Proc. SPIE Workshop on Commercial Applications for High-Performance Computing,
Howard Jay Siegel, Editor, Vol. 4528, pp 138-146, Denver, Colorado,21-22 August 2001.

Creating a Semantic-web Interface with Virtual Reality

David Cleary[a], Diarmuid O'Donoghue[b]

[a]Ericsson Applied Research Labs. Athlone, Ireland.

[b]Department of Computer Science, NUI Maynooth, Ireland.

ABSTRACT

Novel initiatives amongst the Internet community such as Internet2 [1] and Qbone [2] are based on the use of high bandwidth and powerful computers. However the experience amongst the majority of Internet users is light-years from these emerging technologies. We describe the construction of a distributed high performance search engine, utilizing advanced threading techniques on a diskless Linux cluster. The resulting Virtual Reality scene is passed to a standard client machine for viewing. This search engine bridges the gap between the Internet of today, and the Internet of the future.

Keywords: Internet Searching, High Performance VRML, Visualization.

1. INTRODUCTION

Both web-pages and search queries are represented in natural language, and thus suffer from the ambiguities of natural language. We describe different categories of lexical ambiguity: *homonyms, synonyms, local* and *global syntactic ambiguity, etc* before focusing on *acronyms* and *polysemous* queries. We describe how web-page clusters or attractors [3] are identified, and how individual clusters represent different interpretations of the given query. We describe a variety of example queries, where individual clusters each identify different interpretations of the query. We contend that producing lists of seemingly independent web pages like google [4] are fundamentally unsuited to such queries - regardless of how good the results are.

Rather than grouping pages according to categories of content information (like NothernLight) - we focus on page connectivity, as semantically related clusters frequently share a common referring page. Virtual Reality is the ideal medium to properly represent the connectivity found among typical search results [5]. We describe the VR-Net application that identifies and displays clusters of information using existing search engines. VR-Net collects results from search engines, identifying collections of inter related information in a site-centric manner. VR-Net generates a desktop virtual reality (VR) interface to this information using VRML 2.0, for use on VRML enabled internet browsers like Netscape and Internet Explorer. Explicitly depicting each information cluster allows the user to quickly focus on the required interpretation.

2. LEXICAL AMBIGUITY

Natural language is inherently ambiguous [6], with lexico-semantic ambiguity proving the greatest obstacle to information retrieval on the Web. Among the ambiguity problems encountered by web users are: *homonyms* words with the same spelling but different meanings (pontoon boat or the pontoon card-game), *synonyms* (car, automobile), *local syntactic ambiguity* ("I saw her duck") and *global syntactic ambiguity*, ("We pained the wall with cracks."). *Acronyms* are frequently ambiguous ("AA" refers to automobile association and also alcoholics anonymous. For example, suppose we wish to find "Mrs. Cook", this ambiguous term might refer to a number of semantically distinct concepts: "cooks, cooking, the Cook Islands and so forth" [7]. This ambiguity pervades natural language; with terms like "fish" referring variously to all aquatic creatures, the activity of catching fish, fishing for information etc. By submitting a query like "cook" to find the required Mrs Cook to a search engine (even the wonderful google) we can see that lexical ambiguity is responsible for the erroneous results returned. Most search engines return results in a seemingly arbitrary order, blissfully unaware of the confusion that results.

We use existing search engines to identify a seed set of pages for any given query. All identified pages are examined and the destination of all retrieved hyperlinks is compared. All pages pointing to a common target site collectively form

a web-page cluster. In this section we briefly compare some results, indicating how this strategy forms clusters isolating different senses of search queries.

a) Acronyms
The use of acronyms abounds on the Internet, ranging from Internet related Acronyms like W3C and ISOC, to AA and SPIE. Many of these acronyms are ambiguous, with each sense of the term referring to different semantic concepts. We illustrate the use of VR-Net to identify an acronym, and shall see that VR-Net identifies clusters based around different senses of the acronym - identifying some very different sources.

Consider the acronym "AA" and the semantic senses that are identified by the VR-Net application - in decreasing order of (ranked) relevance.

