Research Into A Visualization of an IPv6 Network

Undergraduate Thesis

Keio University Faculty of Environmental Information

Tokuda, Murai, Kusumoto and Nakamura Laboratory

February 1999

Keio University Faculty of Environmental Information

Shu NAKAMAE

baseman@sfc.wide.ad.jp

Abstract

With the growing demands on address space becoming more and more urgent, the utilization of IPv6, as the next generation of IP, is becoming eminent. Much research is presently going into the realization of a IPv6 world. Though as its successor, IPv6 is similar in many ways to IPv4, many factors must be taken into consideration during this transition period.

One such factor is the smooth management of the IPv6 network. As the answer to the lack of address space, IPv6 holds a potential for a network the size of which has never before seen the light of day. The manner in which the network is managed will, to a great extent, determine whether this huge network will function or not.

This research holds its goal in the creation of a visualization for an IPv6 network, as a means of aiding in network management. The objectives of such a visualization would be to

1. display the distinct hierarchy embedded in the IPv6 address architecture,

2. show topographical data simultaneously to the above mentioned hierarchy,

3. make peripheral data easily available, and

4. in so doing aid in the process of network management.

After probing the possibilites of various types of three-dimensional visualization this research choses and implements a system of creating a visualization unique to IPv6. In so doing, the smooth management of an IPv6 network is enabled by giving the manager a more coherent grasp of the network and thereby making decisions such as address assignment and allocation simpler.

概要

アドレス枯渇に伴い次世代 IP としての IPv6 の実用化は日に日に近づいてきている。現在この IPv6 世界の現実化に向かい様々な研究が行われている。その後継者として、IPv6 は IPv4 に似てはいるが今日の移行期において、二つのプロトコルの差は常に念頭に置かなければいけないことである。

IPv6 ネットワークの管理というのもこの差の一つである。アドレスの枯渇を解消するネットワークとして IPv6 は今までになく巨大な規模になり得る可能性を秘めている。ネットワークの運用の有り様がこの巨大ネットワークの実用性につながるのである。

本研究は次世代 IP によって構築されたネットワークの管理を支援する手段としての可視化を目的とする。この可視化は次の要素を満たすよう設計される。

1. IPv6 のアドレス機構に組み込まれている階層性を明らかにする。

2. 上記階層構造と共にトポロジー情報も見えるようにする。

3. ネットワーク関連の周辺情報の獲得を用意にする。

4. 上記三条件によりネットワーク管理の仕事を楽にする。

各種三次元可視化を検討した後、本研究では IPv6 に特有なヴィジュアライゼーションを作り出す機構を構築する。これによりネットワークやアドレスの管理者により分かりやすい形でネットワークを提示する。その結果アドレス割り振りなどの祭に管理に関わる判断を楽にする。

Contents

1 Preface **1**
- 1.1 Background . . . 1
  - 1.1.1 IPv4 . . . 1
  - 1.1.2 IPv6 . . . 2
- 1.2 The Goals of This Research . . . 4
- 1.3 The Contents of This Paper . . . 5

2 Present Status **6**
- 2.1 IPv6 Network Management . . . 6
- 2.2 Visualization in Network Management . . . 8
- 2.3 The Merits of a Visualization . . . 10

3 Related Studies **11**
- 3.1 Different visualizations . . . 11
  - 3.1.1 Surface plots . . . 12
  - 3.1.2 Cityscapes . . . 12
  - 3.1.3 Cone trees and Cam trees . . . 14
  - 3.1.4 Information Cube . . . 14
  - 3.1.5 Fisheye Views . . . 16
  - 3.1.6 Self Organizing Graphs . . . 17
- 3.2 Analysis of the Visualizations . . . 17

4 The Design of an IPv6 Visualization **20**
- 4.1 The Design Philosophy . . . 20
  - 4.1.1 The Problem . . . 20
  - 4.1.2 The Solution . . . 21
- 4.2 IPv6's Address Hierarchy . . . 21

4.3 The Necessity of a 3D Expression 24
4.4 Requirements for a 3D Visualization 27
4.4.1 The Contents and Considerations 27
4.5 The Actual Visualization 28
4.5.1 The Visualization Concept 28
4.5.2 The Visualization System 28
4.5.3 The Visualization Scenario 29

5 The Implementation **31**
5.1 The Implementation Environment 31
5.2 The Interface 32
5.2.1 The Visualization Protocol 32
5.2.2 The Different Modules 33
5.2.3 SQL 37

6 Evaluation **38**
6.1 The Objective and Its Fulfillment 38
6.2 Problems in Implementation and Future Refinements 41
6.2.1 Problems 41
6.2.2 Future Refinements 42

7 Conclusion **44**
7.1 Summary 44
7.2 Conclusion 45

Bibliography **48**

List of Figures

2.1 The Purpose of a Visualization 9

3.1 A Surface Plot 13
3.2 A Cityscape 13
3.3 A Cone Tree 15
3.4 An Information Cube 16

4.1 A Sample of the IPv6 Hierarchy 22
4.2 A 2D Chaos 25
4.3 Topology + Hierarchy = 3D 26
4.4 The Design Concept 30

5.1 A Screen Shot of the Interface 34
5.2 The Modules 36

6.1 A Cone Tree vs An Information Cube 39
6.2 A Prototype of The WIDE 6bone In 3D 40

List of Tables

3.1 Summary of the Visualizations 18

4.1 WIDE 6bone Address Assignment 24

5.1 The Implementation Modules 35

Chapter 1

Preface

1.1 Background

1.1.1 IPv4

With the advent of the Internet, communication possibilities have blossomed at an amazingly rapid rate. Rare is the person who does not benefit from it in this modern age of information and technology. The success of the Internet can be laid, for a great part, at the feet of the basic protocol on which it functions, IPv4 or Internet Protocol Version 4. Developed in the late 70's, IPv4 has come to see such wide use mainly owing to a combination of its simple structure, and extreme flexibility. At the same time however, it is these very same factors which have propagated the faults of IPv4. IPv4, though immensely successful, was never-the-less a protocol designed and created in an age and era where computers were still severely limited, both in processing power and memory. This meant that though the fundamental design was fine, IPv4 carried with it a few inherent faults. These faults include

things such as lack of address space. The 32bit address space, an address space seemingly over abundant at the time, proved woefully inadequate all to quickly. Another problem with IPv4 was its lack of scalability. In its original designs the protocol was designed to support a rough scalability. Unfortunately, the all pressing need for more address space led to temporary solutions which all but negated this ability.

From the point of view of address management, IPv4 is run on a completely centralized administration system. The job of delegating IP addresses is strictly managed by IANA(Internet Assigned Numbers Authority) and carried out by a few regional NICs. Parties who wished for a new IP address were required to apply to this central authority in order to obtain it. With only a 32bit address space, such a centralized management was possible, if barely. To help with the workload many management aids such as visualizations and tools such as ping and traceroute came into use. In IPv4, visualizations were generally a 2 dimensional topographical map of the network. Though in it's original form, IPv4 supported a hierarchy[1], with the introduction of CIDR (Classless Inter-Domain Routing) the layers became meaningless and the network could be thought of as one vast spread of connected nodes, all on the same plane. The topographical maps therefore resembled a spiders web, with nodes being connected to various points by lines going in different directions.

1.1.2 IPv6

History and Introduction of IPv6

IPv6 was introduced by the Internet Engineering Task Force in 1994 as the IP next generation which would solve the inherent problems of IPv4 at a

fundamental level.

The most noticeable problem IPv6 addresses is of course the lack of address space. It does this by providing a 128 bit address space[2]. This makes it 4 times the length of the conventional IPv4 32 bit address and roughly $4 * 10^{27}$ larger in actual usage.

Differences In Network Management

IPv6 also has other outstanding merits, such as a simplified header format and automation. The simplified header decreases the work load of each relay node, thereby more than making up for the increased address length and facilitating quick transmission.

Many of the merits of IPv6 have much to do with the manner in which the network is managed. This is because most of the problems in IPv4 which IPv6 was created to overcome were not problems in how the protocol functioned, but in what the protocol was prepared to handle. IPv4 was simply not made to handle networks of the gigantic proportions of today. It goes to reason then, that many of IPv6's new capabilities were those which allowed it to handle larger sizes.

For instance, due to its large address space, a sizeable increase in setup and management workload is inevitable, making mistakes in various configurations more likely to happen. Automation of various aspects of an IPv6 environment, something not readily available in IPv4, is an important factor in ensuring smooth and proper operation.

Another major point of difference in IPv6 network management, is its scalability. The management of IPv4 was completely centralized, creating an inflexible structure as well as an enormous work load. Not only is such a centralized system impractical to maintain for IPv6, it is impossible. The

workload would be much too great to support in any one place. The solution is to make the whole structure scalable through Aggregatable Address Allocation[3]. With Aggregatable Address Allocation, the whole network on a world wide scale becomes completely hierarchical, with the levels being dictated by the address architecture. The whole system is in turn managed by a distributed registry system placed at various aggregation points, thus bringing us up to date on the present status of the relevant aspects of IPv6.

