Generating a Topically Focused Virtual-Reality Internet

D. Cleary, (david.cleary@eei.ericsson.se)

Ericsson, Athlone, Co. Westmeath,

Ireland.

D. O'Donoghue, (diarmuid.odonoghue@may.ie)

Dept. Computer Science, NUI Maynooth, Co. Kildare,

Ireland.

Table of Contents

Abstract

Surveys highlight that Internet users are frequently frustrated by failing to locate useful information, and by difficulty in browsing anarchically linked web-structures. We present a new Internet browsing application (called *VR-net*) that addresses these problems. It first identifies semantic domains consisting of tightly interconnected web-page groupings. The second part populates a 3D virtual world with these information sources, representing all relevant pages plus appropriate structural relations. Users can then easily browse through around a semantically focused virtual library.

1 Introduction

The Internet is a probably the most significant global information resource ever created, allowing access to an almost unlimited amount of information. In this paper we describe two inter-related difficulties suffered by Internet users, and their combined influence on web use. We then introduce an integrated "search and browse" solution tool that directly tackles both issues. We also examine previous work in the search and visualisation literature, identifying similarities and differences.

The Nordale [1] survey of Internet users identified that the most "Dissatisfying Experiences" of Internet users is that they "Couldn't find what's needed". This is clearly something that existing web-index engines attempt to address - though apparently with little success. The second most dissatisfying experience is "Confusing site navigation"; a problem that can be traced jointly to poorly structured web-sites, and to limitations in current Internet browser technology (*eg*

Netscape and Communicator). Interestingly, the fifth ranked "Complaint using the Internet" also concerns the inability to find needed information, with 80% of pages visited only once by users. The Internet's vast potential will not be realised, as long as users experience significant difficulty in locating and navigating between information sources. We shall now address issues of identification and navigation in turn.

1.1 Web Index Engines

Before we begin, first note that we make a clear distinction between *web indexing engines* and structurally aware *search engines*. Indexers identify all pages containing the search terms, while search engines identify a small subset of this data. Significantly, indexing engines ignore hyperlinks while Page and Brins [2] search engine bases searching on the Internet hyperlink structure.

Indexing engines firstly identify pages containing a users search terms and then apply a ranking function to identify "useful" pages. Different indexes use functions based on features such as: frequency of term occurrence, preference for header text, presence of meta-tags, previous visitation statistics and proximity of search terms. Query term context may also be included with the search results [3], but this requires user intervention. Intelligent agents have also been applied to assist with Internet search [4]. Meta-search engines combine the results of individual indexers, thereby combining their ranking functions. None of these techniques, however, appear to usefully disambiguate between useful and off-topic pages. Later, we shall describe the search engine used by VR-net.

1.2 Site Navigation

Confusing site navigation can be partly attributed to the lack of a "browsing context" when navigating. It is virtually impossible to construct a "mental map" of a site's structure from a traditional browser and this can lead to a feeling of disorientation. Well designed sites minimise, but rarely eliminate, this problem. Site maps are a common technique to minimise disorientation.

Internet surfing generally present users with many conflicting options; up, home, back, forward, index, search, site map, plus numerous hyperlinks. Every time you descend a level, the previous levels links disappear and the user is immediately faced with the choices of the next level. Such a bewildering array of options can be confusing for users, and often leads to an *off topic* page (of no interest to the user), forcing backtracking [4] and causing disorientation.

Thinking of a web page as the root of a tree of pages, usually "back" operation returns you to a parent node. But this *backtracking* looses connection with the recently abandoned page - and many users expect "back" to eventually list all visited pages. To revisit an abandoned page, users must find the correct link from the correct parent page. This can be unrealistic when exploring a complex site, as the user may have examined many pages before wishing to return to that page. Furthermore, web indexing engines present their results as an ordered list of many thousands (or millions) of pages that contain the query terms. Little wonder then, that many users find using the Internet a dissatisfying experience. Later, we shall describe the Virtual Reality infterface created by VR-net.