- `www.aas.org` - The American Astronomical Society
- `www.alcoholics-anonymous.org` - Alcoholics Anonymous
- `www.aa.org` - Alcoholics Anonymous (mirror site)
- `www.aagrapevine.org` - The International journal of Alcoholics Anonymous
- …
- `www.theaa.com` - The Automobile Association
- `www.aahistory.com` - Unofficial Alcoholics Anonymous site

The "American Astronomy" cluster identifies an isolated community of web-pages that is not connect to any other retrieved pages - for this query. There is quite a degree of connectivity between many alcoholics anonymous pages, with www.alcoholics-anonymous.org and aa.org being mirror sites. The "Automobile Association" cluster again identifies a semantically isolated cluster of web pages. The aahistory.com site even contains a statement disassociating itself from Alcoholics Anonymous and from The Automobile Association.

b) Polysemy
Another category of ambiguity concerns a single lexical form used to reference different semantics. Consider the following results for the search term "bass".

- `www.pfeiffer.com` - Jossey-Bass/Pfeiffer - A Wiley Company
- `www.wmi.org` - WebMasters International, Inc. (links to BassFishing)
- `www.josseybass.com` - - A Wiley Company
- `www.bassplayer.com` - Music Magazine
- …
- `www.bassangler.com` - Fishing information
- `www.bassfishin.com` - more fishing

Each of these clusters identifies a different sense of the query "bass". Even `www.wmi.org` includes a variety of links to bass-fishing sites and other

As can be seen from these results, ambiguous terms seem to form clusters of semantically related and inter-linked pages. Identifying these clusters goes a long way to eliminating the inherent ambiguity, and allows the user to focus on the relevant cluster. Thus, rather than trawling through a seemingly arbitrary sequence of pages, clusters form more focused interpretations of the original query. This dynamic process is not subject to the limitation of some pre-defined semantic ontology (like a thesaurus), but arises naturally from an examination of the interconnectedness of search results.

3. VR-NET SEARCHING FOR A SEMANTIC WEB

As we have seen from the current problems facing users of the Internet with the current stagnation of browser interface technologies, a different approach is need to address the fundamentals of users interaction with the web.
VR-Net is built around three-tier client server architecture. The first tier commences by identifying a set of semantically related web pages associated with a user query. This is achieved by leveraging the existing power of web-indexing engines, combining them with a more advance ranking approach [8]. Within the second tier of VR-Net the information needed to depict a visually rich environment is stored and consolidated. The top layer of our architecture is where our

information model is transformed into a virtual reality scene realizing the visualization metaphor of a shopping mall in VRML 2.0 [9]. The architecture of VR-Net is illustrated in the following diagram:

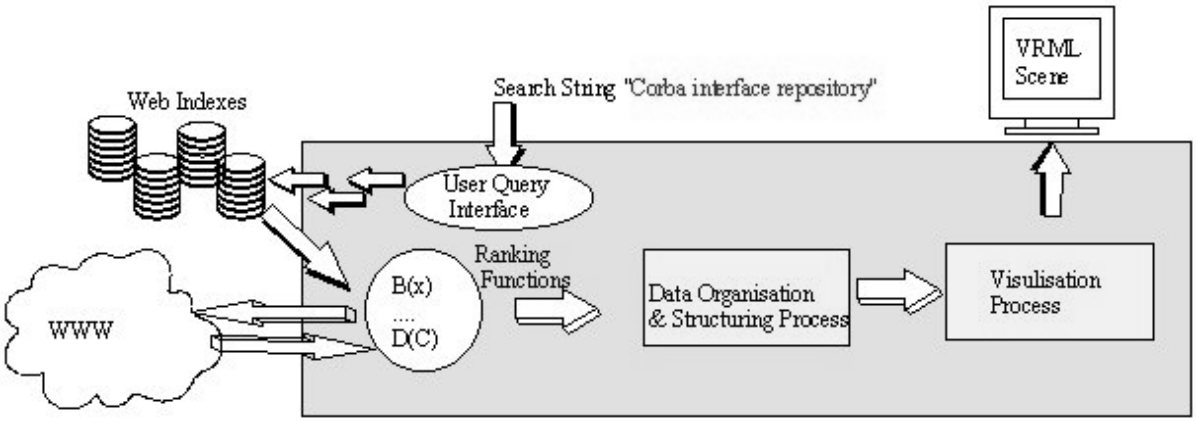

Figure 1 - VR-Net architecture overview.