1.2 The Goals of This Research

With its greatly increased address space, even if just by sheer size, a full scale IPv6 network can be considered to be much more complex than that of a conventional IPv4 network. On the other hand, the clear cut hierarchy allows for the network to be grasped in a very simple organized way. Whether the network can be handled at the abstract level as a complex and convulsed object or a simple and easily understandable object depends on whether the peculiars of IPv6 are correctly taken into consideration. However, because IPv6 is the predecessor of IPv4, its hierarchy is often neglected when considering the network as a whole. This can be seen when looking at IPv6 network visualizations which much resemble a extremely complex IPv4 visualization. Visualizations should be varied according to the object they depict in order to express their differences. Yet in spite of this fact, network visualizations unique to IPv6 have yet to be created.

The goals of this research is to provide such an IPv6 unique visualization, a visualization designed with the specifications of IPv6 in mind.

The visualization will concentrate on the following points.

1. depicting both the hierarchy and the topology of the IPv6, and

2. helping to keep the overflow of management related data at a comprehendable level.

1.3 The Contents of This Paper

This paper will discuss the creation of what is considered as a much needed IPv6 visualization. After a brief summary of the background that led up to this research, chapter 2 will describing the way an IPv6 network is managed and the role visualization plays in such management. Chapter 3 will explore the different types of possible 3 dimensional visualizations as related studies, and then in chapter 4, the design and architecture of the visualization is explained. At the end, in chapter 5 the actual implementation will be described, and chapter 6 will see that the implementation is evaluated and that future refinements are suggested.

Chapter 2

Present Status

This chapter first discusses the present state of address and network management for IPv6 in comparison to that of IPv4. It then goes on to explain the role visualization plays in such management and the merits of having a network visualization.

2.1 IPv6 Network Management

Though much of the specifics for IPv6 itself have been already set down, the manner in which the IPv6 network is to be run and managed is still a matter of much debate and research. For instance a registry system for the assignment of addresses is definitely needed. Though how such a registry system is to be run is still up for debate, certain factors and conclusions are inevitable so long as the registry system must forefill a certain set of functions. These can be reached by looking into the address management of IPv6's predecessor IPv4. The following is a list of such functions.

- Delegation and management of IPv6 addresses

  - organization name
  - address block

- Maintaining of peripheral data
  - routing information
  - site manager
  - type of connection

The registry system for IPv4 holds much the same kind of data as the list above, and makes them public for the use of network management. This is done through means such as the WEB and whois[4]. However as the IPv4 network grew to hitherto unforeseen proportions, even as the need for more address space become more and more urgent, it was evident that the centralized management of addresses was becoming more and more difficult.

IPv6 came about as a solution to the demand for more address space that IPv4 could not provide for. It therefore goes to reason that if address management was difficult in IPv4 because of its great size, the address management for IPv6 would be many times that in difficulty. IPv6 would support a infinitely larger, multi-layered network. The registry data items from the list above would have to be maintained for each of the myriad organizations registered.

Another factor is that much of this information is often accessed in tandem with other elements in order to create and grasp a coherent 'picture' of the network in question. With the onslaught of data inherent to the large address architecture of IPv6, ways in which to more easily manage this data becomes more and more important.

Automation of many of the functions is a step which has already been taken in many of the IPv6 stack implementations. Automation reduces the risk of human error, by delegating repetitious actions to the infallibility of a computer.

Visualizations are also another means which reduces risk of human error. It does this by creating an environment which makes the human decision making process easier. Visualizations allow a user to 'see' the information as a picture, a means which permits a much larger sphere of data comprehension then simple text.

2.2 Visualization in Network Management

In the information saturated world of today, the problem has become not so much a matter of finding pertinent information but filtering out the essentials from the overwhelming flood. Such filtration is called data mining and has become more and more a topic of concern.

The manner and facility in which data is retrieved can be influenced on two fronts. That is the provider side and the accessor side. The provider of the information attempts to make information more accessible and easier to filter by structuring the offered information by means such as creating data bases and search engines. Visualization techniques on the other hand focus on augmenting the accessor or user sides capability in implementing a search. Information visualization systems will allow the user to explore the data universe at a more abstract level, harvesting interesting looking items as they are found. These systems will allow users to make use of their cognitive, perceptual and intuitive skills to find data which may be of interest but could be missed by search algorithms because they are not directly relevant to the

query.

As a close to real life interface, 3D information visualization techniques and research systems have been developed to aid in the comprehension of large scale information systems.

A visualization is necessary in an IPv6 network in order to understand its many aspects. In the case of the IPv6 network, the large scale information system is the network itself along with the many factors that relate to it. The managing of a network requires the grasp of various aspects of the network, from the actual address to the layer which it belongs to, from the points of connection to the were the glitch is, from the contact address of a site manager to routing information. All of these separate aspects all go into the one task called "network management". A valid visualization of the network allows the manager to comprehend the various aspects at once (figure 2.1).

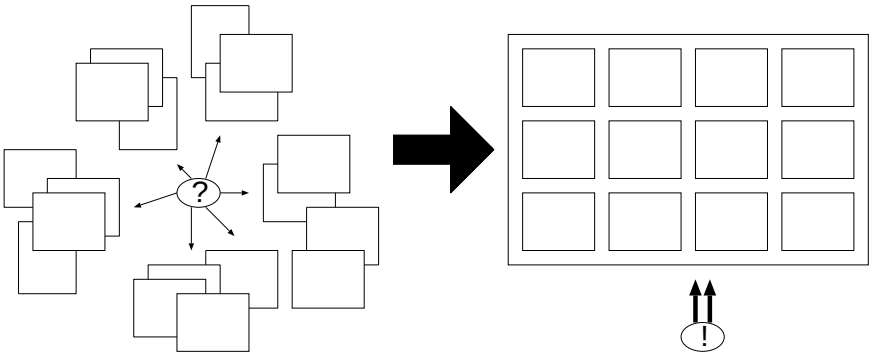

Figure 2.1: The Purpose of a Visualization

2.3 The Merits of a Visualization

Whether in IPv4 or IPv6, a good visualization of the network is often useful and at time outright necessary to facilitate the smooth management of the network. This is because the larger and more complex a network becomes, the harder it is for the manager to comprehend what he or she is working with. As the old adage "a picture paints a thousand words" suggests, visualizations have the ability to make manageable an amount of information which would other wise be incomprehensible due to its size. Network topology maps have long been used in IPv4 as a means of understanding the structure of the Net, as well as for assigning addresses and DNS entries.

As the IPv6 hierarchy reflects its address architecture, all matters related to address management, such as assignment and renumbering are affected by it.

As a network visualization a three-dimensional view of the network will also prove to be a merit to network managers as well.

For instance, when there is a network glitch, showing traceroute results on such a 3D map would greatly facilitate discovery of which aggregation point is faulty, as well as of which portion of the network tree is inaccessible.

Chapter 3

Related Studies

This chapter describes the different types of possible three-dimensional visualizations[5]. The visualizations are roughly grouped according to their capabilities and a visualization type for IPv6 is considered at the end of the chapter.

3.1 Different visualizations

With the high processing speed of today's computers and the leaps forward in spatial expression theory, 3D graphics have become more and more popular and often seen. Not only are these 3D objects used as attractive "pictures" for games and cinemagic, they are important tools as an interface for data comprehension. There are now probably almost as many different ways, if not more, to create 3D visualizations as there are ways to draw simple 2D graphs. Roughly grouped however, 3D visualizations can be grouped into 'mapping techniques', 'information presentation techniques' and 'dynamic information visualization techniques'[5]. Mapping techniques like surface plots and cityscapes use some aspect of the data items to create a mapping of ob-

jects within the visualization. Cone trees, cam trees and information cubes are examples of information presentation techniques which concentrate on the appearance, accessibility and usability of the data. Information presentation techniques attempt to provide a user with a interface which allows for intuitive searching. Finally, visualizations such as fish-eye views and self organizing graphs constitute dynamic information visualization techniques, where the visualization is endowed with dynamic properties allowing for automatic responses to changes in data and or user actions.

3.1.1 Surface plots

Surface plots are the most simple 3D extensions of a traditional 2D plot graph. Data triples are plotted into a map with three co-ordinate axes X, Y and Z. Usually the X and Z axes are used to plot two regular sets of data such as days of the week and time of day. The vertical Y axes is used to plot the variable data. The points thus plotted out are then are connected to form a kind of mesh or a surface, creating a sort of landscape (figure 3.1).

Surface plots make the pattern in any data set evident. This means that irregularities are easily seen.