2 Design Strategy

We now describe VR-nets theoretical foundations, comparing it with existing work in the areas of search and visualisation.. We start by examining our technique for identifying suitable information sources, and its utilisation of existing web-indexing engines. We view this as the

task of identifying a small, semantically credible, collection of web-pages that relate to a users query. The second part then, describes how we use a 3D virtual world to represent this focused Internet subset. VR-net generates a "virtual library" response to each query, displaying chosen pages and their inter-relationships.

A typical users query consists of a sequence of terms that appear specific, but experts can frequently add terms to specify a query - similar to the query expansion of *excite.com*. One cannot point to one "best" page, and a more appropriate response might contain alternative collections of pages - each dealing with a different aspect of the query. This effectively forms a focused virtual library of information, which together provide all necessary information.

2.1 Information Retrieval

Citations have long been used to estimate the importance of academic publications. Hyperlinks have many uses, one being their use to cite references. Authors include in web pages hyperlinks to other topically related web-pages, and new search engines use this hyperlink structure to identify collections of documents related to topics [5], [6]. The Google search engine makes extensive use of link structure plus anchor text to identify useful web pages [2]. As our search technique is similar to Clevers, we shall describe our technique first before highlighting the differences.

We use the number of external links a page receives to indicate that pages merit. However, not all hyperlinks function as citations, so we combine these links with the presence of query terms to approximate genuine citations. Unlike Brin and Page [2] we do not currently distinguish between anchor and other text.

2.1.1 Attractors and Satellites

In the remainder of this paper, we use the following terminology. An *Attractor Site* is a site that is pointed to by many other pages, and all of which refer to a particular topic. Any collection of pages that are referred to by other pages on some topic are deemed to be primary source of information on that topic, by virtue of the consensus of web authors who refer to that site. An *Attractor Member* then is an individual page within an attractor site. A *Satellite Page* is one that refers to the search topic and that also links to an attractor site. Satellites then are secondary sources of information. Useful attractors are composed of many inter-linked attractor members, collecting many satellite pages referring to this information resource. Our topicalized virtual world is composed primarily from these two sources of information.

Attractors and satellites are similar to Clevers' *authorities* and *hubs* [7]. There are two primary difference, firstly authorities are individual pages while attractors members can be spread about a web-site. Secondly, Clever requires an iterative process to rank pages, while ours uses multiple "one shot" functions. Chakrabarti *et al* [7] discuss in greater detail the process of identifying authorities and hubs. Terveen [5] and others also use the web-site as the basic unit of reference when searching the web. We feel that this gives an extra degree of flexibility to our retrieval mechanism, and produces results that are more in tune with our visualisation process.

2.2 Visualisation

Software visualisation is a human-computer interfacing technology, devised to improve human understanding of complex computer software using graphical design, typography, animation and other techniques (Price *et al* [8]). Various techniques exist for visually understanding large information spaces [9]. Portraying software in virtual reality has been shown to have significant benefits. The most realistic way of understanding abstract representations is to map them to the

physical world [10], and in our case this mould be a suitably organised virtual library.

Static data visualisation is most applicable to our task, showing that contents of, and links between, HTML pages. Abstract graphical objects provide the foundation for our visualisation, satisfying the criteria of: completeness, extensibility, hierarchy, parameterization, abstraction [11]. Each page has its own VRML [12], [13] node, and each relevant hyperlink is represented by as line between these nodes. To avoid information overload, we avail of VRML's ability to display information at varying levels of detail - as described below.

Googles' results are displayed in a simple list structure [2], but we wish to alleviate users browsing difficulties. The combined problems of search and navigation have been addressed by Terveen *et al* [5] who use the web-site as the basic unit of reference. Interestingly, these results are displayed graphically in the form of an "auditorium seating visualisation" - like seats arranged around a central state. Wood *et al* [14] combined visualisation techniques with a browsing environment, dynamically graphing users browsing history. Semantically related pages typically end up in depicted in similar regions of space, forming a kind of user generated semantic map. However, we wish to offer a computer generated world before browsing beings. Terveen *et al* [5]. combined a search engine with a graphical depiction of search results. They display their results as a 2D layout graph of pages, each with a "thumbnail" depiction of that pages "front door". Other work [15], [16] has focused on supporting a virtual reality web-browser, however this is focused more in providing interactivity among simultaneous users, and displaying avatars for each

