Visual Exploration of Web Spaces

Víctor Pascual Cid

TESI DOCTORAL UPF / 2010

Directors de la tesi

Dr. Ricardo Baeza-Yates,
Departament de Tecnologies de la Informació i les Comunicacions

Dr. Juan Carlos Dürsteler López,
Departament de Tecnologies de la Informació i les Comunicacions

UNIVERSITAT POMPEU FABRA

"I hear and I forget.
I see and I remember.
I do and I understand."
Confucius

Agraïments

Una tesi doctoral requereix d'un esforç gran i constant. Durant tots aquests anys he tingut la gran sort de comptar amb l'ajut i el suport de molta gent a qui voldria mencionar a continuació.

Per començar, volia mostrar el meu més sincer agraïment als meus dos directors: al Juan Carlos Dürsteler per introduïr-me en el món de la Visualització d'Informació, per ser el primer que va creure i comptar amb mi per iniciar aquest projecte i per haver-me ajudat durant tot el procés que ha portat fins a aquesta tesi. Al Ricardo Baeza per ajudar-me a formar-me com a investigador i per guiar-me durant tots aquests anys, sense els comentaris i propostes del qual mai hagus arribat a escriure aquesta tesi.

Gràcies, de manera especial, al Christian Middleton i al Sergi Mínguez per formar part de l'equip de desenvolupament del WET. Sense vosaltres el WET mai hagués estat l'eina que és avui en dia, i que m'ha ajudat a desenvolupar aquesta tesi.

Gràcies al Josep Blat per tots els consells que m'ha donat des de que vaig acabar la carrera, i que m'han ajudat a ser on sóc ara mateix, i a la resta de companys i amics de la UPF (Sergio, Fabien, Ximena...). Així mateix, volia agrair al Vicente López pel suport donat durant els meus últims anys de recerca, i als meus amics de la Fundació Barcelona Media (Francis, Bernat, Marc, Carlos i molts altres). M'agradaria mencionar especialment a l'Andreas Kaltenbrunner, qui ha estat disposat en tot moment a ajudar-me i a escoltar-me.

Aquests anys de recerca m'han brindat l'oportunitat de conèixer una infinitat de persones que, d'una manera o altra han participat i influït en aquesta tesi. No voldria doncs deixar passar aquesta oportunitat per donar les gràcies al Pere Rovira, al Jaume Clotet, a l'Andrés Flores, al Jose Panzano, al Jordi Roura i a molts altres membres de la comunitat d'Analítica Web de Barcelona i del Conversion Thursday. Gràcies també al meu amic Michael Sedlmair i a tots els col·legues amb qui he compartit grans moments conferència rera conferència. Gràcies també als membres del tribunal d'avaluació d'aquesta tesi,

Ah, i gràcies també a tots els usuaris que han format part de les diferents avaluacions fetes durant tots aquests anys, i a la Karen per la revisió d'aquest document!

En l'apartat més personal, crec que no tinc prou paraules per agrair tot el suport donat per part de la meva familia. Gràcies als meus pares per recolzar-me en tot moment i haver-me ajudat a perseverar i a ser constant, als meus germans (cunyat i nebots!) per estar sempre al meu costat, i tenir la santa paciència d'aguantar-me sempre. I a la meva 'prima' per les llargues converses i reflexions que hem compartit. Gràcies a la Mia per tot el suport que m'ha donat durant aquests anys, a la meva 'coach' Ana, i a tots els meus amics que han patit els meus mals humors, han aguantat llargues converses, i que també m'han acompanyat en la celebració dels èxits.

A tots vosaltres i a molts altres més que d'una manera o altra heu estat part de la meva vida durant els darrers cinc anys, MOLTES GRÀCIES!

Acknowledgements

A PhD research requires great and constant effort. During these years I have been incredibly lucky to count on the support of a lot of people who I would like to mention hereafter.