3.1.2 Cityscapes

Cityscapes are thus called because they are a 3D version of a conventional bar chart and have the appearance of buildings (figure 3.2). They are similar to surface plots, having the variable factor plotted on a uniform 2D horizontal plane in the form of vertical bars or blocks.

Cityscapes are essentially the same as a 2D bar graph or chart. However it provides the capability to display 3 inter-related factors instead of just two.

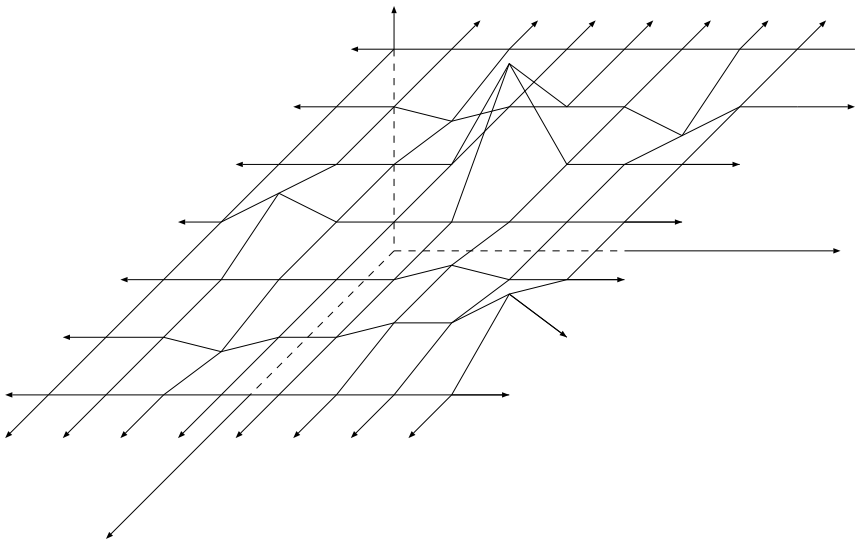

Figure 3.1: A Surface Plot

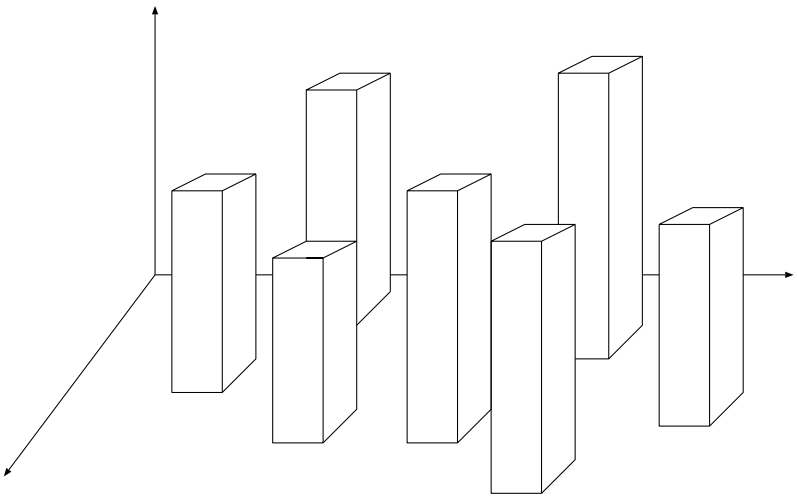

Figure 3.2: A Cityscape

For instance a teacher could plot grades by student, date, and test score. In so doing, the teacher is able to see not only the overall average performance of a student singly, but in comparison with others.

3.1.3 Cone trees and Cam trees

Cone and cam trees are a direct three-dimensional translation of the now familiar hierarchical tree structure. The two are different only in orientation, cone trees being vertical while cam trees are horizontal.

A cone trees places the root node at the apex of a conical structure near the top of the display. All child nodes are then distributed at equal distances along the base of the top cone, creating smaller cones. The process is repeated for each hierarchical layer with base diameter of the cones diminishing toward the lower levels to ensure that all the nodes may be fitted in (figure 3.3).

A cone or cam tree will rotate to allow close inspection of any given node. However, smooth rotation must in ensured for a sudden change in the view of the tree causes disorientation on part of the user.

3.1.4 Information Cube

The information cube is another means of visualizing hierarchical structures. An information cube is created by using nested translucent cubes. The boxes contained within the outer shells are the child nodes, which in turn contain child nodes of their own. Nodes are allotted spacing according to interest such as importance of information, size of data, or size of the contained hierarchy (figure 3.4).

With the hierarchy being express as nested translucent cubes, variations in level of transparency control the 'depth' of the visualization which directly

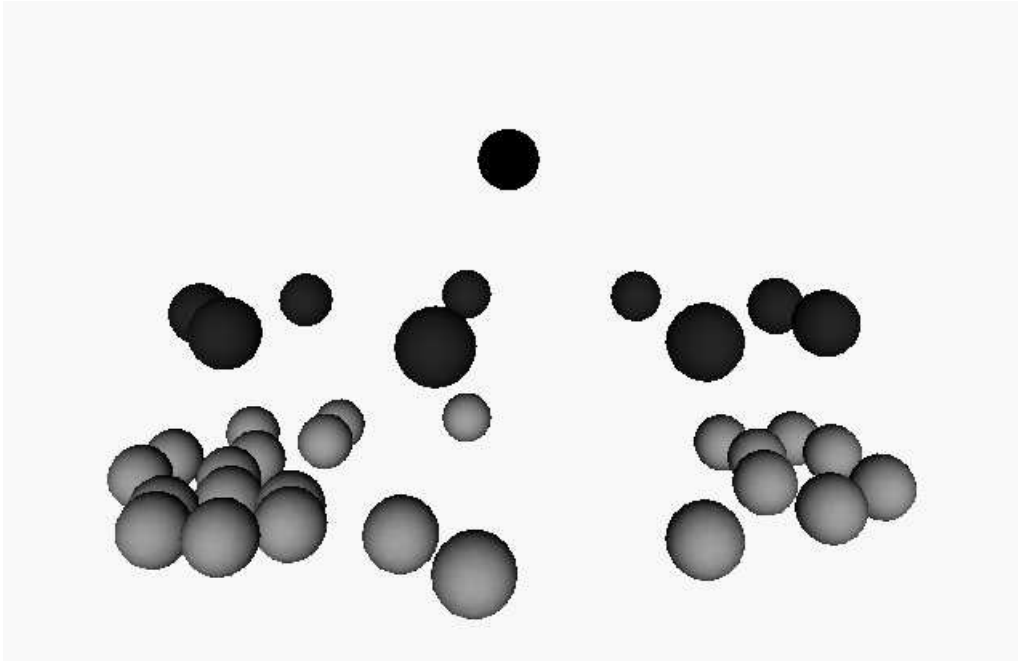

Figure 3.3: A Cone Tree

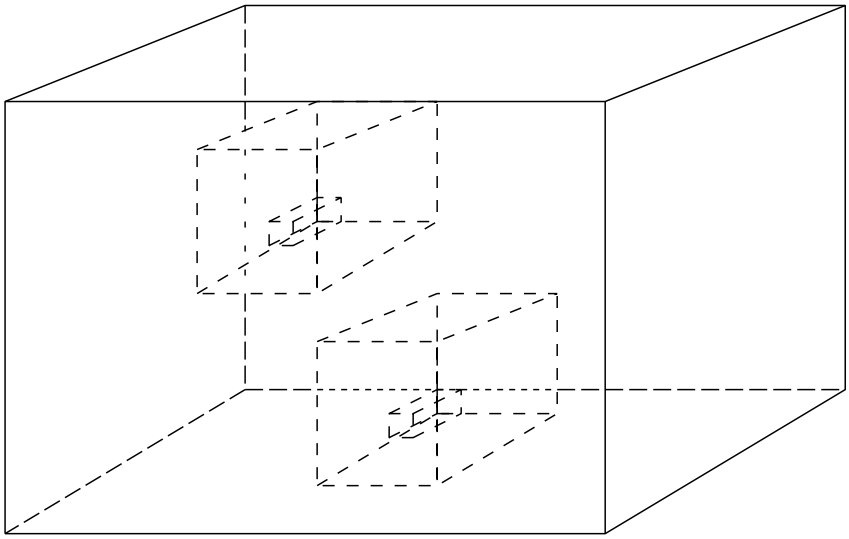

Figure 3.4: An Information Cube

leads to the amount of information being conveyed. Both transparency and shading is used to control the informational content with each cube. This allows the user to view information not of only of the current node but also of the child nodes, with the lower layers gradually fading from view. Without this slow fade out, a many layered structure would become too complex for human comprehension. The title of each node, being represented as a 2D tile, is pasted onto the walls of each cube. Cubes can contain anything sort of information, even other types 3D visualizations.

