Experiments in Information Visualization Using 3D Metaphoric Worlds

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

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

Abstract

In this paper we present experiments concerning the use of three-dimensional (3D) visualization for representing information. These experiments were conducted in the context of the CyberNet research project. The CyberNet project is oriented toward the enhancement of the user interface of network management tools. Its aim is to study how metaphoric 3D visualization may help the user in the process of monitoring large amounts of dynamic information. Although the project focuses on a specific application domain, network management, the core framework is designed so that it can be applied to other domains, and the system may be accessed from a web browser, thus allowing for information and knowledge sharing.

1. Introduction

The objective of the CyberNet project is to study the usability and effectiveness of 3D techniques and virtual reality interfaces for system and network management. Some prior work has already been done in the field of using virtual worlds for network management and visualization, as in [5] and [3]. However this work focused mainly on visualizing static topological information and did not address the problem of visualizing high dynamic network service information. We intend to prove that the visualization of distributed dynamic data can be made more efficient with 3D technology.

For this purpose, we have designed a general framework that makes the development of specific visualization tools easy. This framework is designed to dynamically build and update 3D worlds according to the real world data. It also features some specific support for user interaction and navigation. We consider that the CyberNet framework is specific in its approach because of the following features:

- The processed data are dynamic. The visualizations are designed to cope with this dynamic nature so that the 3D worlds are permanently updated. New objects are added to the 3D scene, existing objects disappear, and the visual appearance of the displayed objects is continuously modified in order to reflect the state of the network data.

- The translation between the real world data and their visual counterparts is handled automatically by the system. We call this translation the mapping process. This feature allows for the automatic construction of the virtual worlds and for updating the visualizations on the fly.

- The visualization is based on 3D metaphors. The traditional way to represent network information in 3D is to extend the usual two-dimensional (2D) graphical business representations such as 3D bar charts, 3D graphs, and so on. These representations are well suited for displaying small amounts of information but they generally fall short when large amounts of information are involved and when it is important to show relationships between different data subsets.

  In order to represent information CyberNet uses metaphors based on real world structured systems such as towns, solar systems, and buildings, for instance. The idea behind the use of metaphors is to take advantage of the natural hierarchical structure found in most of these metaphors (cities are made of districts, each district is made of buildings and streets, each building has floors, rooms, windows, etc.). CyberNet uses this hierarchical structure in order to encode structured information related to a network service.

  The use of metaphors has also the advantage of producing worlds that are familiar to the user. Visual metaphors make appeal to analogies with real world objects that the user is already familiarized with, to introduce new concepts to the user [8]. This way, the end user can relate to a given metaphor, which facilitates data representation comprehension. Using carefully designed visualizations, the CyberNet system can benefit from the user previous knowledge and thus enhance the user understanding of the world, as well as ease the navigation and interaction.

The CyberNet project framework is described in detail in [1]. In this paper, we focus in the different visual representations that were developed to deal with specific network monitoring needs. We also present our approach regarding user navigation and interaction with the virtual worlds.

The paper is organized as follows: Section 2 describes the automatic construction of the 3D metaphoric worlds. Section 3 presents the navigation in the virtual worlds. In Section 4 we describe the interaction with the metaphoric worlds. Finally, Section 5 presents the conclusions and outlines some perspectives for further work.

2. Building 3D worlds automatically

In this section we describe how we construct the metaphoric worlds. We begin by presenting the design of a service model example and then we describe the design of a 3D metaphor. Afterwards, we describe the process of mapping the service onto the metaphor. We also present an example of using different metaphors to visualize the same service, according to the user's interests.

2.1. Designing a service model: the NFS example

In order to be efficient, the representation of information requires the data to be logically structured into a network service model. Each data set has its own structure. This structure is dependent on the application domain and requires expertise knowledge. Basically, structuring the data requires to group data according to common properties and to identify relationships. In our structuring process the final service model is represented by a tree. It is the tree data structure that will be directly translated into visual elements.