First of all, I would like to offer my most sincere thanks to my supervisors: to Juan Carlos Dürsteler for introducing me into the field of Information Visualisation, for being the first to have faith in me to lead this project and for helping me during the whole process that has led me to this thesis. To Ricardo Baeza, for helping me to build myself as a researcher and for guiding me during all these years. Without your comments and suggestions I would have never been able to write this dissertation.

I would like to acknowledge, in a very special way, Christian Middleton and Sergi Mínguez for being part of the development team of WET. Without you guys, WET would have never been the tool that it is now, that has helped me to conduct my research.

Thank you to Josep Blat for all the advice he has given me since I finished my degree, which has helped me get to where I am today; and to the rest of my colleagues and friends from the UPF (Sergio, Fabien, Ximena...). Moreover, I would like to thank Vicente López for his support during the last few years of my research, and to my friends from Fundació Barcelona Media (Francis, Bernat, Marc, Carlos and many more). I would like to specially mention Andreas Kaltenbrunner, who has always been willing to help and advise me.

These years of research have given to me the opportunity to meet an infinite number of people who, in one way or another, have participated and influenced this thesis. Therefore, I would like to acknowledge Pere Rovira, Jaume Clotet, Andrés Flores, Jose Panzano, Jordi Roura and the members of the Web Analytics community and the Conversion Thursday in

Barcelona. Also, thank you to my friend Michael Sedlmair, and to the rest of my colleagues with whom I have shared a lot of great moments in all the conferences which we have attended together. Thank you also to the members of the evaluation committee of this thesis.

Another big thank you goes out to all the users that have been part of the evaluations that I have done over the last few years, and to Karen for kindly reviewing this document!

On a more personal note I can't thank enough my family for all the support they have given me. Thank you to my parents for supporting me and helping me to persevere and be constant; to my brother and sister (brother in law, nephews!) for standing by me, and for having the patience to put up with me, and to my 'prima' for all those long conversations and great moments together. Thank you to Mia for all the support she has given to me these last few years, to my 'coach' Ana, and to all my friends who have put up with my bad moods, long and meaningless conversations, and with whom I have also had the opportunity to share my successes.

To all of you and to many more who have been part of my life during the last five years, THANK YOU VERY MUCH!

Abstract

Web spaces have been the subject of in-depth studies since the Web became the largest data and information repository publicly available. Nevertheless, the vast amount of data that Web mining techniques generate from them is difficult to understand, suggesting the need to develop new techniques to gather insight into them in order to assist in decision making processes.

This dissertation explores the usage of InfoVis/VA techniques to assist in the exploration of Web spaces. More specifically, we present the development of a customisable prototype that has been used to analyse three different types of Web spaces with different information goals: the analysis of the usability of a website, the assessment of the students in virtual learning environments, and the exploration of the structure of large asynchronous conversations existing in online forums.

Echoing the call of the Infovis/VA community for the need for more research into realistic circumstances, we introduce the problems of the analysis of such Web spaces, and further explore the benefits of using the visualisations provided by our system with real users.

Resum

Els espais Web han estat objecte d'estudi des de que la Web s'ha convertit en el repositori públic d'informació més gran. Tot i això, el gran volum de dades que les tècniques de mineria Web proporcionen sobre aquests és generalment molt difícil d'entendre, provocant la necessitat de desenvolupar noves tècniques que permetin generar coneixement sobre les dades de manera que facilitin la presa de decissions.

Aquesta tesi explora la utilizació de tècniques d'Infovis/VA per ajudar en l'exploració d'espais Web. Més concretament, presentem el desenvolupament d'un prototipus molt flexible que hem utilitzat per analitzar tres tipus diferents d'espais Web amb diferents objectius informacionals: l'anàlisi de la usabilitat de pàgines Web, l'avaluació del comportament dels estudiants en entorns virtuals d'aprenentatge i l'exploració de l'estructura de grans converses asíncrones existents en fòrums online.