3.1.5 Fisheye Views

Fisheye views are thus named as they mimic the effect of a wide angle fish-eye lens. The lens distorts the view by greatly magnifying objects toward the center of attention while peripheral objects are magnified to a much lesser

extent. This results in allowing minute inspection of the objects placed within the center of attention while still showing the object on the periphery thereby keeping the context and positioning of the whole.

Because fish-eye techniques allow the nodes of interest to be brought closer to the viewer, they can prove to be useful in visualizing large graphs. The closeup view allows for the details of the important or pertinent node to be seen in great detail while being able to maintain a perspective of the whole structure.

3.1.6 Self Organizing Graphs

Just as their names suggest, self organizing graphs are a technique which create and lay out graphs automatically. While conventional graphs require a function or routine to create a suitable layout while fulfilling certain aesthetic conditions, self organizing graphs allow the graph itself to create the layout. The thus created model may and generally is at first unstable, however it settles into a stable equilibrium as all factors are present and newly inputed are taken into consideration. Furthermore these graphs need not follow the forms of any actual physical system, but may be modified to reflect different criteria such as efficiency, speed, accuracy and aesthetics.

Self organized graphs are useful when displaying large hyper-structures which represent multiple connection and interacting components.

3.2 Analysis of the Visualizations

As stated earlier there are indeed many different types of 3D visualizations, each having its own merits and weaknesses. Or perhaps it would be more correct to say that as a result of the high degree of expressibility offered by

present day technology, the visualization are highly specialized, each catering to certain type of data expression, to highlight the information factor considered the most pertinent and important. Table 3.1 below shows a quick summary of the strengths of each of the visualizations presented in this chapter.

Table 3.1: Summary of the Visualizations

Name	Type	Description
Surface Plot	mapping technique	a mesh of plotted points useful for pattern finding
Cityscape	mapping technique	3D extension of bar charts
Cone/Cam Tree	information presentation	3D extension of the hierarchical tree structure
Information Cube	information presentation	nested cube representing hierarchical layers
Fisheye View	dynamic info visualization	brings center of view into greater magnification
Self Organizing Graph	dynamic info visualization	automatic layout according to data updates

The trick was now not to create a new type of visualization but to correctly choose one which best suits the purposes in mind. For this particular research the objective lay in transmitting as much information as possible to maintain the immense at manageable proportions. To this end the obvious solution was to utilize an information presentation technique such as the cone/cam tree or the information cube. Which of the two was actual used is will be explained in the next chapter, but as visualization which express hierarchy while allowing more a great deal more peripheral information, both

were excellent candidates.

Chapter 4

The Design of an IPv6 Visualization

This chapter explains the design and concept behind the new IPv6 visualization. It goes about this by explaining the factors which the visualization must express and the process from raw data to visualization.

4.1 The Design Philosophy

4.1.1 The Problem

Conventional two-dimensional IPv4 visualizations simply cannot handle the data influx required to be contained in a IPv6 visualization. However at present IPv6 visualizations have followed much in the manner of IPv4 visualizations, mainly because the two versions are so similar and because thus far a valid manner of IPv6 unique visualizations has yet to be proposed. The problem with this is that though IPv4 and IPv6 share many characteristics, IPv6 has many added capabilities which must be expressed in the visualization. In order to support a larger network, IPv6 possesses a much more

complex network structure than IPv4.

On the other hand though the conventional IPv4 visualizations are invalid, any and all means of facilitating its management are necessary and helpful. This is because of the potentially humongous workload inherent in the running of an IPv6 network,

Therefore it is vital that a visualization which caters to IPv6's distinct idiosyncrasies be created.

4.1.2 The Solution

The obvious solution is to try a completely new approach to network visualizations.

The purpose of this IPv6 visualization is to create a way in which the hierarchy inherent in the IPv6 address architecture becomes at once apparent, as well as to help make the many facets of network management data easily accessible. The hierarchy is deeply related to the network mangement, for instance in terms of address assignment policy. However regular topographical information is by no means unnecessary. How then to create a visualization which could handle both? The conclusion is to create a three-dimensional visualization.

4.2 IPv6's Address Hierarchy

IPv6 was built with a distinct hierarchy (figure 4.1) embedded within its very address structure. This hierarchy was thus placed as a means to ensure scalability with the new network.

The 128bit IPv6 address is split up at predefined levels, defining the hierarchy. The top level is called the TLA ID, standing for Top-Level Ag-

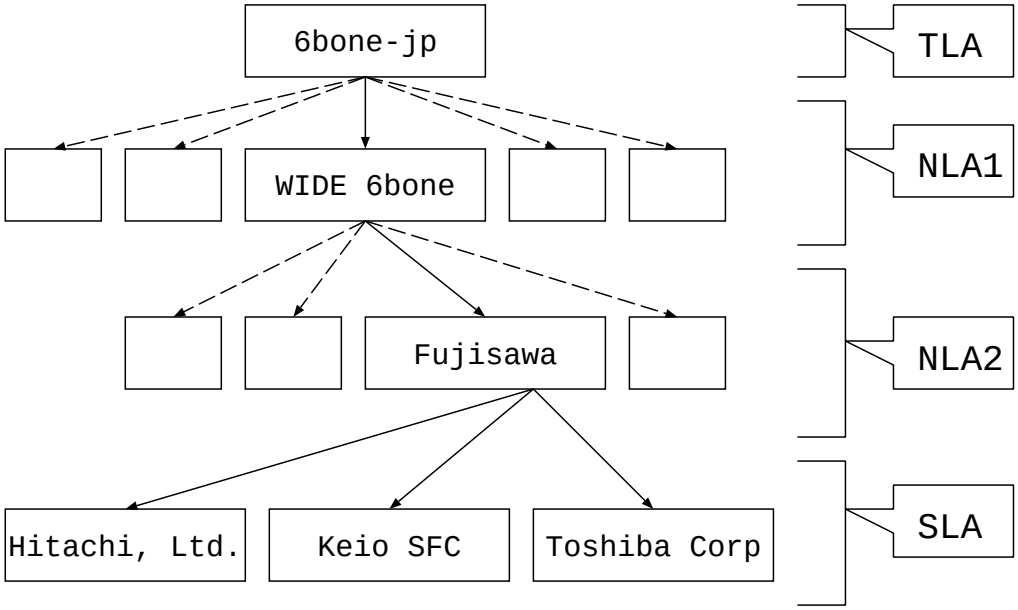

Figure 4.1: A Sample of the IPv6 Hierarchy

gregation Identifiers, the second level NLA ID, standing for Next-Level Aggregation Identifiers, and the bottom level SLA ID, standing for Site-level Aggregation Identifiers[6].

TLA ID's are assigned to organizations which provide a transit topology. In other words a organization with only leaf topology is not eligible for a TLA ID. The assignment of a TLA ID does not mean ownership, but does imply stewardship over the given address resource.

The second level NLA ID's are used to create an addressing hierarchy by organizations assigned a TLA ID. They can also be split up further more according to the address assignment policy of the TLA responsible for managing the NLA. This means that an network can have as little as only one NLA level and as many as 3 or 4 depending on policy and network requirements.

An example of this is the WIDE 6bone. The WIDE 6bone is an experimental IPv6 network being run and maintained in Japan. The WIDE 6bone is itself, run under the 6bone JP, a Japan wide experimental IPv6 backbone. All experiments here at the 6bone JPare done under a temporary TLA of 16bits, 3ffe. A pTLA, standing for pseudo TLA, of 8bits was set up to act instead of the temporary TLA. From there, as seen in table4.1, the NLA portion in the WIDE 6bone has been split up into 3 layers called, NLA1, POP-ID (Point Of Presence ID), and ORG (ORGanization ID) each being given an address block of 8bits, given them an address prefix of 32bits, 40bits, and 48bits respectively.

Under the pTLA 05 assigned to the 6bone-jp, the WIDE 6bone run by the WIDE Project has received a 01 designation as a NLA1. Under the WIDE NLA1, there are now 9 POP-ID's and 30 ORG-ID's, making a total of 5 layers, from pTLA to SLA.

Table 4.1: WIDE 6bone Address Assignment

TLA	pTLA	NLA1	POP-ID	ORG-ID	SLA
3ffe	05	01	??	??	????
16bits	8bits	8bits	8bits	8bits	16bits

4.3 The Necessity of a 3D Expression

To visualize an IPv6 network by displaying both its hierarchy and its topology, a three-dimensional interface is the only one which can be seriously considered. The reasons are many fold and while some are merely useful others are vital to the management of an IPv6 network.

The primary problem with visualizing an IPv6 network by conventional means is the lack of ability to express hierarchy. With a network designed to support hierarchy as means of clean and organized address allotment, there is something fundamentally wrong with not being able to see these layers. At the same time, topographical data was not something that could be ignored. The problem was that displaying hierarchical information on a conventional two-dimensional create a extremely chaotic map.