The data model we have developed is based on the concept of entities. In the context of network management, an entity can represent a physical device (e.g., hubs, routers, computers, etc.) or conceptual items (e.g., connections, processes, etc.). Each entity groups all the values necessary to describe the current state of the data element it represents. An entity has a type (e.g., the hub type) and a defined set of typed values (e.g., for a hub its IP address, its number of ports, etc.). The entities are created by an entity collecting process described in [1]. Entities are stored in a repository that acts as a Database Managing System (DBMS).

The data model tree is composed of service nodes. A service node is an object that knows how to interpret a small part of the data model. The way a service node works is the following: the service node searches, using SQL(Structured Query Language)-like queries, for the entities that have some specific properties and analyzes the retrieved entities in order to create the children service nodes.

The service model together with the CyberNet framework are designed to be able to handle data dynamics: the repository updates the service nodes each time an entity that matches the query is created. This paper will not address the problems and solutions for managing the dynamic nature of the data and it will not address the mechanisms involved in the data collecting process either. Further information about these topics is given in [1].

Client/Server applications (like mail, DBMS, etc) are typical services for which we believe 3D metaphoric worlds can help. An example of such a service is the NFS (Network File System) client/server application. This application allows for computer disks to be shared in a user transparent way. NFS servers export their file systems to other computers and NFS clients import file systems from servers. Using NFS a remote disk is viewed in the same way as a local disk.

Nonetheless, from a network administrator and system manager's point of view, each access to a remote disk generates network traffic. A typical NFS site configuration involves a high number of computers on a typical Unix/NT workstation network. We chose NFS as an example of the administration of a complex networked client/server service. For this purpose, we developed data collecting agents that are able to get the NFS information from all over the Local Area Network (LAN) at Eurecom Institute.

The NFS service model designed by us is hierarchical as it is shown in the left part of Figure 1. The root service node of the hierarchy is the LAN, the second level comprises the different workgroups of the institute (split according to subnets). Each subnet service node groups a set of computers, each computer service node features a set of local disks and a set of remote disks (accessed through NFS). The local disk may be shared using NFS. In that case, the computers that remotely access the local disk are added as children of the local disk service node. Each client computer service node also groups all the filehandles opened on that file system.

The NFS service model is composed of the previously described service nodes hierarchy. In addition to the service nodes, the hierarchy includes as leaf nodes the references to the entities retrieved from the repository. For instance, a computer service node has the following children: one reference to the actual entity that describes all the characteristics of the computer (as retrieved by the collecting agents) plus a storage service node. This storage service node has two children service nodes: one for managing local disks and one for managing remotely accessed disks. The former has as many children service nodes as actual local disks; the latter has leaf nodes that are direct references to the entities that represent each remotely mounted disk on that computer.