Aquesta tesi pretén acceptar el repte proposat per la comunitat d'Infovis/VA de fer recerca en condicions més reals, introduint els problemes relacionats en l'anàlisi dels espais Web ja esmentats, i explorant els avantatges d'utilizar les visualitzacions proporcionades per la nostra eina amb usuaris reals.

Resumen

Los espacios Web han sido objeto de estudio des de que la Web se ha convertido en el mayor repositorio público de información. Sin embargo, el gran volumen de datos que las técnicas de minería Web generan sobre ellos puede llegar a ser muy difícil de entender, provocando la necesidad de desarrollar nuevas técnicas que permitan generar conociomento sobre esos datos con el fin de facilitar la toma de decisiones.

Esta tesis explora la utilización de técnicas de InfoVis/VA para ayudar en la exploración de espacios Web. Más concretamente, presentamos el desarrollo de un prototipo muy flexible que ha sido utilizado para analizar tres tipos distintos de espacios Web con distintas metas informacionales: el análisis de la usabilidad de páginas Web, la evaluación del comportamiento de los estudiantes en entornos virtuales de aprendizaje y la exploración de la estructura de grandes conversaciones asíncronas existentes en foros online.

Esta tesis pretende aceptar el reto propuesto por la comunidad de InfoVis/VA de llevar a cabo investigaciones en condiciones más reales, introduciendo los problemas relacionados con el análisis de los espacios Web ya mencionados, y explorando las ventajas de utilizar las visualizaciones proporcionadas por nuestra herramienta con usuarios reales.

Contents

List of Figures xii

List of Tables xvi

1 Introduction 1
1.1 Motivation: Visualising Web Spaces 1
1.2 Research Goals 3
1.3 Contributions 4
1.4 Organisation of the Dissertation 6

2 Background and State of the Art 9
2.1 Information Visualisation and Visual Analytics 9
2.2 The Process of Information Visualisation 12
2.3 From Data to Information 14
2.3.1 Representing Relations between Data: Graphs 16
2.3.2 Characterising Web Spaces 18
2.3.3 Web Mining 19
2.3.4 Web Analytics and Search Engine Optimisation 27
2.4 From Information to Visual Representation 29
2.4.1 Graph Drawing 29
2.4.2 Tree Layouts 33
2.5 From Visual Representation to Understanding 39
2.5.1 The Interaction Process 40
2.5.2 Interaction Techniques 41
2.6 Evaluating Interactive Visualisations 42

3 WET: A System to Support Visual Data Mining 47
3.1 System General Overview 47
3.2 The Graph's Logic System 49
3.3 Visualisation System 52

3.3.1 Evaluation of the Radial Tree Layout 52
3.3.2 User Interface and Interaction Design 58
3.3.3 Implementation Details 63
3.4 Conclusions 66

4 Visual Mining for Website Evaluation 67
4.1 Research Problem 67
4.2 Related Work 68
4.3 A Hybrid Approach for Visualising Website Data 71
4.3.1 Gathering and Preprocessing Website Data 74
4.3.2 Visualising the Webgraph Structure as a Hierarchy . . 75
4.3.3 Combining Usage Data and Website Structure 81
4.3.4 Interaction Design 83
4.3.5 Evaluation 86
4.4 Visual Analysis of Users' Behaviour 88
4.4.1 Identifying Users' Paths 88
4.4.2 Visualising User Paths 90
4.4.3 Characterising Users' Navigation as a Hierarchy . . . 91
4.5 Evaluation of the System 99
4.5.1 Evaluation Results 99
4.5.2 Lessons Learned and Study Limitations 104
4.6 Conclusions 105

5 Visualising Virtual Learning Environments 109
5.1 Research Problem 109
5.2 Related Work 110
5.3 Evaluation Methodology 111
5.4 Case Studies 114
5.4.1 AutoLearn Project 114
5.4.2 e-STUB Project 121
5.5 Lessons Learned and Recommended Features 126
5.6 Conclusions 129