Attempts have in fact been made to express the hierarchy with a conventional two-dimensional topology map, such as is used in IPv4. These entail hierarchical expression by such means as color coding. However these attempts at best convey very little and at worst only serves to make the already much convulted map even more confusing.

For instance figure 4.2 shows the network visualization of the experimental WIDE 6bone used at present. As can be seen, even for a small experimental network such as this, the map can become extremely complicated.

What is not so evident is that this map shows a network of 3 different layers. The shaded nodes in dotted lines belong to the highest layer (NLA1), the large nodes in the middle the next (POP-ID), and the small nodes on either side and on top the lowest (ORG-ID).

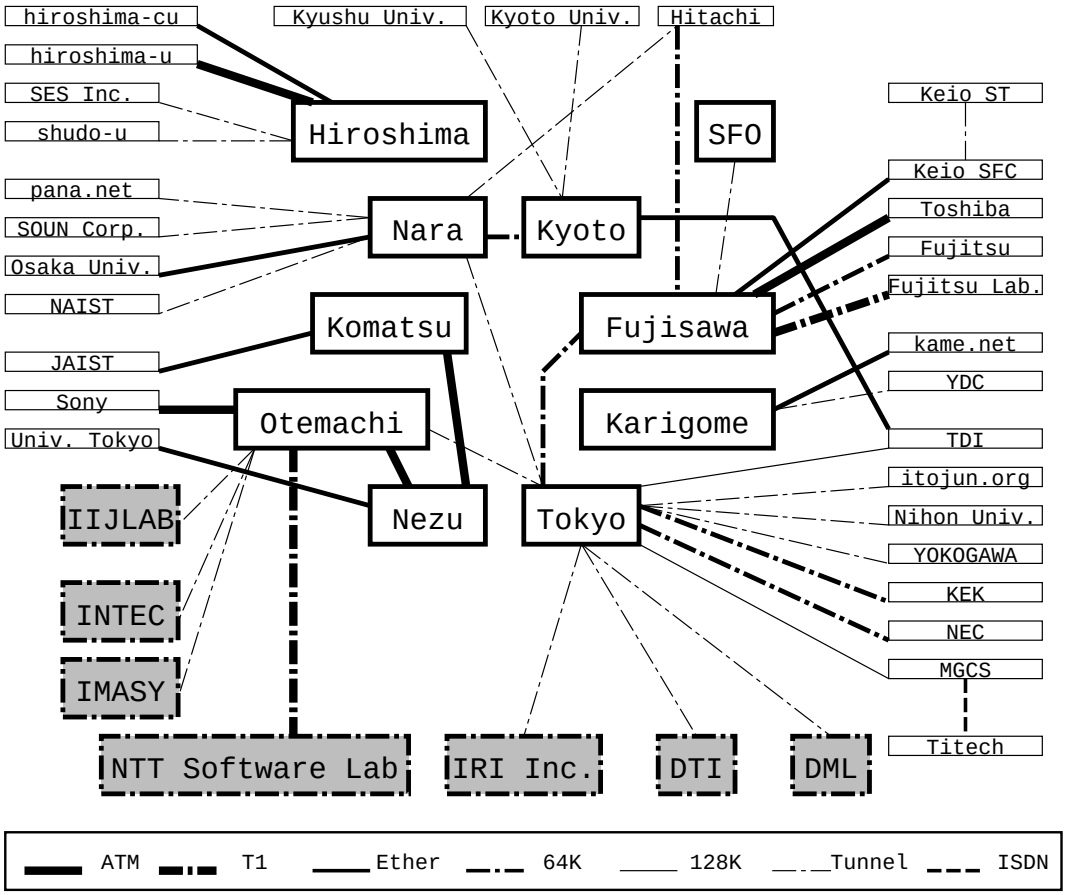

Figure 4.2: A 2D Chaos

The core reason for a three-dimensional visualization lies here. The hierarchy in the IPv6 network was in fact a added dimension to the conventional

IPv4 network. Though the topographical maps of IPv4 could be satisfactorily be expressed in 2 dimensions, IPv6's added dimension of 'depth' implied another dimension of 3D (figure 4.3). A 3 dimensional visualization would allow the user to view either only topography, only hierarchy or both at once, just by the simple means of changing viewing angle.

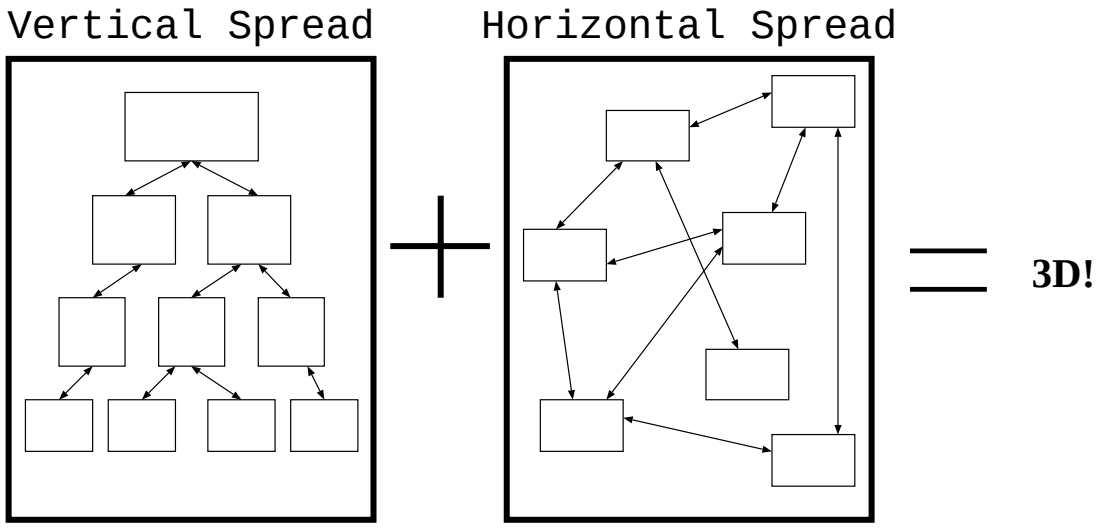

Figure 4.3: Topology + Hierarchy = 3D

It is also arguable that because the address itself contains the hierarchy, one has only to look at the address to grasp the layers. However, though it is true that the address prefix of any given site defines the layer it belongs to, the prefix is not at once apparent when looked at. By physically separating the layers in a 3 dimensional visualization, it becomes at once apparent whether one is referring to a TLA holder, a NLA holder or a SLA holder.

4.4 Requirements for a 3D Visualization

The creation of a valid visualization is a problem which requires decisions in many different areas. First and foremost a three-dimensional interface must be decided on. The actual interface will directly influence the manner in which the actual visualization will be implemented. Another major point of consideration is what is to be displayed in this visualization and how to display it. The kind of information which would be deemed useful to a network manager must be chosen as well as in which manner to display these to the best effect.

4.4.1 The Contents and Considerations

As mentioned before it was necessary to decide the precise details of what exactly is to be shown in the visualization. How to obtain this information once it was deemed necessary was also an important decision.

The proposed visualization was a visualization at the site level as opposed to a visualization at the end node or leaf level. The objective was to represent the various networks at their various layers and as well as the connection between them.

The information needed to implement the visualization was to be found in the registry system found at the various registry points. The data with in the registry system is kept in database accessible by SQL. The pertinent information such as the address prefix, organization name and routing can all be access from the registry system through the use of an SQL query.

To further facilitate the understanding of the network presently being visualized for management, the ability to display traceroute results was included in the implementation. Because the whole implementation has a site

level focus such a display would not help to understand routing within any given site, but it would make any site level network disruptions at once obvious as well as which portions of the network 'tree' are presently inaccessibly.

An interesting point which should be mentioned is that in the future, much of the information presently thought necessary for the implementation may come to be obtained from a future version of DNS. The absolute minimum necessary to create the representation of the IPv6 network hierarchy is the IP address itself and it's network prefix.

A final point to be mentioned here is that the present visualization was designed under the assumption that the network in question was a clean network conforming to IPv6 standards.

4.5 The Actual Visualization

4.5.1 The Visualization Concept

As it is the hierarchy which regulates and organizes the large IPv6 network by ensuring its scalability, it was decided that the primary concept of the visualization be in showing the layered tree structure. The secondary goals were to express the topographical structure and peripheral management data and thus justifying the use of a 3D interface.

4.5.2 The Visualization System

As the goal of this particular visualization was to ensure the visibility of the IPv6 network hierarchy, the two kinds of visualizations which were put into consideration were the Cam/Cone Trees and the Information Cubes mentioned in the previous chapter.