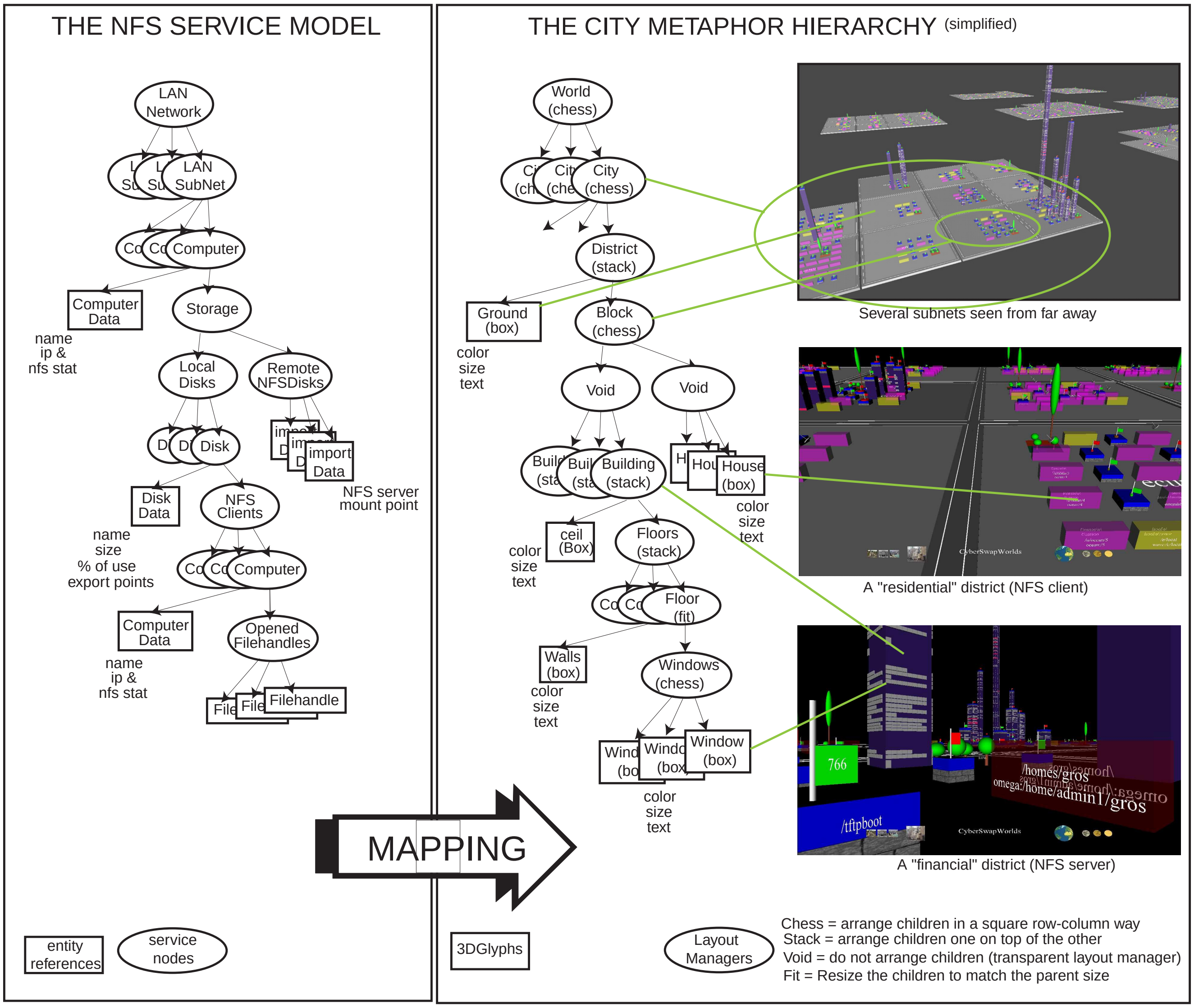


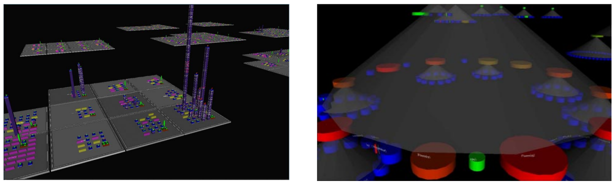
Figure 1. The NFS service model and its mapping onto the city metaphor.

One interesting characteristic of this example is that each local disk service node has descendents that reference the computer entities that are its NFS clients. Thus, the computer entity may be referenced several times in the same hierarchy. This characteristic will be exploited by the interaction mechanisms described in Section 4.

2.2. Designing 3D metaphors: the city example

Our visualization model is based upon the concept of 3D metaphors. There is previous work done in the field of information visualization using 3D metaphors, either real-based, such as a cityscape [7], a solar system [4] or a garden [4], or more abstract, such as a cone-tree [9]. Metaphors are used because their underlying structure is familiar to the user and therefore the user is able to relate to the relationships existing between different objects of the metaphoric world [8]. For example, in a city metaphor it is straightforward that houses in the same district share some common properties. We have designed real world based metaphors such as a building, a solar system or a city. It is important to note that a given metaphor is not designed for a specific service model. In fact, a service model can be visualized using any metaphor, provided the metaphor offers sufficient capabilities for viewing the service information.