6 Exploring Asynchronous Online Discussions 131
6.1 Research Problem 131
6.2 Related Work 132
6.3 Visualising Online Conversations 138
6.4 Use Case: Visualising Slashdot Discussions 141
6.5 Controlled Experiment 144
6.5.1 Study Design 145

6.5.2 Results and Analysis . . . 146
6.6 Conclusions . . . 151

7 Final Discussion 153
7.1 Contributions . . . 153
7.2 Evaluation Process and Generalisability . . . 158
7.3 Future Work . . . 159
7.3.1 Within the Web Analytics Community . . . 159
7.3.2 Within the Virtual Learning Environments Community 160
7.3.3 Within the Context of Asynchronous Conversations . 160
7.3.4 Within the Fields of Information Visualisation and Visual Analytics . . . 161
7.4 Epilogue . . . 161

Bibliography 163

List of Figures

1.1 Scope of the thesis. 3

2.1 The building blocks of Visual Analytics (TC05). 11
2.2 Values from the Dow Jones index (left), and a plot of the same values showing trend during 2009 (right). 12
2.3 Diagrams of the process of InfoVis. 13
2.4 The process of converting raw data into information. 15
2.5 Local graph measure: k-neighbourhood. 17
2.6 The three different types of Web Mining. 19
2.7 Example of the hierarchy of a typical online store. 21
2.8 An example of a directed graph and its associated distance matrix. 22
2.9 Example of a log file in ECLF. 23
2.10 The WUM preprocessing process proposed in (CMS99b). 24
2.11 The process of converting information into a representation. . . . 29
2.12 Example of orthogonal drawing. 30
2.13 Example of a Force Directed Layout. 31
2.14 Example of the Sugiyama layout. 32
2.15 Example of a circular layout. 32
2.16 Example of a classic tree layout. 34
2.17 Converting a hierarchy into a treemap. 34
2.18 Example of treemaps algorithms. 35
2.19 Radial space filling visualisations. 36
2.20 The radial and hyperbolic tree approaches. 37
2.21 3D approaches for visualising hierarchies. 38
2.22 The process of converting the visualisation into understanding. . 39

3.1 WET architecture overview. 48
3.2 The web mining system of the Data Management and Mining System. 49
3.3 Example of the GLS output. 51

3.4 The three visualisations used in evaluation of the radial tree layout. 55
3.5 Mean times for tasks T2-T6 57
3.6 WET user interface. 59
3.7 Sequence of images showing the animation performed when changing the root node of the hierarchy. 60
3.8 The mapping tool. 61
3.9 Interactive legends used un WET. 62
3.10 Example of a scented widget in WET. 63
3.11 Building blocks of the implementation of WET. 64
3.12 Basic WET class architecture. 65

4.1 Image of Ben Fry Anemone. 70
4.2 User's behaviour visualisation from (KE02). 71
4.3 First analytics reports from Analog showing keywords used (left) and number of page views per day (right) 72
4.4 Examples of click density visualisations showing most clicked areas/links in a web page. 72
4.5 Pageviews report from Google Analytics. 73
4.6 An example of the hierarchies extracted from the BFS and the W-BFS algorithms. Relevant links highlighted in red in Figure (c) are not visualised with the BFS approach. 76
4.7 Website structure of three different websites. 77
4.8 Average of the number of highly used links selected by the BFs and the W-BFS algorithms. The W-BFS always provides more relevant links. 78
4.9 Time performance plot of the BFS algorithm. 79
4.10 Time performance plot of the W-BFS algorithm. 80
4.11 Time performance comparison between the BFS and the W-BFS algorithms. 80
4.12 Usage data mapped on top of the website structure of three different websites. 82
4.13 Left image shows the messy visualisation with all the links of the site. Right image show most relevant links after using a scented widget. 83
4.14 Broken links report shown with Google Webmaster Tools and WET. 84
4.15 Radial distorsion applied to improve the overlapping of the nodes. 85
4.16 Example of the path completion algorithm. 89
4.17 Examples of the Session Graph. 91