VR-net supports browsing within a virtual world of attractors and satellites. Users can perform all browsing within a topicalized "site map", which serves to structure browsing thereby minimising disorientation. All "on topic" pages are located within this world, so the user shouldn't normally need to follow a hyperlink from a given page. Even pages that reference each other can be visited through the "virtual world" interface - though this is supported. Users navigate through the virtual world, and also examine the underlying pages, through a standard (VRML enabled) browser. By presenting all relevant information within a static environment, we aim to eliminate the users "dissatisfying experiences". The remainder of this paper describes how the search engine and the visualisation mechanism were created.

3 VR-net Implementation

In this section we discuss the implementation details of VR-net. VR-net is a Java based end-user application that first accepts a user's search terms and identifies atttractor sites and satellites related to this topic. These results are used to generate a VRML world depicting all discovered information within a single 3D environment. This visualisation framework is based on our previous work that generated a Virtual Reality Internet [17], [18] . This generated a virtual world representing individual sites, and inter-linked worlds supported a 3D Internet. VR-net is partly an adaptation of this previous work, but replaces co-located pages with the scattered nodes identified by our search mechanims.

VR-net uses a number of key components that collectively create the required VRML world. It starts by the collecting a seed set of URLs on the given topic from existing Internet indexes. Two iterative ranking schemas are applied to this set, identifying Attractors and Satellites. Attractors are then mapped into the VRML world, integrating many topicalized pages into a single information world. Satellite nodes, relevant hyper-links and any additional information are super-imposed around the core Attractor nodes. The architecture of VR-net is summarised in the UML Package Diagram (Figure 1).
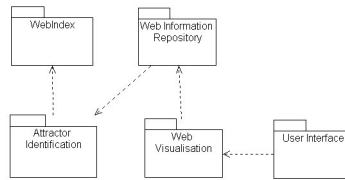
Figure 1: *UML Package Diagram for VR-net.*

3.1 Web Information Repository

At the heart of VR-net is a complex information model that stores the Attractors, Satellites and the inter-relationship between them. As this is central to both the search and visualisation processes, we will examine it first. The web information respository is built by creating a series of associated object hierarchies, where the aggregation of these hierarchies provides a full view of the Internet domain in question. The construction of these object families is based on the notion of a site reference rather than an HTML page reference, which is necessary to support our concept of attractor sites. Also, within the attractor hierarchy, added structural and content information is stored to provide added information to the final visual representation of the section of web been investigated. This information is designed to facilitate the creation of VRML worlds. As the underlying structure of the web is based on a complex maze of electronically linked documents the interrelationship of the nodes contained within the repository are modelled within their own object model.

One of our chief design goals when designing the information repository was to achieve a distributed and scalable data structure. To realise this we implemented the information repository in a location independent manner. The ability to serialise the information repository to a platform independent format supports the ability to transport this information the Internet itself. Thus, information retrieval may be performed separately from visualisation. This was accomplished by building the complex object oriented containers on top of the new Java 2 container framework implementing the *java.io.Serializable* interface for each of the information hierarchies.

3.2 Identifying Attractors

Building the web information repository is a three-stage algorithm. 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.

Initiation Stage:

The first stage begins by retrieving a seed set (A) of URLs from a number of web indexes, for some given search topic. We then apply a ranking algorithm (α) to this base set, followed by a threshold function (μ) applied. This identifies the set of candidate Attractors (B) and candidate satellites (F).

(i) $B = \mu \, (\alpha \, (A))$

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. The output set of URLs (B) from formula (i) contains Attractors (C), and satellites (D), and also topically unfocused candidate attractors (E), which we wish to eliminate. B is used to drive a second-phase information retrieval process

which in turn is subjected to an attractor ranking function (δ) . It is from this data that attractors (C) are identified, together with relevant satellite nodes (D).

