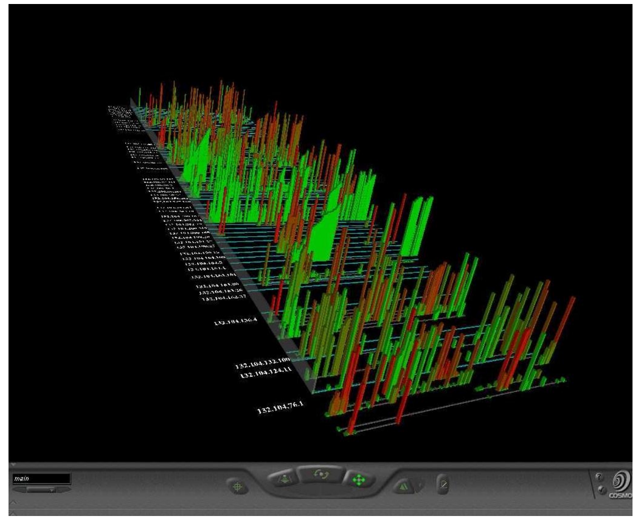

Figure 4. CyberNet Project: The information landscape metaphor. Network traffic data visualization tool.

traditional navigation tools, this kind of navigation is not an easy task. Metaphor-aware navigation will help him accomplish this task: the system will automatically route him along the right path in the building and will take care of the details of the navigation (e.g., turn, go down, and so on).

The basic principles we took into account when designing the CyberNet navigation tools were the following:

- Navigation tools should be adapted to the metaphoric world and may sometimes mimic real world navigation. This helps preventing user disorientation. We gave an example based on the building metaphor in the previous paragraph.

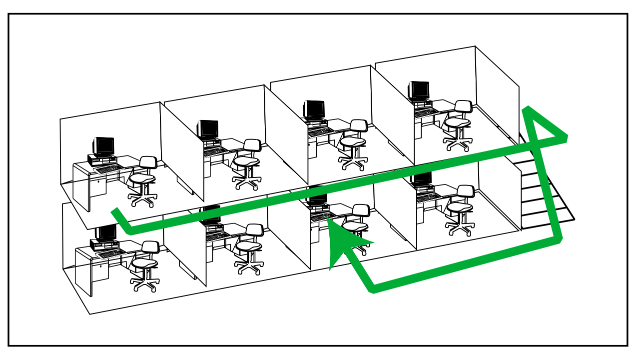

Figure 5. The metaphor-aware navigation path from one office to another.

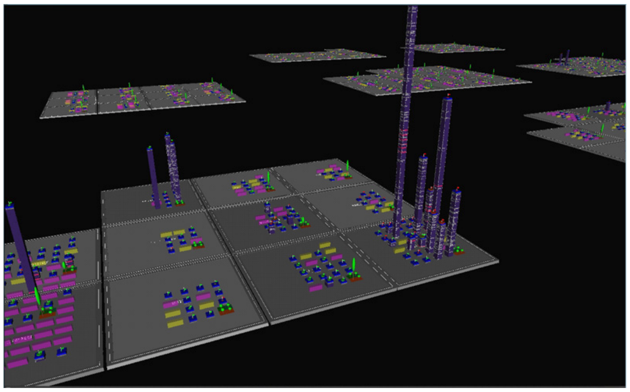

Figure 6. CyberNet Project: Flying at the district level in a city metaphor. NFS data visualization tool.

Figure 7. CyberNet Project: Walking at the street level in a city metaphor. NFS data visualization tool.

- Different navigation schemes may be used for navigating at different levels of the metaphoric world hierarchy. This kind of navigation is helpful to acquire survey knowledge. For instance, in a town, the user may navigate at the district level, like a bird flying over the town (see Figure 6), or he may navigate at the street level (see Figure 7), walking in the streets in order to go from one building to another. The user will have a global view (seen from far away) when he is navigating at the district level and a narrowed, more local view when navigating at the street level.

- Navigation tools should take advantage of the metaphor hierarchical world and provide schemes to navigate through the various levels of the hierarchy. The user will then benefit from his knowledge of the underlying structure of the world, which matches the network management service hierarchy. He may just use a navigation interface proposing standard movements in the hierarchy (i.e., up, down, next, and previous).