As can be seen from the above flow diagram moving from left to right, after a query has been entered in to the system three conceptual stages are passed through. A coarsely grained pipeline algorithm is used to move the newly forming semantic information between each of the computational tiers. However figure 1 only illustrates the conceptual flow of information through the system. To fully understand the software architecture we must examine a running system load balanced over a number of servers.

To create a truly high performance application needed to realize figure 2 architecture, one of two options are available to the software developer. The first option is to run your application on dedicated high performance ‘super’ computer, designing the computing algorithm such as to optimize the underling hardware architecture. The second option is to create a distributed application that manipulates the collective power for multiple machines in a cluster configuration.

Within VR-Net a hybrid approach is adopted to create a computing platform that provides the required computation power to create a search engine that resolves the issue discussed in the previous section. The architectural approach is based on a n-tier distributed module utilizing Java RMI [10] as the underline object oriented middle-ware. The physical architecture that VR-Net runs on is a set of diskless Linux servers connected over a 100 Mbit Ethernet network, using a distributed file system NFS [11]. A central control server manages this hardware, whose topology is shown in figure 2.

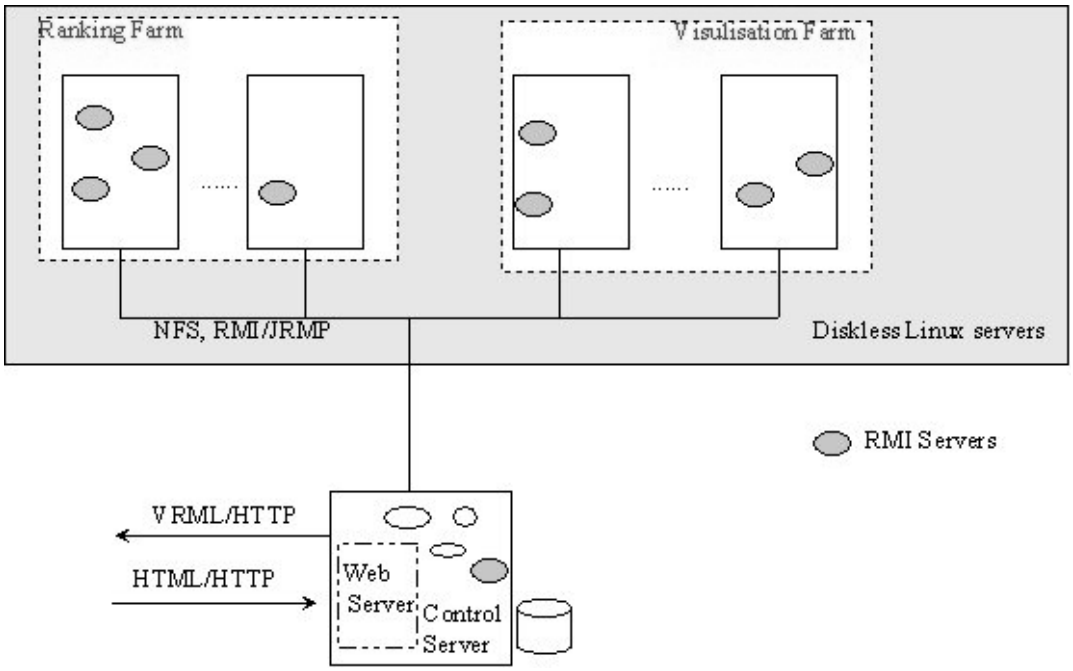

Figure 2 - VR-Net hardware configuration

VR-Net system is divided into to two centrally controlled server farms. These server farms are logical division of two types of activities:

1) ***Ranking Servers***; These are servers that take a users query and begin to build a semantic information map.
2) ***Visualization Servers***; these are server that perform a lossy-cross compilation for the internal semantic information model and generate a VRML scene.