A metaphor may be viewed as a world construction pattern. The metaphor is a hierarchical structure that will drive the world construction. Each node of this hierarchy is called



The NFS service represented through the City metaphor

The NFS service represented through the Cone-Tree metaphor

Figure 2. Representing the same service model using different metaphors.

a graphical component (GC). Intermediate nodes are called layout managers (LMs). The presentation of large volumes of information needs to be organized in space in order to make its interpretation easier [6]. Layout managers are a step towards organizing information spatially: the primary function of the layout managers is to arrange their children in space. They also have additional functions concerning navigation and interaction and may also generate some visual appearance such as semi-transparent bounding boxes, to enhance visual representation effectiveness, as suggested in [11]. Leaf nodes are 3D glyphs (3DGs).

3D glyphs are 3D graphical objects that represent data through visual parameters [2]. The level of complexity of a 3D glyph can be measured from the number of visual parameters it offers for modification and hence to the dimension of data that can be displayed. Some examples of complex 3D glyphs can be found in [2] and [10].

In the city example, visible on the right of Figure 1, the root of the metaphor hierarchy is the world LM. This layout manager arranges its children in 2D in a row/column way (like a chess plate). The children of the world LM are cities LMs. Cities are also LMs that arrange their children in a chess plate way. The children of cities layout managers are districts LMs, which stack a block LM on a ground 3DG. The block LM is a chess-like layout manager, whose children are buildings LMs and houses 3DGs. Buildings LMs are stacks of floors LMs, that are made of walls 3DGs and windows LMs. The window LM just arranges its children windows 3DGs.

This metaphor is used as a pattern for constructing a 3D world, examples of which can be seen on the far right part of Figure 1. This pattern is exploited by the mapping process described in the following section.

2.3. Mapping the service model onto the metaphor

Mapping is the process that automatically constructs a 3D metaphoric world from the information contained in a service model. It results in a graphical hierarchy where internal nodes are layout managers and leaves are 3D glyphs. In the CyberNet system, special objects called *adaptors* handle the mapping. Adaptors are dependent on the type of metaphor used for displaying the information and provide the necessary mechanism to automatically construct the virtual worlds. An example of a mapping is given on Figure 1.

In our current implementation, each metaphor is defined as a hierarchy of graphical components (i.e, a city metaphor is based on GCs called city, districts, streets, buildings, floors, etc.). We name *hierarchical mapping* the process of defining which service element (entity or service node) will be visualized using which metaphoric component.

The main idea behind the mapping process is to define a set of association rules for each service based on the type of each service element and its position in the service tree. In particular, since an entity may be a part of several relations (thus being located at more than one position in the service model tree), it may have several visual counterparts in the presentation domain. The adaptors implement the mapping association rules.

For instance, in the previous NFS example, the model mapping rules state that computers=districts, local disks=buildings, remote disks=houses, NFSimport-points=floors of the building, FileHandles=windows, and so on. So far, the mapping process is hard-coded but we are currently developing an automatic mapping process based on information and visual properties characterization.

The attributes of each entity must also be translated into visual information. This is the purpose of the *visual parameters mapping*. Each GC has a number of visual parameters that may be dynamically modified in order to display information (e.g., position, orientation, size, color, etc.).

A simple example is the mapping of a disk entity on a box glyph. In this example, the size of the disk could be mapped on the size of the box while the percentage of use of the disk could be mapped on the color of the box (green corresponds to 0%, yellow to 50% and red to 100%; the values in-between are mapped to the color transition between the limits). When defining the mapping rules, care must be taken to preserve metaphor coherency. For example, if one rule uses color for identification purposes, other rules can not use this visual parameter to represent a quantitative value.

In the previous NFS example, several mappings have been tested. Basically, one mapping is oriented toward NFS configuration while another is oriented toward NFS real-time monitoring. In the former, color and size are used to represent the type (e.g., system, user, etc.) the properties (e.g., backup strategy, speed, etc.) and the access rights (e.g., the rules for NFS exportation, etc.). The latter mainly shows NFS network statistics: mounted points, disks size and percentage of use, and so on. Both mappings are useful, depending on the operator task.