As both were kinds of visualization slated to the display of hierarchical systems, it became a matter of examining the capabilities of each. The Cam/Cone Trees allowed for perhaps a clearer view of the layered structure while the Information Cube would have the added plus of being able to clearly display related data.

In the end the Cam/Cone Tree choice won out, and the decision was not a hard one as it won on merit of its visible layers.The new visualization would be a cone tree, each node would be a site or aggregation point within the IPv6 network. Further more, it was decided that the various would layers would not only be visible by physical separation, but also color coded to enhance distinction. The lines connecting the various sights would represent kind of connection by means of changing line types as well as color and width, much in the same way as it often been done for IPv4 visualizations.

The visualization would be a cone tree opening out toward the bottom. Each node would be represented by a sphere posted with it's organization name, address, and prefix length. Nodes would be connected to each other, according to their actual network connections, by a thin pipe.

The system would consist of 2 main parts and 3 modules. The first part communicates to the registry system to get the raw data for the visualization. The second part processes that raw data to create a VRML script to be read into a browser. A description of the modules will be given in the next chapter.

4.5.3 The Visualization Scenario

It was finally decided that the ideal visualization scenario would go as follows; the data to be made into the visualization would be taken from the registry system data base through the use of an SQL query, on basis of the range of view specified by the user. The results of the query would be dumped into a

file which would in turn be processed by a translation program and turned into a VRML script. The resultant VRML script now has only to be accessed by a browser to be viewed (figure 4.4).

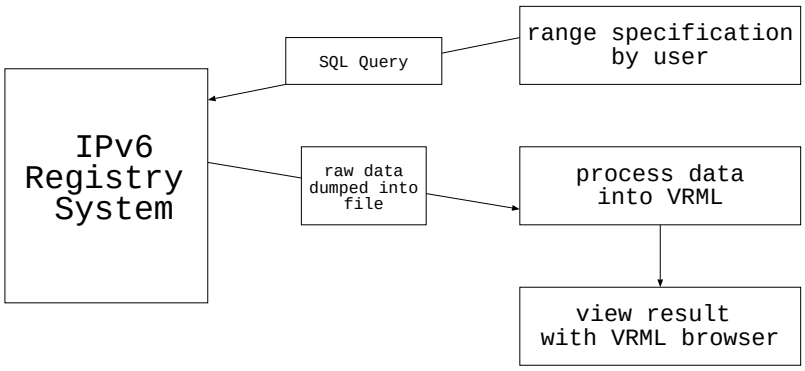

Figure 4.4: The Design Concept

Chapter 5

The Implementation

This chapter describes the actual visualization implementation. First the merits of the chosen interface are explained, and from there, the actual protocol used in visualization, from the manner in which the raw data is obtained to how it is processed and finally visualized.

5.1 The Implementation Environment

The research was done under the 6boneJP and the WIDE 6bone. The 6bone JP, as mentioned in the previous chapter, is an experimental Japan wide IPv6 backbone. The WIDE 6bone, which is run under the 6boneJP, is an IPv6 experimental network maintained by the WIDE Project. The address registration of the two networks are both managed by the WIDE Project. All data for the research done in this paper was taken from the WIDE 6bone Registry.

5.2 The Interface

Several options were considered as an interface for the visualization. The first was to use Virtual Reality Modeling Language, more commonly known as VRML[7]. Another option was to use some different 3D implementation such as Natto View. The final option was to create a completely new 3D interface using Java.

Each one had its merits. VRML was a powerful candidate as an already extremely widely known and used 3D interface. However depending on the manner of realization, some other implementation may have capabilities which would cater more exactly to the needs of the proposed visualization. The option of making a completely new and original interface with Java takes that merit even further as it would be tailor made to suit the needs of the new visualization.

In the end however, it was VRML's already wide spread use which was the deciding factor. VRML was versatile enough to present the needed visualization while already being a widely recognized and heavily used interface. VRML also had the added benefit of being a WEB baseed viewer, which meant that the visualized nodes could be directly hyper-linked to the nodes web site. By using VRML the same visualization could be seen on various different platforms and browsers without compatibility problems.

5.2.1 The Visualization Protocol

The visualization is at its most basic level, a kind of client/server implementation. The registry data is held with the various SQL registry system databases. The user access the data through a SQL query and the thus received data is diverted into a file. The implementation then take the data

from that file and using the information writes out a VRML script which when seen on an appropriate browser shows the visualization.

The process of taking the data in the file and converting it to VRML is done by a simple program, which in this case was written in C. The program takes the data, and after going through the file, calculates the number of layers which are needed, and upon that number how far to space each of the nodes. The nodes on the same hierarchical level are plotted on the same X,Y axes plane and are placed as if on the perimeter of a circle at equal distances from each other. The radius of the circle around which the nodes are placed, grow in size as they go up in the hierarchy, ensuring that there is enough space for all nodes, even at the bottom. The difference in actual layer is expressed by difference in the Z axes value.

By the regulations set when registering for an IPv6 address, eash node is required to provide a topographical map of its site on the WEB. This site is directly linked to the new visualization using the WEB capabilities of VRML. The data processing program therefore also writes in this "link" into the VRML script.

All that is left to do is for the user to start up a browser with a VRML viewing pluggin and load in the newly created VRML script(figure 5.1).

5.2.2 The Different Modules

The implementation consists of 3 major modules shown in table 5.1.

As the table shows, the first module collects raw data from the registry systems dispersed throughout the network. The registry system for the 6bone JP and the WIDE 6bone is maintained in a database accessible through the use of SQL. The module collects the data and dumps it into a preset file.

The second module then takes this file and rearranges the contents into a

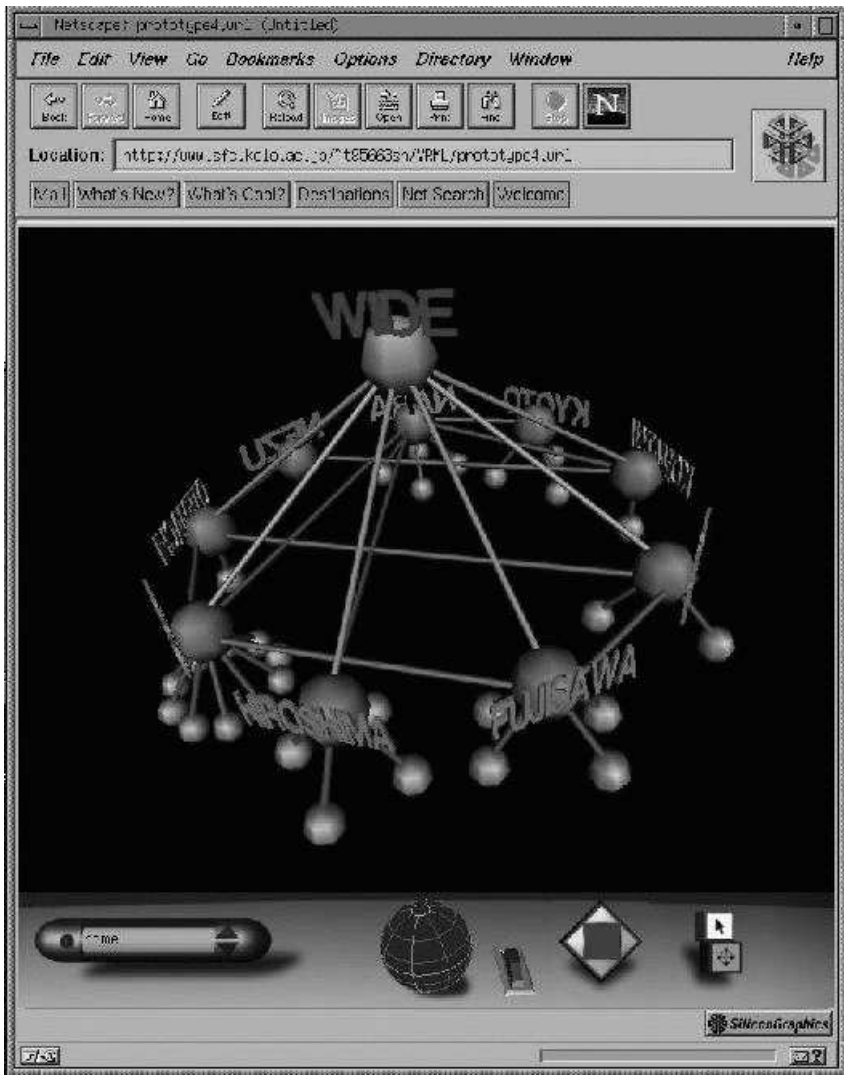

Figure 5.1: A Screen Shot of the Interface

Table 5.1: The Implementation Modules

Name of Module	description of function	medium of implementation
Data Collecting Module	collects raw data from the registry	SQL
Data Organizing Module	rewrites data into processable form	C
Data Processing Module	processes data into VRML form	C

usable form, which is again dumped into a file. The final processing module can now turn the contents into VRML, dumping the information into the final file (figure 5.2). The second and third module are written in C.