The system has on key central server that has a network attached storage space. This Control Server is a unique machine with in the system. It is used to boot the diskless clients and manage the distribute synchronization and load balancing between the *Ranking Server's* and *Visualization Server's*. The Control Server has two key functions.

1) The Control server allows for dynamic configuration of the diskless Linux servers into two server farms one for the ranking engines and the second for the cross compilers. The Interaction between these two server farms is controlled by the cluster manager which handles the load algorithms for the two farms. The size of the two farms various with the system load. Machines can be added or removed to the collective farm without stopping on going queries. The control server communicates with each member of the cluster in a peer-2-peer fashion over an RMI control plain. The control interface which each server must implement is based on an asynchronous message communication the RMI interface utilizes a distributed event mechanism similar to JINI event model [12]. This allows for the complex control of given query passing from the ranking stage through to the VRML scene generation.
2) The second key function that the control server provides is the termination of the users query request. The query request is over HTTP and is terminated in a Java servlet controlled by apache tomcat servlet engine. On completion of users query being processed it returns the VRML scene graphical depicting the semantic web associated with the query.

Now that we have established the computing principles of our system we will examine each of the key functional areas.

4. SEMANTIC INFROMATION FORMATION

To achieve the most semantically correct information need to realize our visualization metaphor our information retrieval stage is comprised of a number of ranking and retrieval algorithms. The key to understanding the underling

ranking and data retrieval tasks within VR-Net is centered on the key concept of categorizing data within the web. Our view on the importance of web sites is based on the weight of reference within a given page to another of similar topic; this is similar to the mechanism used to rank the importance of academic publication through citation reference. This approach is along the same track that was adopted Clevers' authorities and hubs [11]. However with the primarily difference been that we only allow Attractor members [12] to be spread about a web-site while authorities are viewed as individual pages located anywhere within the web. Within our ranking and information retrieval process unlike existing work in web ranking algorithms that link structure plus anchor text to identify useful web pages [13] we do not differentiate between distinguish between anchor and other text. Clever algorithm is built on an iterative process to rank pages, where as our approach is based on a sequential series of linear ranking functions.

The first stage identifies candidate information sources, while the second identifies attractors from this data, and the final stage organises and collects satellite nodes. To fully understand the underlying parts needed to build the information model, we will examine each step in detail.

- The first stage begins by retrieving a seed set of URLs from a number of web indexes, for some given search topic.

- Attractor ranking Stage: Within this stage of the algorithm our key information groupings are identified using an attractor ranking functions based on the concepts that the URLs themselves contain valuable information in relation to the web site that they are pointing to.

- Satellite Organization Stage: The final stage is used to complete our collection of satellite nodes. These pages were initially discarded by the threshold function as part of the initial ranking functions, but recovered due to their proximity to attractors from stage one.

The internals of the ranking functions center on the understanding of URLs. To aid comparison of URLs and to allow a useful structure to be determined, we apply a series of analysis stages to each URL. The ranking algorithms are run in a multi-threaded Java [14] server '*Ranking Server'*. The ranking server, offer a polymorphic container and data retrieval process that allow information be retrieved from various web indexes. Multiple of these containers are controlled from a single control server over an asynchronous RMI interface. Internally the key data that requires synchronization between threads is stored in a signal *volatile* Java data structure allowing for efficient low level synchronization for the main data structure be carried out by the JVM. The use of this technique reduces the need for synchronized regions of code.

5. DATA ORGANIZATION AND STRUCTURING

After the completion of the web extraction and data ranking process a primitive data structure is generated who's primary focus is in ordering clusters of semantically related pages established from the user specified search. This data structure is not rich enough to generate the 3D visualization metaphor in VRML. The main objective of this processing stage is to provide an interface for the visualization activities of the presentation level. To achieve a clean interface for the visualization process a number of activities are done within this processing layer. These can summaries as follows.