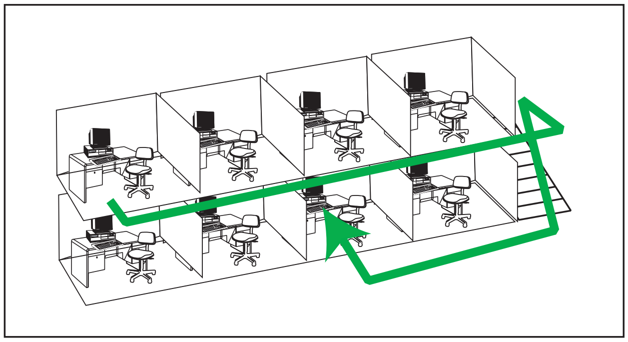
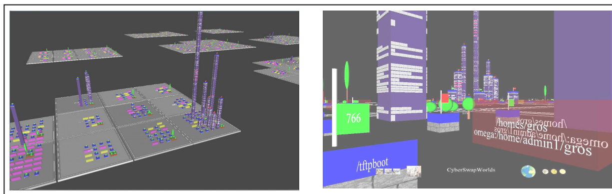


Figure 3. The metaphor-aware navigation path from one office to another.



Flying at the district level in a city metaphor

Walking at the street level in a city metaphor.

Figure 4. Navigating at different levels of the metaphor.

The result of such mappings is the following: all the NFS information is available in one virtual world (Figure 1 on the right). In the virtual world, there are several cities (one city per subnet). Each city comprises several districts (a district is a computer) and it is easy to identify servers (districts with buildings) and server loads (tall buildings). NFS clients are residential districts made of houses.

2.4. Changing the current metaphor

Whenever a mapping between a service model and more than one metaphor is feasible, it is possible to swap from one representation to another. This is very useful to experiment different metaphoric representations and determine the one that is most suited to the current service model or to the user's task. Figure 2 shows an example of different metaphoric representations of the same service.

3. Navigating in 3D metaphoric worlds

In this section we present an example of metaphor based navigation. We also give an example of different types of navigation in the same metaphor.

3.1. Metaphor based navigation

Anyone who has ever experienced 3D interfaces will agree that navigating in a 3D world is not a trivial task. The user interface of traditional 3D browsers provides navigation mechanisms that allow the user to modify the virtual camera parameters. Using these tools, it is common that, after some movements, the user gets lost in space and tries to restart from the beginning. We feel that the navigation schemes provided by traditional 3D browsers are too generic for our specific navigation needs. It is evident that when the user navigates from office to office in a virtual building he does not use the same navigation mechanisms that when he is exploring a landscape of data or studying the topological structure of a cone tree.

In the CyberNet visualization framework, all the metaphoric worlds are constructed using standard well-defined graphical components: layout managers are used to arrange 3D glyphs in space. For example, the same stack layout manager is used to align offices along a corridor and to stack floors on top of each other, in order to construct a building. However, the navigation along a corridor is not the same as the navigation from one floor to another. Our belief is that the navigation mechanisms should be associated to the graphical components. The objective is that the user navigates in the world with the mechanism most suited to the metaphor. We call this principle *metaphor-aware navigation*.

In order to be able to assist the user navigating, we have implemented this metaphor aware navigation mechanism in the context of the building metaphor (Figure 8). The building metaphor is used to present information about Eurecom Institute offices, staff and internal computer network. The offices and network devices location in the virtual world follow the real world location of these elements, but the virtual building is not an actual copy of the real life existing building.

The basic idea of our navigation system is to track the user current position in order to always be able to associate the user's current position with a node in the hierarchy. In addition, the system provides the user with a tool that allows him to select a destination node. This selection can be done, either by using a 3D embedded interface (e.g., by clicking on an object), or using an external menu. This menu is hierarchical and context sensitive.