- Navigation tools should help the user when he has a specific navigation task and should handle the navigation details for him.

- The navigation tools should take advantage from the fact that, although sometimes based upon the real world, the virtual worlds pose fewer constraints to the user (e.g., gravity, size, and speed) and that the user may even change the virtual world.

4.3. Metaphoric components

The aforementioned principles led to the design of what we call *metaphoric components*. Metaphoric components are the graphical elements of a metaphor, i. e., 3D glyphs and layout managers, with embedded interaction and navigation capabilities. Thus, according to our terminology, metaphoric components and graphical elements of a same metaphor do not exactly mean the same thing, from a theoretical point of view.

Graphical elements are just the basic elements that are used to construct a metaphoric virtual world and serve to visually map the information. Metaphoric components are all the graphical components of the metaphor, but we consider that this type of components must have embedded interaction and navigation features, if the case applies. As an example, a rectangular box may be a graphical element (3D glyph) for constructing a given metaphoric world (e.g., a virtual building). But that rectangular box when instantiated in the same metaphor to play the role of a floor with navigation features (e.g., moving the user along the floor's corridor) is then perceived by our system as a metaphoric component. It should be noted that metaphoric components also provide additional functionality, not related to the navigation (e.g., built-in interaction mechanisms), thus beyond the scope of this paper. For further information on this (and other) topic please refer to the CyberNet's project webpage [1].

5. Navigation control

In order to assist the user in its navigation task, the system should know the *user's current state*. We define the user's current state as constituted by three parameters: the *user's current location* in the 3D world, his *current node of interest* (the node that currently has his attention) and a *new node of interest*.

When the user is simply moving around in the world, the user's current node of interest and his current location are identical. But some navigation tasks require the user to be located in a place and to have his node of interest on another object; an example is the "look at" navigation mode. The basic functionality of the navigation system is to allow the user to modify his current state (current location and node of interest), by choosing a new node of interest.

Our helped navigation mechanism is target oriented: the user should specify a *node of interest*. This node of interest is a node in the metaphor hierarchy that can either be the destination of a movement or an object to look at. In CyberNet, the node of interest can be defined using two basic mechanisms. *Absolute selection* allows the user to identify an object by its name (using a hierarchical VRML-like viewpoint menu) or by direct selection in the 3D space. *Relative selection* allows the user to identify an object by its position in the hierarchy relatively to the previous node of interest. Once a target is defined, the system automatically handles the details of the navigation for the user. For instance, the user is transported to the target location by following a logical path according to the metaphor.

5.1. Absolute/relative navigation

Absolute navigation requires the user to select a new node of interest using absolute selection: a selection mechanism similar to the one supported by most VRML browsers for viewpoint selection. For this purpose, when a metaphoric component is created it should notify its existence to the system. The system is responsible for offering a scheme to select that new object as the new node of interest. This selection can be done, either by using a 3D embedded interface (by clicking on an object for example), or using an external menu, that is hierarchical and context sensitive. This general mechanism looks like the mechanism supported by VRML browsers for handling viewpoint selection. Once the user has selected a new node of interest, the system has to determine what steps to take according to the current navigation task.

Relative navigation requires the user to select a new node of interest using relative selection: that is the new node of interest is chosen relatively to the underlying metaphor hierarchy using traditional browsing operators such as up/down (in the hierarchy), or next/previous (element at that level). For example, when the user is in an office, choosing "next" will automatically take the user to the next office in the corridor, while "up" should take the user to the corridor or to the next floor (according to what the designer has specified as being the higher level).

This relative navigation scheme is important since CyberNet users are generally aware of the hierarchical service model represented by the metaphoric world. Since the virtual world structure follows the hierarchical service model they may rapidly access points of interest. The translation of the user action (i.e., "up") into a precise node of interest is metaphor dependent. This is implemented using a neighboring table provided by each metaphoric component. The navigation system uses this table to determine the new node of interest and then the absolute navigation algorithm is used.

5.2. Automating the selection

It is important to be able to automate the selection of the new node of interest. The first use of automation is to be able to go back and forth in the historical list of already visited nodes. Another important automation application is for defining round trips in the world in order to monitor a set of nodes. Simple navigation tasks such as glancing at all the nodes that are children of a given node also require automation. Automation involves getting to the next target (when several objects are involved) or back to the original location (when the end of the movement is reached). The automation requires the system to manage the time spent at each intermediate node of interest as well as the possibility to interrupt the navigation tasks (e.g., suspend, resume, stop).