- Normalization of the initial information model to remove unwanted data not needed within the visualization process.
- Transformation of the information from its raw state in to a structure that is easily manipulated by our lossy cross-compiling VRML engine. Within in this process a relationship meta-model is also derived. This meta-model is central to the organization of information within the VRML scene and also is used to aid navigation thought the complex object orientated containment model.
- Graphical rendering and storage of thumbnail images to added extra visual information to our virtual environment. Conceptual on can be though of as rendering a given HTML page in its WYSIWYG structure and then converting the contents to an image format. This image visualizes the HTML page, as it would be seen in a traditional browser. A size-reducing algorithm is then applied to the image so it can be easily textured mapped on to a VRML object.

To achieve a fully flexible architecture the information is finally stored in a polymorphic data container. The reason for not tailoring the container is so a variety of web entities can be stored in it allowing for the expansion of the data model used in the visualization stage. Contained within the information model is also a site centric meta-model of the relationship of the web data. This generic information repository emits location transparency features allowing it be

transferred clustered machines. This is realized by serialization the information model for data transfer. To make the transfer effective we apply a custom serialization algorithm derived especially for the data structure. This data model is then transferred over a low level socket interface for greater bandwidth to one of the visualization process. This broadband interfaced is controlled by an administration asynchronous RMI interface. This allows both end of the connection pipe be controlled for the control server.

6. VISULIZATION PROCESS

After the information model creation is completed the visualization process begins. Various techniques exist for visually understanding large information spaces [15], [16]. The most realistic way of understanding abstract representations is to map them to the physical world. Within VR-Net we adopt the concept of abstract graphic objects to provide the foundation for our visualization, satisfying key design parameters, completeness, extensibility, hierarchy, parameterization and abstraction [17]. These techniques allow our VRML world contain all the required pages plus their inter-relationships, the users performs all browsing within a topicalized shopping environment, which serves to structure their browsing activities. All "on topic" pages are located within this world, so the user need (almost) never follow a hyperlink from a given page. Users are freed from the disorientation associated with conventional browsing, as the relationships between pages are clearly visible. Even pages that reference each other can be visited through the "virtual world" interface. The VRML scene also facilities the direct interaction with the underlying HTML pages when required.

The VRML scene is created using a number of cross-compilation techniques. As well as the predefined navigation tools provided by VRML browsers, the visualization process also creates a set of navigation routines within the virtual environment to add user interaction. The generated VRML relies on three main features of language:

- Prototypes: This VRML construct allows for complex VRML object from primitive shapes (box, cone, cylinder, line and sphere). Each prototype can accept input parameter to allow runtime behavior of the object be shaped as user interacts with the virtual environment.
- Inline files: Allow the VRML scene be defined in numerous files across multiple files, dividing complex worlds into smaller unitized files and also reducing the initial footprint of the scene for been loaded until it is require. VR-Net adopts a user event based mechanism to load element of the scene on the fly as required.
- LOD (Level of Detail): Allow alternate representations of one virtual object, with the greater detail being displayed when the user approaches the object. This is triggered by proximity sensors within the virtual world detecting the users approach to a VRML group node.
- Anchor nodes: anchors associate a URL with an object in a world. Anchors allow the VR-Net user to browse through from the virtual world and interact directly with the underlying web-pages.

The generation of the VRML world is achieved by a combination of static composed VRML and Dynamic generated code. The major actors and use cases are depicted in the following UML Use Case diagram.