The system determines the set of movements that must be done within the context of the current metaphor in order to go from the user's current node of interest to the new node of interest. These movements are dependent upon several user navigation modes but, basically, the user follows a logical path from his current position to his desired destination. The notion of path-based navigation is central to

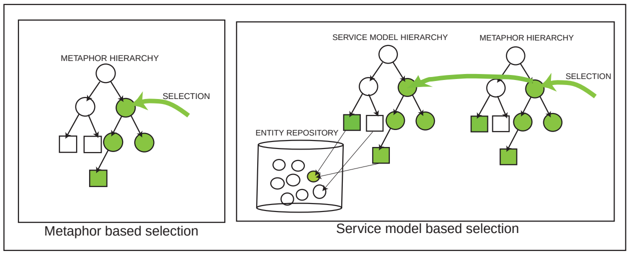


Figure 5. The selection mechanisms.

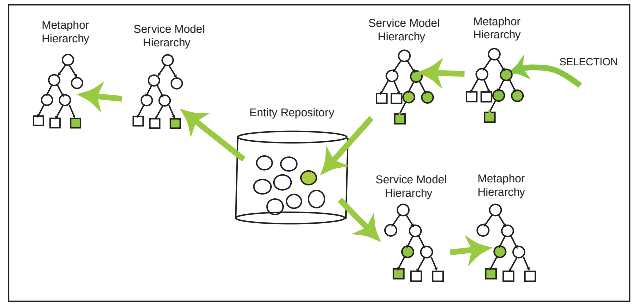


Figure 6. Propagating selection between views.

our navigation. The main interest of the path-based navigation is to give the user knowledge of the relative locations of objects in the virtual world. We already referred that it is of prime importance for the paths to be metaphor-aware. Navigating in a building is a good example of such a metaphor-aware navigation mechanism. When a user is in an office and wants to go to another office, the system automatically computes the logical path and animates the user position along that path (through the corridors and the stairs). Figure 3 shows an example of a metaphor-aware navigation mechanism.

3.2. Navigating at different levels of the hierarchy

In order to ease the navigation, the metaphor aware mechanism takes into account the actual level of the metaphor hierarchy the user is currently in. Different navigation schemes may be used for navigating at different levels of the metaphoric world hierarchy. This kind of navigation is helpful for easily acquiring survey knowledge, when navigating at a higher level. For instance, in a city, the user may navigate at the street level (Figure 4 on the right), walking in the streets in order to go from one building to another, or he may navigate at the district level, like a bird flying over the town (Figure 4 on the left). 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.

4. Interacting with 3D metaphoric worlds

This section presents some interaction mechanisms already implemented in the CyberNet project, namely the selection mechanism. We describe how the user can select objects inside a virtual world and across different worlds. We also briefly present other interaction mechanisms.

4.1. Selecting objects inside a world

Selection is one of the most important interaction mechanisms. The user should be able to select a node in the metaphor hierarchy using either direct manipulation or external menus. The user should also have the possibility to extend the selection to all the descendant nodes of the selected nodes.

In the building metaphor (Figure 8), these kinds of mechanisms can be used, for instance, to select all the computers that are located at a given floor. This kind of selection only allows to select groups of objects that belong to the same metaphor hierarchy sub-tree. For instance, selecting a building in the city metaphor (Figures 1 and 4) automatically propagates the selection to all the floors, walls and windows of that building. We call this selection mechanism a *metaphor based selection* (Figure 5 on the left) since the selection criteria are based on the metaphor structure.

However, the user often wants to select objects according to service model criteria, which may not be directly available from the hierarchical structure of the 3D world. In the city metaphor (Figures 1 and 4), when the user selects a building he may want to visualize which elements are related to that building. This is dependent upon the service model. We name this mechanism *service model based selection* (Figure 5 on the right).

In our previous NFS example (Figure 1), each floor of the building represents a client computer. NFS client computers are also represented as districts (generally, clients are residential districts). When selecting a building (which is related to a disk) the user may want to select all the districts/computers in all the towns/subnets that are related to that building/disk.