5.3. Assisted movements

So far, we have only discussed the schemes provided to select the new node of interest. The system will now have to determine a set of movements that should be done within the context of the current metaphor in order to go from the user's current node of interest to that new node of interest. These movements are dependent on combined use of a *user mode* and a *movement mode*. The movement possibilities obtained by logically combining these two modes are explained in Table 1.

Once the system knows the new node of interest, it supports two user navigation modes: the first mode is called *go to*. In this mode the user's location is modified in order to be transported to the location of the new node of interest. The second mode is called *look at*. In this mode, the user stays at his current location and his direction of view is modified in order to look at the new node of interest. The precise movements are dependent on the movement mode described in the next paragraphs.

The system supports three possibilities for movements modes: point to point, interpolated, and path. Using the *point to point mode*, the user can directly go to (or look at) the node of interest. This is very simple to implement

User mode	Movement mode Jump	 Interpolated	 Path
Go to	The user jumps to the new node of interest and gets attached to it. (VRML)	The user flies in a straight line from his current location to the new node of interest and his orientation is modified. (VRML)	The user travels from his current location to the new node of interest according to a metaphor-based path.
Look at	The user stays at his current position and looks in the direction of the new node of interest. If the object is moving then it is tracked.	The user stays at his current position, his direction of view is animated from its current value to the direction of the new node of interest.	The user stays at his current location, but his direction of view follows a metaphor-based path from the current node of interest to the new node of interest.

Table 1. Movements obtained by combining the different navigation modes (movements marked VRML are supported by most VRML browsers).

since it only requires modifying the viewpoint. Although we all dream of "tele-transportation" in our car every morning, this navigation scheme has some drawbacks: the user tends to lose his spatial knowledge of the world, since he does not have information on the relative position of the objects anymore, as he cannot observe other objects along the path.

In their "fly to" mode, some VRML browsers support interpolation from the starting viewpoint to the destination viewpoint. We call this mode the *interpolated mode*. The result is somewhat unpredictable as soon as the viewpoints have different directions of view: the user gets the impression of doing some strange looping. This is the reason why we implemented the path mode.

The main idea behind the *path mode* is that the user should follow a logical path in order to move from one node of the hierarchy to another; this path relies on the metaphor itself and cannot be independently determined. For example, when a user wants to go from one office to another, he will automatically be routed through the corridor. If the office is located on a different floor, the corridor will have to route the user to the elevator (or stairs), the elevator will route the user to the desired floor and then to the desired corridor, and the latter will take the user to the destination office.

6. Navigation implementation

In this section we are going to present the CyberNet's navigation wizard and describe the implementation of our distributed navigation system.

6.1. The navigation wizard

A navigation wizard manages the CyberNet navigation system. Upon creation each metaphoric component notifies the wizard that it is a potential node of interest and provides, at least, a predefined viewpoint location and a point of interest. The navigation mechanism requires the user to define a node of interest and a navigation mode. The role of the navigation wizard is:

- To manage the navigation user interface. This interface should allow the user to have access to both relative and absolute navigation. Relative navigation only requires the use of standard navigation buttons (that is, up, down, next, and previous). In order to support absolute navigation, the wizard has to make each metaphoric component viewpoint available to the user by inserting an item in a hierarchical menu and/or provide direct selection.
- To always track the user's current position and current node of interest in relation to the metaphor hierarchy.