Data Collection Module

The Data Collection Module collects the raw data for the visualization and pipes this to a file, in this case called "registrydata.txt". This is done through an SQL query. A basic query produces the name of the organization and its address block for the level which is being asked for. For instance with the WIDE Registry, an SQL query for the information on the NLA1 level will produce 9 entries, all under the WIDE Project TLA.

Data Organizing Module

The data now in the file "registrydata.txt" cannot be used as is. Therefore the Data Organizing Module rewrites the contents to usable form. The SQL query result comes with small emblemishments to make the data easier to see. These include labeling and lines. The Data Organizing Module first removes all these excesses. Then it takes the organization name and address block which are on the same line and separates them, the name on the first

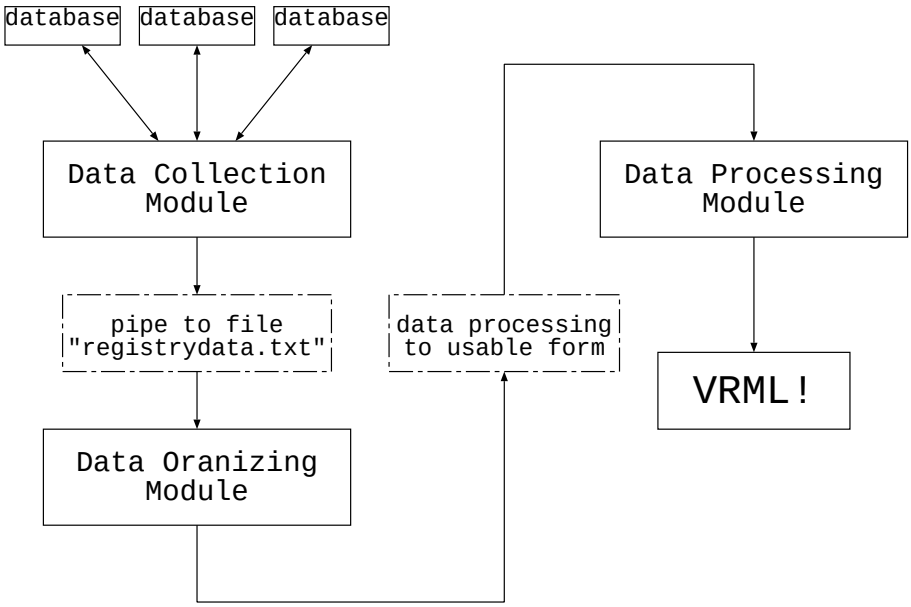

Figure 5.2: The Modules

line, the address block on the second. Each hierarchical level is separated by a signifier, the word "fin".

Data Processing Module

The Data Processing Module now takes this data and turns it into VRML. On basis of the number of occurences of the word "fin" the module knows how many layers to create. The module also goes through a comparison routine to discover which child nodes belong under which parent nodes. As each layer is drawn in a circle, the spacing for this circle can now be automatically calculated, according to the number of child nodes belonging to each parent node.

5.2.3 SQL

SQL stands for Structured Query Language[8] and allows users to access data in relational database management systems, such as Oracle, Sybase, Informix, Microsoft SQL Server, Access, and others, by allowing users to describe the data the user wishes to see.

The SQL used in the WIDE 6bone registry is "ANSI" or standard version of SQL called postgreSQL. SQL allows users to specify which factors of what database table he or she wishes to see. The SQL query in the implementation askes specifically for the organization name, IPv6 address block, and url of topology map from the registry database using the "SELECT" and "FROM" command.

Chapter 6

Evaluation

This chapter evaluates the implementation described in the previous chapter, and then offers some points of refinement based on this evaluation.

6.1 The Objective and Its Fulfillment

The underlying objective of the visualization lay in helping make the potentially enormous IPv6 network manageable. To this end the visualization's first goal was to enable the display of horizontal topographical data of the network or its 'spread' while at the same time managing the vertical hierarchical data or its 'depth'.

The implementation discussed here attempted to do this by the physical separation of the layers. The hierarchical structure was further emphasized by color coordination.

In this aspect the visualization was a success. As the cone tree type of visualization is a direct three-dimensional extension of the now familiar 2D hierarchical tree structure, the network layers were indeed beautifully apparent. As can be seen in the comparison between the two visualizations

in figure 6.1 , even when compared to an implementation of an Information Cube, the other hierarchical data expression type explained before, the cone tree approach was better in respects that the layers where evident as such, at all levels within the present scope of view.

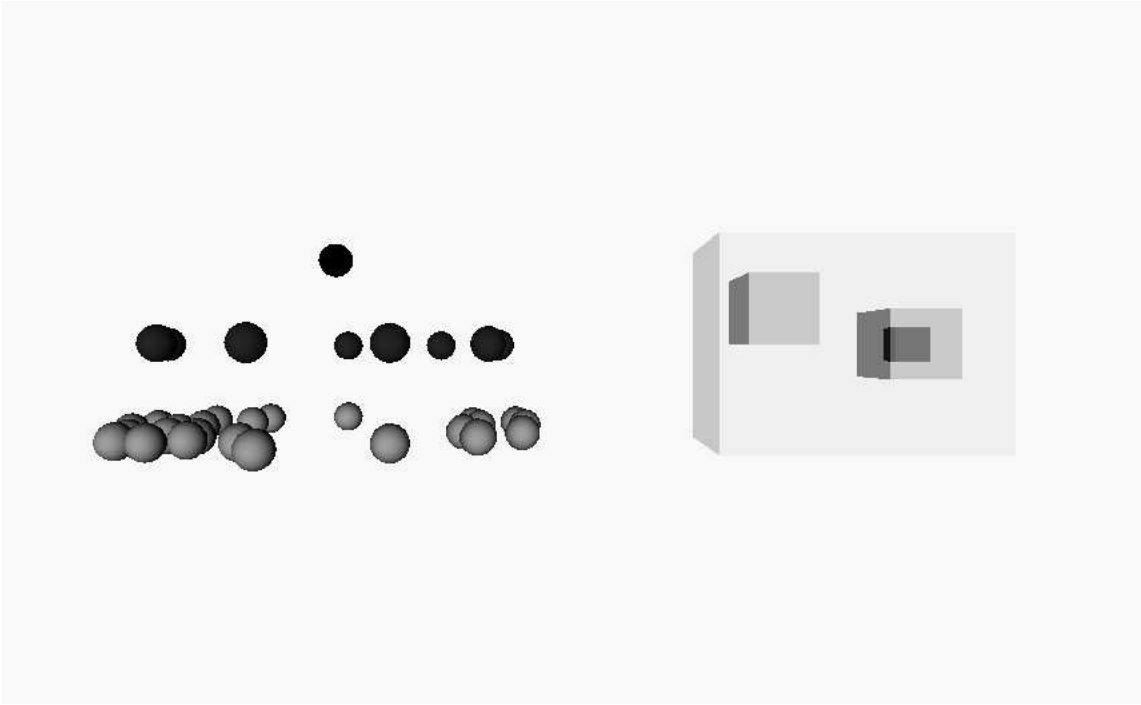

Figure 6.1: A Cone Tree vs An Information Cube

However due to the characteristic 'cone' created by the connection lines between the layers, the topographical data becomes difficult to view. In this aspect the visualization did not quite come up to the standard as expected and some remedial action must be considered. The obvious answer is to remove the connecting lines and consider another means of showing vertical connection such as color coding.