(ii) $C = \delta (B)$

Satellite Organisation Stage:

The final stage is used to complete our collection of satellite nodes, as E contains some important satellites relevant to the attractors C, therefore the subset of FE pointed to by C is recovered. These pages were initially discarded by the threshold function, but recovered due to their proximity to attractors (C).

(iii) $D = \varepsilon (C+\beta (E))$

Significantly, both ranking functions δ and α are similar useing the key premise that authorities are entities that are pointed to by numerous external web pages. δ differs from α by extending its base set of URLs by propagating each web page and extracting new URLs contained within the each and add them to its internal reference store. Furthermore, as F contains some important satellites relevant to the attractors C, the subset of F pointed to by C is recovered. These pages were initially discarded by the threshold function, but recovered due to their proximity to attractors (C).

3.3 Attractor Identification

Our application does not focus on all aspects of a web site, concentrating on the URLs that are extracted each page in the web information repository. To aid comparison of URLs and to allow a useful structure to be determined, we apply a series of analysis stages to each URL, including the following steps.

- Resolution of partial and relative URL to absolute URLs.

- Protocol analysis of URLs to classify the web resources. Only a subset of the protocols contained within URLs; HTTP, FTP, GOPHER, and MAILTO are currently considered [19].

- Identify attractor nodes by examining a DNS (Domain Name Server) to which the URLs belongs.

As well as analysing URL syntax, additional information is extracted from each HTML page to summarise its content. This information is then used to animate the VRML environment, and providing a detailed view of the attractors, and illustrating the inter-relationship between attractor members and satellites. Each of the detailed views is represented as VRML primitives combining texture mapping with textual information. The representation of hyperlinks between attractor members is divided into two categories depending on their position within the web-site (recursive links are not graphically represented):

- Links from a node to another node located further down the parse tree

- A link coming from a node lower down the site tree back to the current node.

4 Visualisation

Following the population of the web information repository, the visualisation process begins. We now examine the task of visualising these search results using Virtual Reality, created using VRML. This virtual world is designed for transfer across the web and for viewing through a desktop VRML browser, such as Silicon Graphics Cosmo player - a plug-in for conventional web browsers. The Cosmo player provides the user full three dimensional interaction with the VRML world.

Our visualisation relies on four main features of VRML; prototypes, inline files, LOD nodes, and anchor nodes. Prototypes combine primitive shapes (box, cone, cylinder, line and sphere) to create complex shapes that VR-net can then treat as a primitive within any given world. Inline files allow large worlds to be defined across multiple files, partitioning complex worlds into smaller unitized files. LOD (Level of Detail) nodes allow alternate representations of one virtual object, with the greater detail being displayed when the user approaches, triggered by proximity sensors. LOD nodes are used extensively to increase the amount of information presented when focusing upon one attractor site, revealing satellite detail and related information. Finally, 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-page.

4.1 Navigating the Virtual Environment

The VRML world contains various representations of attractor sites, the goal of this is to give the user varying levels of information depending on their location within the world. Theses perspectives are achieved by incorporating proximity sensors around attractor sites. This checks if the viewer is within a predefined region around a node, and when activated, reveals additional information on an attractor site.

VR-net supports both a *summary* and a *detailed* view of every attractor site. Initially, the user is presented with a summary view of all attractors. This view uses simple groups of VRML primitives and prototypes, limited textures, and some text to allow speedy rendering of large number of attractors. This allows the user to see the entire world and to understand the relationship between the various attractor sites. On activation of a virtual world proximity sensor, in conjunction to the LOD node, extra information to that part of the scene is added. Detail on attractor members is now clearly visible, revealing each one with summary information on the contents of each.

4.2 Creation of the VRML worlds

Generating any VR-net world involves the integration of authored VRML with dynamically generated code. Authored code defines standard entities within the world (eg mail-to, http), creating 3D web icons to aid browsing. The web information repository is used to generate the dynamic components, which account for most code in the final world.