4.18 An example of the visual abstractions generated by the different hierarchical approaches from a web graph. 92
4.19 Characterisation of the websites used in the experiment and its results. 93
4.20 The exploration of the different visualisations helped us to evaluate the usefulness of the different algorithms for extracting hierarchies. 95
4.21 Relation between nodes and edges of the graphs to be handled by the Edmond's algorithm. 96
4.22 Time performance of the Edmond's algorithm applied to the graphs formed by users' paths starting at a specific page. 97
4.23 The structure and the usage tree allow the user to compare shortest paths with the most used. 98
4.24 Screenshot of the WET visual system showing the three different visualisations available. 105

5.1 Parts of a record generated by WET's logging system. 113
5.2 Hierarchy of AutoLearn structure. Leafs of the tree represent course exercises, that can be mapped according their specific quantitative and categorical metrics. 116
5.3 Detail of an AutoLearn course in WET. 117
5.4 Students behaviour from AutoLearn project. 119
5.5 Pogo-sticking behaviour pattern found in the different courses of the AutoLearn project. 120
5.6 Statistics of the usage of the different visualisations (left) and visual attributes (right) in the AutoLearn use case. 121
5.7 Paths and frequency of access in a 1st year course; most common paths from the homepage (left) and site structure highlighting shortest path to search results pages (green nodes). 124
5.8 Differences between high and low performing students' most common paths in the 2nd year course. 125
5.9 Different visualisations of the "site galaxy" with different segments of users provided by the analyst of the e-STUB project. . 127
5.10 Statistics of the usage of the visualisations (left) and visual attributes in the e-STUB use case. 128

6.1 Web interfaces of conversations in Slashdot (left) and SourceForge (right). 133
6.2 Social visualisations presented by Viegas and Smith (VS04). . . . 134
6.3 Social visualisations presented by Viegas and Smith (VS04). . . . 135

6.4 Conversation thread visualisation from Netscan. 136
6.5 Screenshot of the Reddit forum with the TLDR system. 137
6.6 Screenshot of the user interface of WET adapted for visualising large conversations. 139
6.7 The search engine helps to locate comments with a specific text. 140
6.8 A sub-thread from a conversation. 143
6.9 Number of errors with the different experiment settings. 149
6.10 Location helps to easily identify the comment with the biggest subthread. 149
6.11 Post-test questions were ranked from 0 (low ranking) to 4 (best ranking). 150

List of Tables

3.1 Tasks used in the evaluation of the radial tree visual metaphor. . 56

4.1 Tasks performed by the users in the first evaluation of the hybrid model proposed in WET. 86
4.2 Tasks performed by the users during the final formative evaluation of WET. 100

6.1 Tasks used in the experiment with their type. 145
6.2 User's time per task with the small dataset and the Slashdot web interface. 147
6.3 User's time per task with the big dataset and the Slashdot web interface. 147
6.4 User's time per task with the small dataset and WET. 148
6.5 User's time per task with the big dataset and WET. 148

Chapter 1

Introduction

The digital revolution we are living in has led to a large scale digitalisation of data. We, as single users, are generating, on a daily basis, vast amounts of data such as bank transactions, visits in the internet, medical records, or footprints of our movements in our city thanks to the use of new technologies and social systems. The potential of interpreting and discovering uncovered patterns in this data is endless.

Nevertheless, data does not become information until it informs, that is, until it can be used as a basis for problem solving and decision-making. However, while the amount of data available is increasing at an exponential rate, our brain's capacity to interpret and analyse it remains relatively constant. Hence, its correct treatment and appropriate presentation constitute a key issue in the knowledge discovery process. These are the foundations of the fields of Information Visualisation and Visual Analytics, which lay on the generation of data representations that take advantage of the human brain's capabilities for processing visual information.

1.1 Motivation: Visualising Web Spaces

The Web has been the subject of lengthy studies due to its increasing growth and interest, becoming the most popular global communication medium and knowledge repository. As such, one of its critical issues is the ease of use, which can be understood through the investigation of users' navigation.