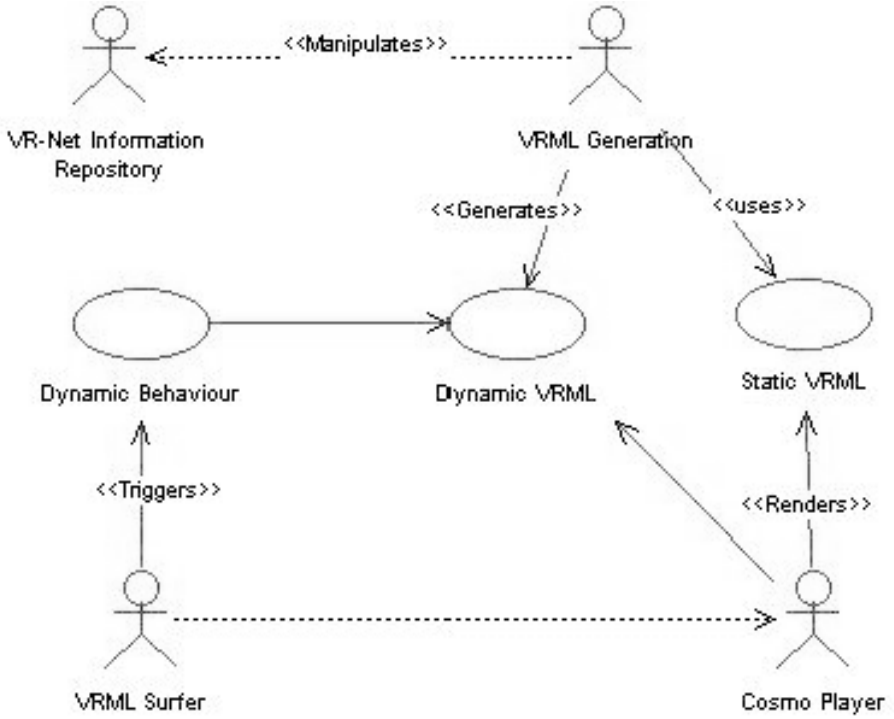

Figure 3 - Visualization process Use Case model

As we can seen the predominate interaction with the lossy-compilation is the dynamic VRML creation achieve by iterating through the generic information model that was populated in the lower two tiers of VR-Net. The VRML surfers interaction with the VRML scene is used as dynamic input allowing the virtual scene emit real life characteristics of information leveling depending on the physical location of the user relative to the shop been viewed at a given instance in time.
Within the VR-Net generated VRML shopping mall scene there are a number of navigation cues that are added to aid the understand and navigation of the visual representation of the users initial search. These navigational aids can be summarized into two key categories differentiated by their interaction with the virtual surfer. *Static behavior:* as well as the standard three dimensional navigational tool offer by standard VRML browsers VR-Net also provides a set of angle view points to the user. These viewpoints offer summary and detail views of products and shop within the mall as well as an overview aerial and front on map views of the entire scene. *Dynamic behavior:* as the user navigates through the mall extra information is added to the scene triggered by the users proximity to a given shop. This allows the scene to contain a large amount of information but still does not overwhelm the user.

7. VISUALIZATION COMPONENT

Consider the seemingly detailed query "AA". Rather than identifying a single page VR-Net identifies attractors based around the following web-sites, with each cluster identifying a different aspect of the "AA" query. The American Astronomical Society - `www.aas.org`; Alcoholics Anonymous `www.alcoholics-anonymous.org` -; Alcoholics Anonymous (mirror site) - `www.aa.org`; The International journal of Alcoholics Anonymous - `www.aagrapevine.org`; The Automobile Association - `www.theaa.com`. All relevant sources of information are displayed in the world, and their relations are clear by their spatial locations. Attractor members are the golden multi-sided figures next to the attractors, while the spheres stretched across the top of the Attractor nodes represents the other pages who are also information suppliers. It seems almost ludicrous to think that any one page could address the needs of all users who initiated this query particular with the query constitutes an ambiguous search term.