For that selection to be effective, the selection must be propagated from the metaphor to the service model down to the entity level. When a building is selected, the selection is transmitted to the corresponding local disk service node and all its descendant service nodes and related entities are also selected. When an entity is selected, the selection is retro-propagated to the service nodes that reference that entity and from these service nodes up to their graphical representation in the metaphoric world.

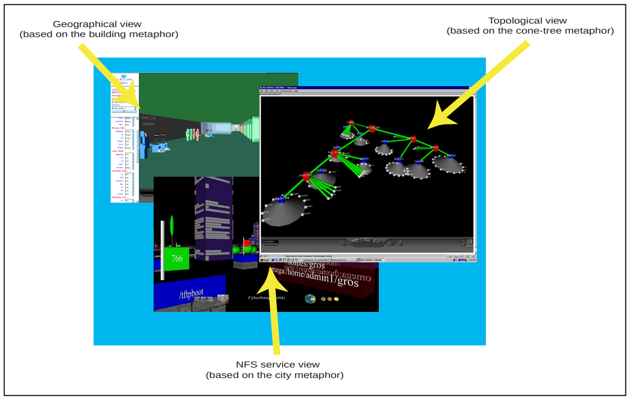


Figure 7. Working with several metaphoric worlds.

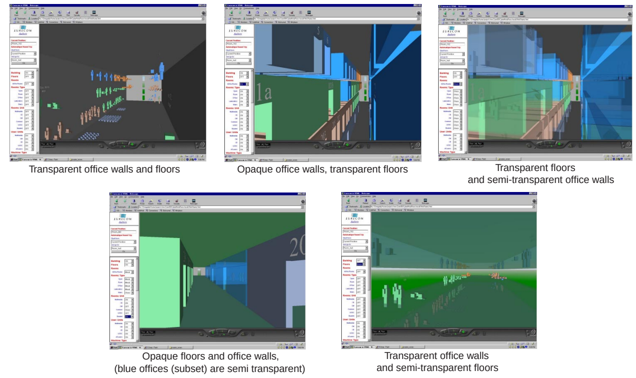


Figure 8. Modifying the appearance of the virtual representation of the Eurecom Institute building.

This mechanism can be used to help the user identifying the clients of a NFS server and, specially, the clients that are not located in the same subnet of the server. This is useful for performance tuning or for security purposes.

4.2. Selecting objects across different worlds

Being able to propagate the selection down to the service model and entities is a very powerful mechanism that can be exploited to extend the scope of the selection mechanism to several metaphoric worlds. The main idea is to allow the user to interact with several metaphors that represent different service models connected simultaneously to the same entity repository. We believe that one of the strengths of 3D metaphoric representation will come from the combined used of several interacting tools.

The multi-world selection mechanism (Figure 6) can be used to allow for specifying free selection criteria; i.e., selection criteria that are neither metaphor based nor service model based. The user may view the same network according to several views, each view being based on the metaphor that is most suited to represent each service model (Figure 7). In one view he may visualize the building and geographic location of computers and network devices. In another view, he can visualize the topology of the network (using for example the cone-tree metaphor). In a third view, the user can monitor the NFS service using the city metaphor.

With the multi-world selection mechanism, it is possible to select a network device (e.g., a hub) in the cone-tree view where they are more easily identified, and all the computers connected to that hub will be selected in all the three views. The user can also select the clients of the a NFS server in the city view and locate them in the building and in the topology views. The selection mechanism can be further extended in order to allow the user to combine several criteria (by ANDing or ORing successive selection actions).

4.3. Other interaction mechanisms

We found that being able to interact with the world in order to modify its appearance was important. An example of such a study has been done within the context of the building metaphor. We felt that having a model of the building and being able to navigate and interact with the objects according to the actual building is important but sometimes restrictive. Virtual worlds should bring more possibilities than just mimic the real world. For this purpose, we have developed mechanisms that allow the user to have control over the world appearance itself.