The discipline of Web Mining was born to tackle such problems, focusing on applying data mining techniques to the Web in order to discover its structure, usage and semantics. Such techniques can be applied to many contexts within the Web, usually called Web spaces. A Web space is a set of connected pages that together, aim to deliver interconnected contents of a specific matter. A single website or the pages appearing in a search engine results page are typical Web spaces that can be studied and understood with Web Mining techniques.

Nevertheless, the vast amount of data provided by Web Mining techniques has encouraged researchers to explore the usage of visualisation in an effort to improve its understanding, taking advantage of perceptual capabilities of our brain to communicate information in a more intuitive way (Chi02; FdOL03). Such visualisations attempt to benefit from the preattentive capabilities that allow us to digest information more easily. While much effort has been made in order to provide visualisations that aid in the analysis and understanding of Web spaces (e.g. visualisations for overcoming the "lost in cyberspace" problem (And95; MB95)), less solutions have been investigated in depth and further evaluated to support problems such as the analysis of the usability of a website, the tracking of the success of online campaigns, or the assessment of eLearning platforms.

The discipline behind the usage of visualisations to amplify cognition is Information Visualisation. Since the nineties, researchers from this field have focused on the development of new interactive visualisations that resemble theoretical abstract data structures, called visual metaphors, proving their benefits in gathering insight (CMS99a). However, the visual illiteracy of real users still means they struggle when using simple business graphics tools effectively (Pla04), which is one of the reasons why some of the most relevant visual solutions proposed during the last few decades have not reached current analysis tools. Moreover, in the last decade, the emerging discipline of Visual Analytics has tried to extensively use Information Visualisation techniques to specifically support analytical reasoning. In a sense, Visual Analytics can be understood as a broader field, as it embraces areas such as visual representations, interaction techniques, data transformation, and techniques to support production, presentation, and dissemination of the results of an analysis (TC05).

In order to provide useful clues to push forward the boundaries of the research in interactive visualisation, many authors have already proposed the need to move research into practice (Pla04; TC05; Mun09), deploying Information Visualisation and Visual Analytics tools in real world scenarios in order to learn from failure and success stories. Along the same lines of thinking, such stories can help to identify the appropriate methods that may assist the generation of insight into specific contexts and needs.

Hence, in this dissertation we will study, develop and evaluate a visualisation system that allows the representation, analysis and exploration of Web spaces in an attempt to join the disciplines of Web Mining, Information Visualisation and Visual Analytics in real world scenarios. We will cover the development of an interactive system extensively used to visualise and analyse three different types of Web spaces, characterising the main problems to be tackled as well as reporting evaluations with real users and domain experts.

1.2 Research Goals

The visualisation of Web spaces has been tackled for a long time, as the Web has become a large repository of information (And95; MB95; PB94). The "lost in cyberspace problem" led researchers to develop of visual representations of networks representing the Web to assist users in the understanding of their location and current context based on the surrounding pages.

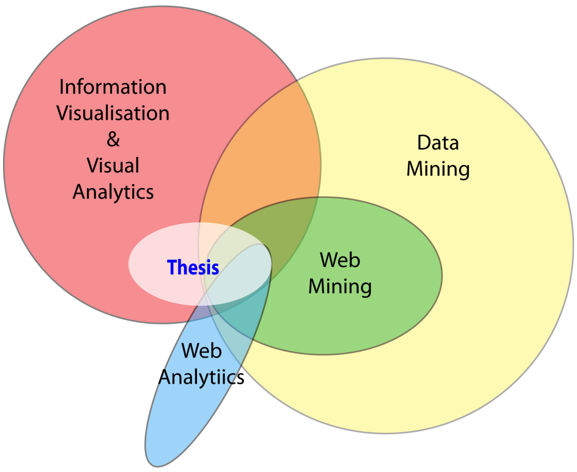


Figure 1.1: Scope of the thesis.