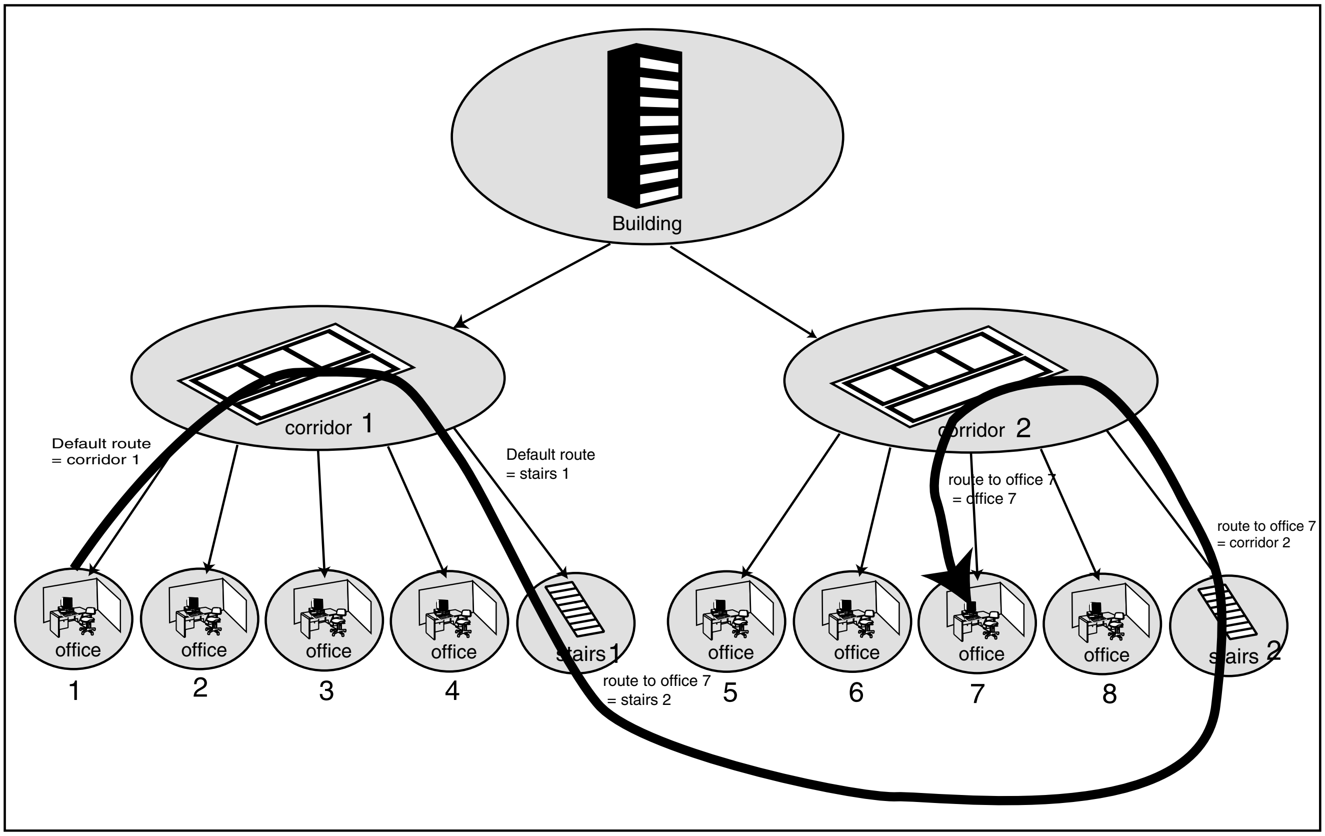

Figure 8. Path navigation mode: The metaphor hierarchy tree is traversed using routing tables.

- To determine a set of movements that should be done within the context of the current metaphor and according to the current navigation mode.

6.2. Embedded navigation

In order to avoid complex centralized algorithms, the management of the path navigation algorithm is distributed between all the metaphoric components. Each metaphor component implements a routing table, very much like the one used by IP (Internet Protocol) routers in order to route packets on the Internet. The navigation wizard basically follows the routing and activates the traversed nodes. The effective movements are handled by each activated node within the space it has under its responsibility (usually related to its bounding box). For this purpose, each metaphoric component has three tasks:

- To provide a *neighboring table* used to support relative navigation. The goal of this table is to be able to translate user action regarding relative selection (e.g., up, down, next, previous) into a destination node according to the metaphor.
- To implement a built-in *routing table*. The goal of this table is to define which is the next node of the hierarchy that should be traversed to go from the current position to the desired destination. In the previous corridor example, going to an office on the next floor requires to go to some intermediate node like the elevator.
- To control the navigation in the part of the space it has under its responsibility. In the previous corridor example, the corridor should effectively translate the user onto a path from one point to another. Of course care should be taken during the design to insure that the path is continuous when moving from one node to the next.