VRML prototypes, which provide the basic building blocks for creating our VRML world, are pre-written segments of VRML. As can be seen from the process diagram (Figure 2), the VRML world is build up in many stages. The first stage creates a series of inline VRML files to represent the attractors and satellites. Following this, the position and layout to the world is determined. Sensor information is then added to the world, revealing additional information when the users attention is focused on a particular location. The relationship between nodes is determined and added next. Finally graphical textures are applied to certain nodes types as an aid to recognition, reflecting the contents of the underlying entity (eg web-page).
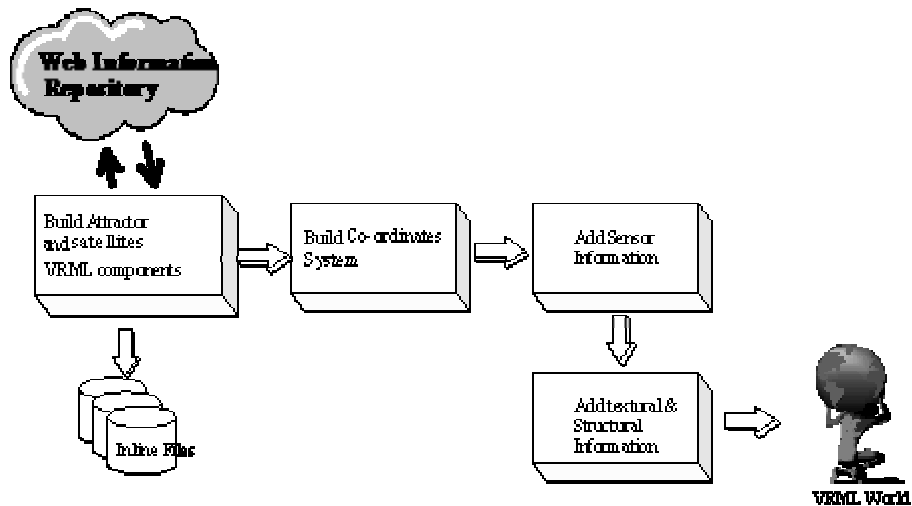
Figure 2: *Visualisation Process Diagram*

The VR-net visualisation components are:

1) *Build attractor and satellite components* generates summary and detailed manifestations of the information repository.

2) *Build co-ordinate system* determines the best graphical layout of attractors in relation to each other.

3) *Add sensor information* translates the co-ordinate system into VRML code, and maintains the inter-dependencies between other VRML components.

4) *Add texture & structural information* annotates the world with texture, text, and other information to enrich the users browsing world. This enriches the virtual world, facilitating page identification.

The final VRML world is subjected to a number of cross-compilation and validation stages before it is ready for display. To achieve the cross-compilation to VRML a hierarchy of VRML component is created. As the visualisation stages progresses, A polymorphic VRML generation algorithm is used to combine the four visualisation components (listed above), with information from the information repository. This method involved using runtime binding, manipulating the Java class loader (*java.lang.ClassLoader*) to dynamical bind the required VRML components together - thereby creating the required VRML world.

5 Discussion

The detection of attractor sites and satellite nodes can be seen as successful approach to increase understanding the complex information model that is the world wide web. Our technique employs a URL referencing approach of determining the importance of a information source, and identifys information from the glut of data identified by traditional index based search engines.

Another major problem when dealing with information is the means by which it is presented to the user. The solution we adopt is to represent the resulting information in graphical from. Various techniques for visually understanding large information spaces exist (Jerding, Dean and Stasko, 1995). The most realistic way of understanding abstract representations of structures we

believe is to try and map them to the physical world, by manipulating three-dimensional graphics that allow interaction with constituent graphical objects.

As VR-net depends on web index to start the ranking process, future study into integrating VR-net with a web index is needed. This dependency is one of key features slowing down its operation, as all information must be retrieved via the internet from busy indexing engines. The process of constructing a world is therefore typically not done interactively. However, useful virtual world for often used topics can prove invaluable - a specialised virtual library. Because of the structured nature of the world, a familiar topicalized world can become as useful as a well organised bookcase. Future distribution of the application is also needed to optimise the performance of the system.