For instance, in the building metaphor, the user may modify the transparency of walls and grounds. The user may render all the ground and office walls transparent in order to easily observe the contents of all the building. He may also render some offices partially transparent according to criteria such as the department they belong to, etc. Further stretching the interaction, the user may even decide to “rebuild” the building by placing a department per floor (in real life, the departments at Eurecom Institute are spread over the different floors). Figure 8 shows some examples of the above interactions.

5. Conclusions and perspectives

We have presented experiments done in the context of network management in order to study the advantages of 3D metaphoric virtual worlds for monitoring large volumes of multidimensional information. We have described how the system automatically builds virtual worlds for information display. We have also shown how the navigation and interaction are dealt with in order to ease user’s tasks.

Preliminary feedback from users showed that 3D metaphoric visualization seems promising. It eases problem detection and understanding. However, we are still looking for new visualizations that make the most of the opportunities of virtual environments. We are developing new metaphors and associated 3D glyphs and layout managers with special focus on their adaptive representation and navigation capabilities. New interaction mechanisms also need to be studied further.

Although this work has been done within the context of network management, every application that involves the user to interact with huge amounts of dynamically structured information may be targeted (such as stock exchange, bank trading or web administration). We must also refer that the end user may access the system from anywhere since it only requires a VRML (Virtual Reality Modeling Language) enabled WWW browser.

Several topics are currently under study. Data characterization is one of them. At current implementation, data values mappings onto visual parameters are hard-coded. We are developing a visual parameters taxonomy and a data characterization. These classifications will allow us to implement generic adaptors, and thus automatically map real world data values onto visual parameters, according to the current metaphor.

Further work also has to be done in the field of user interaction and combined use of different tools. We also have to study how to add persistence to the system in order to be able to playback and analyze some crucial sequence of events off-line. We also want to provide ways to interact with the real world (e.g. kill a process on a workstation).

The CyberNet project is supported by France Telecom and the Eurecom Institute. The CyberNet project web site is: http://www.eurecom.fr/˜abel/cybernet. The authors would like to thank: B. Lugeon, P. Miauton and J. Pasini for their work on the building metaphor; M. Girardot and L. Oria for their work on the cone-tree metaphor; and P. Garcia, S. Monterde and R. Vincent for their work on the city metaphor.

References

[1] 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.

[2] M. C. Chuah and S. G. Eick. Information rich glyphs for software management data. *IEEE Computer Graphics & Applications*, 18(4), July – Aug. 1998. ISSN 0272-1716.

[3] K. C. Cox, S. G. Eick, and T. He. 3D geographic network displays. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 25(4):50–??, Dec. 1996.

[4] M. Crossley, N. J. Davies, R. J. Taylor-Hendry, and A. J. McGrath. Three-dimensional internet developments. *BT Technology Journal*, 15(2):179–193, Apr. 1997.

[5] L. A. Crutcher, A. A. Lazar, S. K. Feiner, and M. X. Zhou. Managing networks through a virtual world. *IEEE parallel and distributed technology: systems and applications*, 3(2):4–13, Summer 1995.

[6] K. M. Fairchild, S. E. Poltrock, and G. W. Furnas. SemNet: Three-dimensional graphic representation of large knowledge bases. In R. Guindon, editor, *Cognitive Science and its Applications for Human-Computer Interaction*, pages 201–233. Lawrence Erlbaum Associates, Hillsdale, New Jeresy, U.S.A., May 1988.

[7] N. Gershon and S. G. Eick. Visualization's new tack: Making sense of information. *IEEE Spectrum*, pages 38–56, Nov. 1995.

[8] W. M. Newman and M. G. Lamming. *Interactive System Design*. Addison-Wesley Publishing, Wokingham, England, 1995.

[9] 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.

[10] R. M. Rohrer and E. Swing. Web-based information visualization. *IEEE Computer Graphics and Applications*, 17(4):52–59, July/Aug. 1997.

[11] S. Zhai, W. Buxton, and P. Milgram. The partial-occlusion effect: utilizing semitransparency in 3D human-computer interaction. *ACM Transactions on Computer-Human Interaction*, 3(3):254–284, Sept. 1996.