The general path navigation mechanism uses these built-in mechanisms as follows: the routing tables are used as a mechanism to find the metaphoric route that links two nodes (see Figure 8). The navigation wizard basically follows that route and activates the traversed nodes along the route. Each activated node handles the effective movements. The current node first has to use its routing table in order to determine the next traversed node. When this node is identified, then the current node should animate the user's viewpoint from its current position to some position provided by the next node.

The path followed is dependent on the metaphor but is generally simple to implement because our basic graphical components have usually simple geometric characteristics. The complexity of the resulting world is an effect of the combination of a high number of simple components.

It is important to note that it is not mandatory that every node of the graphical hierarchy participates in the navigation mechanism. One reason is that it may not make sense, in the context of a given metaphor, to provide some glyphs with navigation capabilities (e.g., usually a window does not handle navigation).

There is also another type of metaphoric components that do not possess navigation features: some intermediate graphical nodes are inserted in the hierarchy only for visual purposes and they are transparent from the user perspective since they have no real world meaning. For example, the "fit" layout manager is inserted to force the child hierarchy to fit into a specific space allocated by the parent layout manager. It is generally implemented by scaling down its children elements. In that case, the graphical node becomes "transparent" in terms of navigation and the user does not even notice its existence.

In order to be transparent, a node does not notify its existence to the navigation wizard and, thus, is not referenced in any of the neighboring and/or routing tables of other nodes. It also happens that a node acts as transparent in a determined navigation node, and must handle navigation when in a different navigation mode.

The ultimate task of the navigation wizard is to manage navigation automation. The navigation wizard also maintains a history buffer in order to go back and forth to previously accessed locations. The navigation wizard is able to manage loop sets of nodes of interest and to loop through them in order to implement round trips. It also supports automatic scanning (one level in the hierarchy) and traversing (depth first traversal of the hierarchy) of the children of a given node. For all these tasks, the navigation wizard exploits the information stored in the metaphor hierarchy itself, and then translates it to actual movements that occur in the visual world.

It should be pointed out that the design of the metaphor-aware navigation is really part of the metaphor design. It has the same level of importance that the purely graphical/modeling work. An important point is to define a coherent navigation scheme for the metaphoric world as a whole. When designing a metaphoric navigation, the mechanisms to navigate in the world should be developed with the same care that is put on the pure 3D modeling (which involves the graphical design and the mapping between real world values and visual parameters of the graphical components).

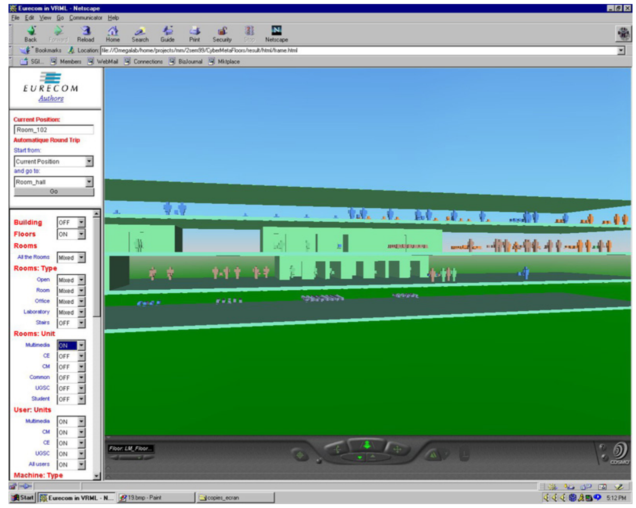

Figure 9. CyberNet Project: 3D visualization tool of the Eurécom Institute building with visual information on the staff, machines and network infrastructure.

7. Metaphoric navigation example

As part of the CyberNet project we have developed a demo tool to visualize and inspect the physical location of network devices, according to their actual location in the Eurécom Institute building (Figure 9 and Figure 10). This demo also contains information regarding the personnel and the physical structure of the building, with the correct relative location of offices, labs, and so on. You may access the online demos at CyberNet's webpage [1].

Although, at a first glance, the visualizations may seem like an faithful three-dimensional reproduction of the actual building, the interaction possibilities with the virtual representation go way over than those of the real life building. The user may render transparent any wall, or he may decide to hide/unhide information (for instance, rooms or entire departments) and he may even change the whole configuration of the building by displaying a department per floor (in real life and in the default visualization, departments are spread across all floors).