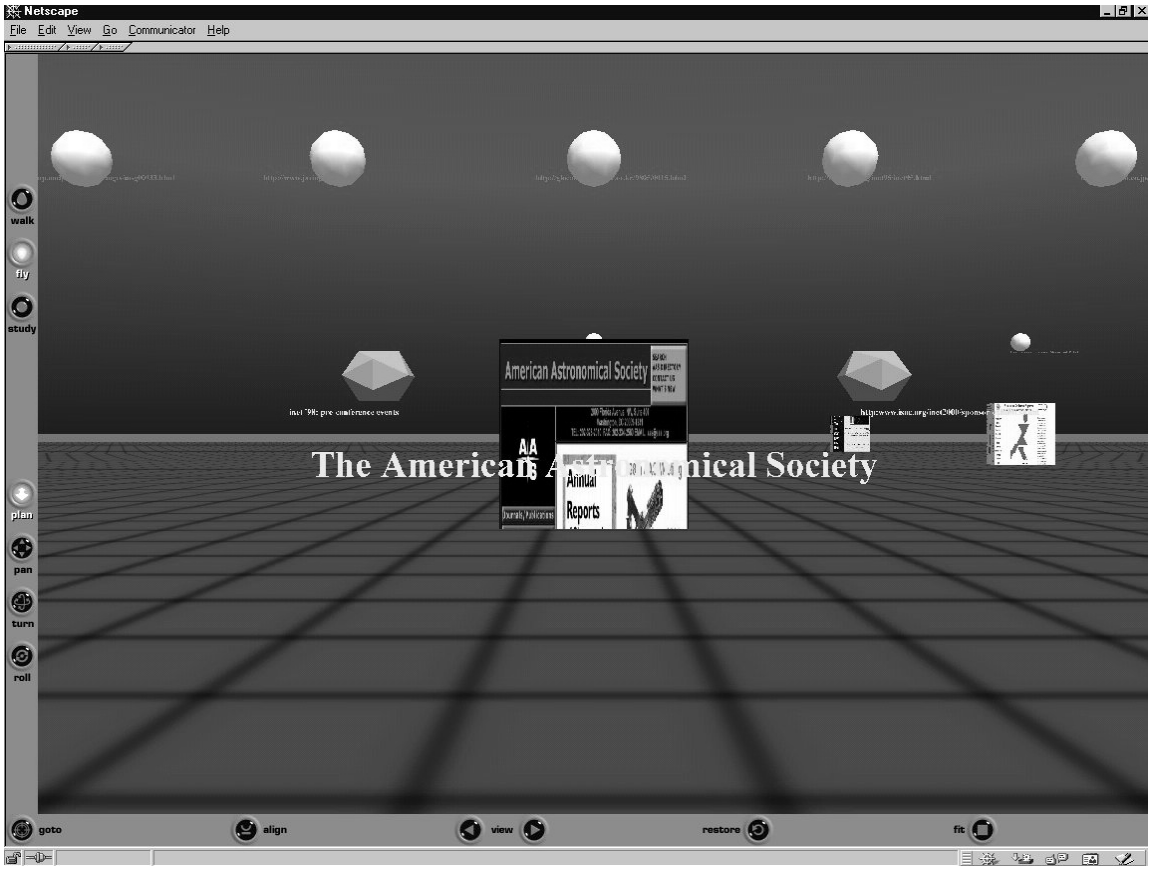

Figure 4 - One VR-Net cluster for the search term "AA"

- Thus the browsing difficulties described are immediately eliminated. Visiting any identified page merely involves clicking on the relevant icon.
- Browse-through can be supported within a separate window (or in same window).
- Structural relations are explicitly displayed, to the referencing structure used by each author is immediately visible.
- Thumbnail previews reinforce recognition of pages within the world.
- Move about the world to any of its semantic neighbor-hoods . VR-Net supports predefined viewpoints that focus upon each shop, including associated products etc.
- The shopping mall metaphor eliminates the potential difficulties in dealing with a desktop virtual environment - with only forward and backward directions necessary to navigate the world.

VR-Net supports a new level of web interaction, by directly addressing some of the prevalent difficulties experienced by users. A cognitively aware approach yields an information environment that is familiar to users. Eliminating the cognitive overhead of traditional browsers allows users to focus the semantics of their query, without distraction. Because shops form around common themes, after sampling one product from a shop the user can make an informed decision about the potential usefulness of the remaining products without actually examining these products. VR-Net empowers users to reason about information clusters, rather than iteratively examining individual pages in turn.