6 Conclusion

We considered the problem of dealing with the data overload that frequently accompanies Internet use, and at the limitations of current Internet indexing engines. We also considered some of the inherent limitations of current Internet browsers, which can have a disorienting effect when users perform extensive "manual" searching. We introduced VR-net as a tool for interacting with the Internet in a more useful and productive manner.

VR-net firstly identifies attractor sites as credible sources of information on a users topic, detecting satellites that refer to these attractors. This information is used to create a complex information model representing that subset of the Internet that's of interest to the user. VR-net generates a virtual reality world depicting the content and structure of the identified Internet subset, and it is in this virtual domain that users interact with the web. Attractors sites are depicted and navigation with this environment reveals increasing levels of detail, including the satellites associated with each. All relevant structural relations are graphically depicted, and text and graphical cues provide summary information on information sources located within the world - and detailed examination of each pages contents is also supported.

References

[1] Nordale Services: http://nservices.com/internet.htm, 1999.

[2] 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.

[3] Lawrence, S. Lee Giles, C. *"Context and page analysis for improved web search",* IEEE Internet Computing, 2,4, 38-47, 1998.

[4] Lieberman, Henry. Letizia: *"An Agent That Assists Web Browsing",* International Joint Conference on Artificial Intelligence, Montreal, August, 1995.

[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] Chakrabarti, S. Dom, B. Gibson, D. Kleinberg, J. Kumar, S.R. Raghavan, P. Rajagopalan, S. Tomkins, A. *"Hypersearching the Web",* Scientific American, June 1999 - 2.

[7] 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.

[8] Price B. Baecker R. Small, I. *"A principled taxonomy of Software Visualisation",* Journal of Visual Languages and Computing, 3, 4, 211-266, 1993.

[9] Jerding, Dean, F. Stasko, J. *"The Information Mural: a technique for displaying and Navigating Large Information Spaces",* Proc. IEEE Symposium on Information Visualisation, Atlanta, GA., 43-50, Oct. 1995.

[10] Kalawsky R. S. *"The Science of Virtual Reality and Virtual Environments",* Addison-Wesley, 1994.

[11] Resiss, S.P. *"A Framework For Abstract 3D Visualisation",* Proceedings of the 1993 IEEE Symposium on Visual Languages, 108-115, Aug. 24-27, 1993.

[12] VRML Consortium, "*The Virtual Reality Modelling Language",* ISO/IEC 14772-1: 1997.
[13] Ames A., Nadeau, D. and Moreland, J.L. *"VRML 2.0 Sourcebook"*, 2nd Edition, Wiley, 1996.

[14] Wood, A.M. Drew, N.S. Beale, R. and Hendley, R.J. *"HyperSpace: Web Browsing with Visualisation" i*n Third International World-Wide Web Conference Poster Proceedings, pp. 21-25, Darmstadt, Germany, April, 1995.

[15] Huang J. Fang-Tsou, C. Chang J. *"A Multiuser 3D Web Browsing System",* IEEE Internet Computing, 2,5, 70-79, 1998.

[16] Selfridge, P. Kirk, T. *"Cospace: Combining Web-Browsing and Dynamically Generated 3D Multiuser Environments",* SIGART 10, 1, 24-32, 1999.

[17] Cleary, D. *"Visualising Web-Site Structure using VRML",* Tech. Report, Department of Computer Science, NUI Maynooth, Ireland, 1998.

[18] Cleary, D., O'Donoghue, D., *"VisualExpresso: Generating a Virtual Reality Internet",* in P. Sloot, M. Bubak, A. Hoekstra, and B. Hertzberger (Eds.) High Performance Computing and Networking HPCN-99: Virtual Reality Workshop LNCS 1593, 797-806, Springer-Verlag, 1999.

[19] Network Working Group T. Berners-Lee.T. Request for Comments: 1738, *"Uniform Resource Locators (URL)"* Editors L. Masinter Xerox Corporation, M. McCahill University of Minnesota, December, 1994.