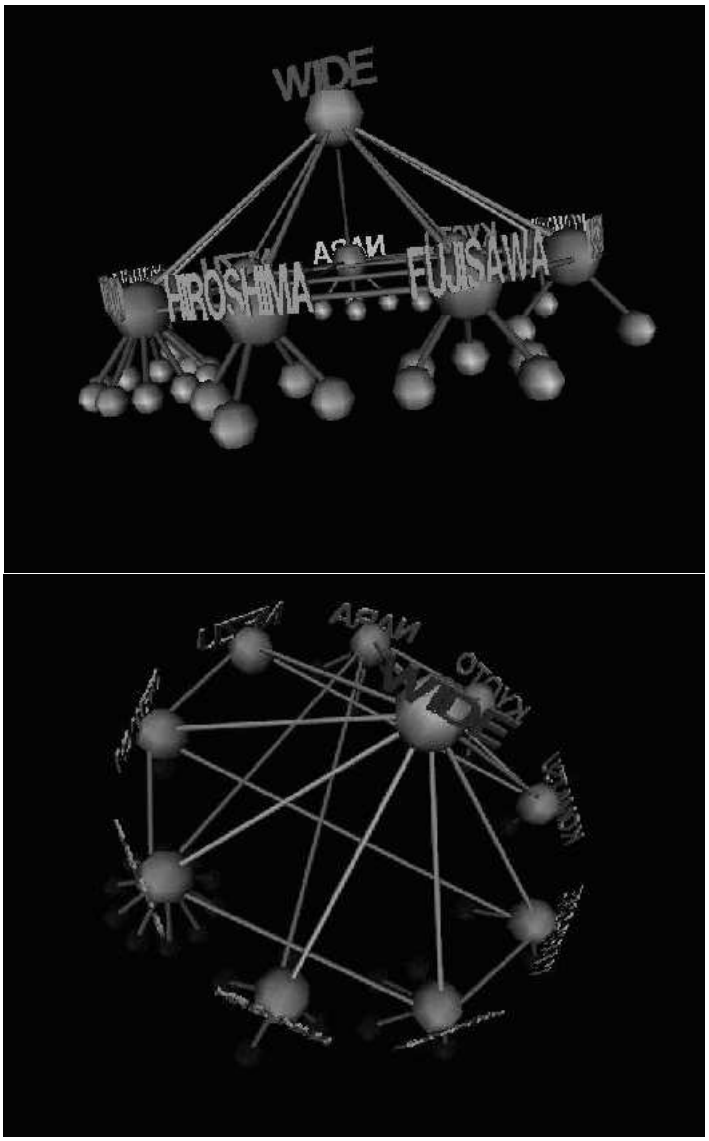

Figure 6.2: A Prototype of The WIDE 6bone In 3D

Another comparison to be made is the difference between figure 4.2 and figure 6.2. The virtually non-existent hierarchical layers in figure 4.2 are apparent in figure 6.2, which at the same time does not convey quite the chaotic hassle of the flat visualization.

The visualization also links to each node's site topology map, and can be made to display the address and/or address prefix. This allows network managers to obtain more information from any one single source than was ever previously possible.

6.2 Problems in Implementation and Future Refinements

6.2.1 Problems

There were many problems which came up during the implementation of the IPv6 visualization. Many were simply hard decisions to make. While others were limitations imposed by the tools being used to create the visualization.

The manner in which to plot the various nodes was one of these. Though the visualization concentrated on the hierarchical aspect of the IPv6 network, the merit of creating a three-dimensional visualization was that both topological data and hierarchical data could be viewed the same time. The difference hierarchical layer was easily expressed by giving different layers on different values on the Z axes. However, how to incorporate the topographical data at the same time was something to consider.

The difficulty in plotting also had much to do with the limitation of the interface. VRML was chosen as the interface for its already wide spread use, and not necessarily because it offered the optimum capabilities. Experimentation proved that though in many ways VRML is indeed a versatile language

in which to create three-dimensional images, it was extremely difficult map out irregular shapes.

Another problem was the way in which to handle the scaling of the visualization. Of course VRML itself provides the ability to zoom in and out as well as to focus on certain points. However this is within the scope of a already created and displayed visualization. If a user after seeing the whole network, now wished to have only a portion of it displayed, then he/she would be required to specify as such to the SQL server in another query, run the converter program and create a new VRML script. In other words the inability to dynamically modify a up and running visualization was a problem.

Simple scope was also a problem when considering the scale of the visualization. In the network visualization, the depth of the the hierarchy did not seem to be much of a problem. The numbers of layers are to a certain extent limited and one can always scroll down. However, because of the tree structure used and displayed in this type of visualization, the lower layers would end up become tremendously wide. So wide in fact as to exceed the limits of the screen when seen up close.

The last major problem lay in the decision of how to handle multihomed sites. These sites that at once have a single identity at 2 or more levels should be expressed as such.

6.2.2 Future Refinements

From the evaluation above, the first point of refinement would of course be clarification of the network topology view. The topographical view can be greatly improved in two aspects. First the horizontal view is greatly interrupted by the diagnal lines connecting the adjacent layers. Though the

cone tree visualization displays the hierarchy admirably and should not be changed, perhaps it should be modified to showing hierarchical connection in a different manner. Secondly the manner in which to plot the topological information itself can be considered as a small field of research on its very own.

As the research progressed, it became more and more apparent that this sort of multi-faceted visualization could prove to be extremely useful to network and address management. The fact that hierarchical layers were now apparent meant that the address structure itself was evident. This factor could become useful in situations such as address renumbering. An obvious refinement fo the visualization would then be to create an interface where address renumbering could be done directly through this 3D interface. This would of course add more reasons to change the interface from simple VRML to something with more dynamic and interactive capabilities.

The new interface should also enable more filtration techniques as well. In the present implementation the filtration was done by the first module. This means that any time some new aspect is required the whole visualization process must be done over again. If however the interface could be made to incorporate such a filtration mechanism, the visualization could then be created with all the possible information from the beginning. The interface would make sure that the user sees only what he/she wishes to see, with extra and confusing data items.

Chapter 7

Conclusion

7.1 Summary

In a world where the spread of the Internet grows at exponential speeds, the lack of address space is a dire problem. As the world rushes to solve this problem through the creating of IPv6, many problems surface. One such problem is the management of address assignment and as well as of the network itself. The two jobs are made difficult by the sheer size of the network. Another problem is the fact that present day IPv6 experimental networks are managed in large part by IPv4 network managers. This means that the IPv6 network is tied to the concepts of IPv4.

However, IPv6 is not simply a version of IPv4 with a mammoth address space. It has characteristics all of its own, which require a completely new outlook. This new outlook is provided by the visualization create in research stated here.

People understand things by the way they picture them. Because of this visualizations have a great effect on peoples understanding of things. With

only conventional IPv4 visualizations available for the IPv6 network, IPv6 will never amount to more than a jacked up version of IPv4. Instead it must become a whole new protocol, a new world unto itself, something to completely replace IPv4. IPv6 unique visualizations will help IPv6 to attain this status.

The research stated in this paper explore the differences in the two versions of IP and strive to create a visualization which reflects these differences.

7.2 Conclusion

In conclusion, the results of the visualization can be said to be a success. The hitherto hard to distinguish hierarchy was indeed well marked out. Compared to the conventional flat layout, the new 3D visualization allows for a true understanding of the new IPv6 network and its structure. The manner of address assignment and network management information are at once easily understood and readily available.

Unfortunately with the topographical side still left as a project for further improvement, the results cannot be called an complete success. However that the present state of visualization in IPv6 management was sadly lacking, is an indisputable fact. The road to understanding IPv6 must begin somewhere. The visualization presented here in this research has shown ample results to pave the way for clearer, more useful visualizations and therefore understanding of the superstructure that is an IPv6 network.

With this new visualization, the IPv6 network becomes a sphere of space which can be easily navigated through, instead of the daunting jungle it could easily become. Address assignment and network maintenance will become more manageable and will continue to be manageable even in the face of an

ever expanding IPv6 network.

Acknowledgments

I would like to now take this opportunity to thank Professors Hide Tokuda, Jun Murai, Hiroyuki Kusumoto, and Osamu Nakamura of the Faculty of Environment and Information for their supervision into this research.

Special thanks also to Mr Yuji Sekiya, Mr Yusuke Doi and Mz Shoko Miyagawa for their unstinting help in my research and writing this paper. Without their advice in implementation and evaulation as well as their encouragement, I would never have made it this far.

A word of thanks also for the members of the IPng WG here at Keio University Tokuda, Murai, Kusumoto, Nakamura Lab. Their help and support was invaluable.

A final word of thanks to all friends and family members for your kind words of support. Life would have been intolerable without you guys.

Bibliography

[1] K. Hubbard, M. Kosters, D. Conard, D. Karrenberg, J. Postel, "Internet Registry IP Allocation Guidelines" RFC 2050, November 1996.

[2] Christian Huitema, " The New Internet Protocol, Second Edition" Prentice Hall; 11/97; ISBN: 0138505055

[3] R. Hinden, M. O'Dell, S. Deering, "An IPv6 Aggregatable Global Unicast Address Format" RFC 2374, July 1998.

[4] S. Williamson, M. Kosters, D. Blacka, J. Singh, K. Zeilstra, "Referral Whois (RWhois) Protocol V1.5" RFC 2167, June 1997.

[5] Peter Young, "Three Dimensional Information Visualisation" Visualisation Research Group Centre for Software Maintenance Department of Computer Science University of Durham, http://www.dur.ac.uk/ dcs3py/pages/work/documents/lit-survey/IV-Survey/index.html, March 1996.

[6] R. Hinden, "Proposed TLA and NLA Assignment Rules" RFC 2450, December 1998.

[7] 河合朝雄, 河西雄一, "VRML 入門" 技術評論社;7/97; ISBN: 4774104655

[8] James Hoffman, "Introduction to Structured Query Language" http://w3.one.net/ jhoffman/sqltut.htm, 1996.