8. CONCLUSION

In this paper we have highlighted how natural languages inherent ambiguity contributes to Internet searching problems. Emphasising how, by harnessing the power of distributed systems it is possible to create a high performance search engine capable of resolving some of the ambiguity that is normally faced by user of other search engines. We described

how this search engines search for clusters of related information, clarifying the semantic meaning(s) of the user-specified subset of the Internet. VR-Net reduces the complexity associated with searching the Internet by presenting each interpretation of an ambiguous query to the user as separate regions of a virtual world. This enriched interface can immerse a user in a visually enhanced virtual where the semantic ambiguity is resolved, leaving the user free to operate in the semantic environment of their original search problem.

9. REFERENCES

1. Internet2 Web Homepage: http://www.internet2.edu
2. K. Almeroth, "The evolution of multicast: From the MBone to inter-domain multicast to Internet2 deployment," IEEE Network, January /February 2000
3. Chakrabarti, S. Dom, B. Kumar, S.R. Raghavan, Rajagopalan, S. P. Tomkins, A. Gibson, D. Kleinberg, J. "Mining the Webs Link Structure", IEEE Computer, June, 1999 - 1.
4. Page, L. Brin, S. "The Anatomy of a Large-Scale Hypertextual Web Search Engine," 7th International World Wide Web Conference, Brisbane, Australia, 14-18 April, 1998.
5. Terveen, L. Hill W. and Amento, B., "Constructing, organizing, and visualizing collections of topically related Web resources", ACM Transactions on Computer-Human Interaction, Volume 6, 67-94, 1999.
6. Jurafsky, Dan and Martin, James "Speech and Language Processing", Prentice-Hall, 2000
7. Berners-Lee Tim, Hendler James and Lassila Ora, "The Semantic Web", Scientific American, May, 2001.
8. Cleary D., O'Donoghue, D. "VisualExpresso: Generating a Virtual Reality Internet", in P. Sloot, M. Bubak, A. Hoekstra, and B. Hertzberger (Ed.) High Performance Computing and Networking HPCN-99: Virtual Reality
9. VRML Consortium, "The Virtual Reality Modelling Language" ISO/IEC 14772-1:1997
10. F. Breg, S. Diwan, J. Villacis, J. Balasubramanian, E. Akman, and D. Gannon. Java RMI Performance and Object Model Interoperability: Experiments with Java/HPC++ Distributed Components. In Proceedings of ACM 1998 Workshop on Java for High-Performance Network Computing, pages 91 -- 100, Feb 1998.
11. Hal Stern, Managing NFS and NIS, O'Reilly & Associates, Inc., Sebastopol, California, 1991
12. Jim Waldo. The Jini Architecture for Network-centric Computing. Communications of the ACM, pages 76--82, July 1999.
13. Jearding, Dean F. and Stasko, John, T. The Information Mural: Atechnique for Displaying and Navigating Large Information Spaces. Proceedings of the IEEE Symposium on Information Visualisation, Atlanta, GA., pp 43-50, Oct. 1995.
14. Cleary D., O'Donoghue, D. "Generating a Topically Focused Virtual-Reality Internet", INET 2000 - The 10th Annual Conference of The Internet Society, Yokohama, Japan, 18-21 July 2000.
15. M. W. Macbeth, K. A. McGuigan, and Philip J. Hatcher. Executing Java Threads in Parallel in a Distributed-Memory Environment. In Proc.CASCON'98, pages 40--54, Missisauga, ON, 1998. Published by IBM Canada and the National Research Council of Canada.
16. Resiss, S.P. A Framework For Abstract 3D Visualisation. Proceedings of the 1993 IEEE Symposium on Visual Languages, pp 108-115, Aug. 24-27, 1993.
17. Spence, R. "Information Visualisation", Addison-Wesley/ACM Press, 2001.17 Spence, R. "Information Visualisation", Addison-Wesley/ACM Press, 2001.
18. Jearding, Dean F. and Stasko, John, T. The Information Mural: Atechnique for Displaying and Navigating Large Information Spaces. Proceedings of the IEEE Symposium on Information Visualisation, Atlanta, GA., pp 43-50, Oct. 1995.