Since the interaction capabilities do not constitute the scope of this paper we will not develop the subject further. Nonetheless, we must bear in mind that these kinds of changes also have a big impact on the navigation (e.g., when determining the navigation path that is to be followed). For instance, if the user chooses to visualize one department per floor, all the neighboring and routing tables used to determine the intermediate and destination nodes have to be updated, since almost all the elements's locations are bound to

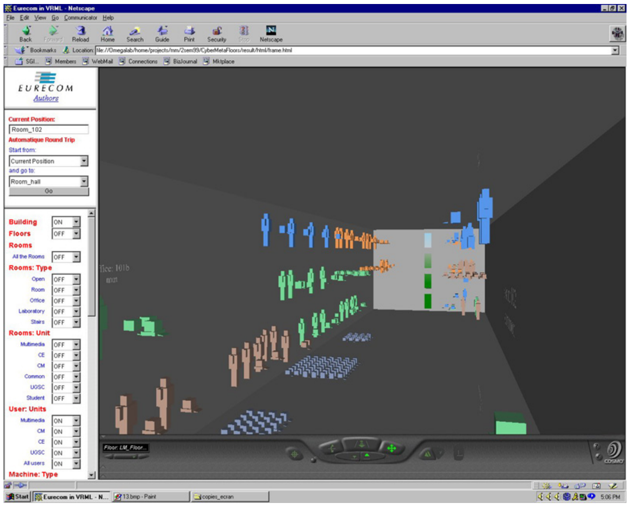

Figure 10. CyberNet Project: The Eurécom building with transparent walls and floors, a suitable environment for testing the look-at navigation mode.

be affected.

The main reason for developing the virtual building demo was to test a first version of the navigation wizard and some of our ideas regarding the navigation in 3D virtual worlds. Mainly, we wanted to test the embedded aspect of the navigation, where the graphical components are responsible for handling the navigation in their "territory", and the selection of points of interest and the subsequent determination of the metaphoric path.

In the 2D (two-dimensional) interaction interface (that is visible on the left of Figure 9 and Figure 10) the user may choose a destination point. The navigation mechanism then takes him there following the shortest "feasible" path (i.e., without traversing walls, taking the stairs whenever it is necessary to change floors, and so on). The selection menu is context sensitive (gives the current position as default for the departing point) and hierarchical.

For instance, if the user wants to go from his current location to a new location in the upper floor, he just has to choose the new target in the hierarchical menu on the 2D interface (the default departing position is always the user's current location). Then the navigation wizard will handle for him all the cumbersome details like moving along the corridor, finding the stairs, climbing up the stairs, and moving along the upper floor's corridor till the destination point. The user will be automatically taken along this path, thus getting information on the relative location of the elements he passes by, which will enable him to build an accurate spatial model of the elements's locations.

8. Conclusions and further work

In this paper we have presented the fundaments of the CyberNet project Navigation Wizard. We have stated our views regarding enhancing and making it easier to navigate in a 3D virtual world. Our major contribution is the metaphor-aware navigation concept and the fact that we are able to implement this in an automatic manner.

The metaphor-aware navigation concept is basically stating the principle that the way we navigate in a given virtual world depends on the characteristics of said world, namely on the metaphor employed to depict the information. In other words, the solution to an easier 3D navigation does not lie in one generic way of navigating that is applicable and appropriate for every single case. Furthermore, in the same world, navigation may also take different modes, depending on the hierarchical level of the metaphor in which the user is moving; e.g., flying at district level and walking at street level in a metaphoric city (see Figure 6 and Figure 7).

The task-dependent navigation was also approached. If the user is doing some monitoring work, he probably wants to navigate in the world in a serendipitous mode, just making sure that he passes by all the devices that must be monitored, in a kind of an all round-trip. On the other hand, if he wants to do some specific management task, he probably wants to get to the final destination taking the fastest/shortest path to get there. We have different navigation modes that take into account these differences.

In order to implement the automatic navigation mechanisms that follow those requirements, we have described how we embed the navigation capabilities in the graphical elements. We have also described how we delegate the simple navigation tasks on the graphical elements and make them responsible for all the navigation that must occur within their "turf" (usually defined by the graphical object's bounding box). This avoids implementing a large centralized algorithm and speeds up the navigation. We have shown how, based on the concept of neighboring tables and routing tables, we are able to determine the metaphoric path that is the support of the path-based navigation. And we have stated the role played by the navigation wizard, specially regarding keeping updated information on the user's state.

As further work we intend to do some user testing in order to validate (or not) our work. In fact, a major lack in our work so far is the absence of real user testing. Up to the moment we have some promising in-house feedback but the users are people who are somewhat familiarized with the project.

We also intend to enhance the navigation wizard to take into account some of the ideas that were already stated but have not yet been implemented. For instance, choosing multiple points of interest in a single step or implementing all the navigation modes (so far, for example in the demo presented above, only the path navigation mode is implemented). But the major test will be to implement the same navigation wizard in a dynamic world, i.e., a world that is updated on the fly to reflect the changes that occur in the network data. This poses several problems; for instance, what if an object that was selected as a new point of interest is deleted?

References

[1] Cybernet project's webpage: http://www.eurecom.fr/~abel/cybernet.

[2] P. Abel, P. Gros, C. R. D. Santos, D. Loisel, and J.-P. Paris. Automatic construction of dynamic 3d metaphoric worlds: An application to network management. In J. R. Robert Erbacher, Philip Chen and C. Wittenbink, editors, *Visual Data Exploration and Analysis VII*, volume 3960, pages 312–323. SPIE - The International Society for Optical Engineering, Jan. 2000.

[3] R. P. Darken and J. L. Sibert. Navigating large virtual spaces. *International Journal of Human-Computer Interaction*, 8(1):49–71, 1996.

[4] J. D. M. Edwards and C. Hand. MaPS: Movement and planning support for navigation in an immersive VRML browser. In R. Carey and P. Strauss, editors, *VRML 97: Second Symposium on the Virtual Reality Modeling Language*, New York City, NY, Feb. 1997. ACM SIGGRAPH / ACM SIGCOMM, ACM Press. ISBN 0-89791-886-x.

[5] J. Gabbard and D. Hix. Taxonomy of usability characteristics in virtual environments. Technical report, Virginia Polytechnic Institute and State University, Nov. 1997. Final Report to the Office of Naval Research.

[6] C. Hand. Survey of 3D interaction techniques. *Computer Graphics Forum*, 16(5):269–281, Dec. 1997.

[7] A. J. Hanson and E. A. Wernert. Constrained 3D navigation with 2D controllers. In *IEEE Visualization '97*, Oct. 1997.

[8] T. Kaye. The tracking viewpoint: http://reality.sgi.com/tomk/demos/vrml2/trackingviewpoint.

[9] J. D. Mackinlay, S. K. Card, and G. G. Robertson. Rapid controlled movement through a virtual 3D workspace. *Computer Graphics*, 24(4):171–176, 1990.

[10] A. Murta. Vertical axis awareness in 3d environments. In *Framework for Immersive Virtual Environments '95*, pages 169–176, London, 1995.

[11] G. G. Robertson, J. D. Mackinlay, and S. K. Card. Cone trees: Animated 3D visualizations of hierarchical information. In S. P. Robertson, G. M. Olson, and J. S. Olson, editors, *Proc. ACM Conf. Human Factors in Computing Systems, CHI*, pages 189–194. ACM Press, 28 Apr.–2 May 1991.

[12] G. Satalich. Navigation and wayfinding in virtual reality: Finding proper tools and cues to enhance navigation awareness. Master's thesis, University of Washington, 1995.

[13] R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a WIM: Interactive worlds in miniature. In I. R. Katz, R. Mack, L. Marks, M. B. Rosson, and J. Nielsen, editors, *Proceedings of the Conference on Human Factors in Computing Systems (CHI'95)*, pages 265–272, New York, NY, USA, May 1995. ACM Press.

[14] P. W. Thorndyke and S. E. Goldin. Spatial learning and reasoning skill. In H. L. J. Pick and L. P. Acredolo, editors, *Spatial Orientation: Theory, Research, and Application*. Plenum Press, New York, 1983.

0-7695-0804-9/00 $10.00 @ 2000 IEEE