However, Web spaces have undergone a tremendous evolution since then, with problems arising regarding their understanding and analysis. To tackle the visual exploration of Web spaces this multidisciplinary thesis spans disciplines such as Information Visualisation, Visual Analytics, Web Mining and Web Analytics. Figure 1.1 shows the scope of this dissertation, showing the intertwined nature of its related research fields.

With this, the main research goals of this thesis are:

- The design and implementation of a flexible and customisable system able to explore and analyse graph-based data from Web spaces: the first steps involved the development of a reusable architecture to store and manipulate data that will eventually be explored through a visual tool that provides different hierarchical views of graph-based data. We will follow a user-centred methodology, conducting evaluations at different development stages to understand the success of the developed techniques as well as to comprehend real users needs.

- The study of the possibilities of Information Visualisation and Visual Analytics in the field of web analytics: we have implemented our tool into different web analytics contexts, gathering evidence of its usefulness and adequacy for specific tasks and within specific contexts.

- Explore contexts where non-expert users may benefit from the exploratory data analysis capabilities of our exploration tool: in this regard, we have used slightly modified versions of our prototype to address the exploration of large asynchronous conversations, which are gaining popularity in the Web due to their widespread usage.

- Evaluate our visualisations and provide further guidance on best practices for developing visual tools to understand Web spaces.

In general, the aim of this thesis is to move research into practice, using well known visualisation techniques to assist in the analysis of different types of Web spaces which currently present analysis difficulties.

1.3 Contributions

The main contributions of this thesis are:

- the development of a system to assist the visual exploration of web spaces.

- a novel approach for extracting contextually meaningful subgraphs that reduce the problem of dealing with very large graphs.

- a novel method for extracting hierarchies from the users' navigation of a site.

- the implementation and evaluation of new techniques for visualising web navigational data in real analysis scenarios.

- the characterisation of the problems that evaluators and instructors of Virtual Learning Environments encounter, as well as the application and evaluation of Information Visualisation and Visual Analytics techniques in such a field.

- the evaluation of the usefulness of our system for assisting the reading and exploration of large asynchronous conversations.

Most of these contributions have been previously published in international conferences and journals:

- Pascual, V. and Dürsteler, J.C. *WET: a prototype of an Exploratory Search System for Web Mining to assess Usability.* Proceedings of the 11th International Conference in Information Visualisation (2007)

- Pascual-Cid., V.*An information visualisation system for the understanding of web data.* Poster at the IEEE Symposium on Visual Analytics Science and Technology, 2008. VAST '08. (2008) pp. 183 -184

- Pascual-Cid, V. and Kaltenbrunner, A. *Exploring Asynchronous Online Discussions through Hierarchical Visualisation.* Proceedings of the 13th International Conference in Information Visualisation (2009) pp. 191-196.

- Pascual-Cid, V., Baeza-Yates, R., Dürsteler, J.C., Mínguez, S., Middleton, C. *New Techniques for Visualising Web Navigational Data.* Proceedings of the 13th International Conference in Information Visualisation (2009) pp. 621 -626.

- Pascual-Cid, V., Vigentini, L., Quixal, M. *Visualising Virtual Learning Environments: Use Cases of the Website Exploration Tool.* Proceedings of the 14th International Conference in Information Visualisation (2010) pp. 149 -155.

- Pascual-Cid, V., Baeza-Yates, R., Dürsteler, J.C. *Visual Web Mining for Website Evaluation.* To appear in Journal of Web Engineering, Rinton Press (Princeton, New Jersey).

1.4 Organisation of the Dissertation

After this introduction chapter, the reminder of the thesis is organised as follows:

Chapter 2 discusses the related work and the state of the art in the fields of Information Visualisation and Visual Analytics as well as in the representation of web spaces. We also cover the different definitions of the 'InfoVis process', that will guide the reader in the introduction of the techniques and methodologies related to the generation of visual and interactive representations of web spaces.

Chapter 3 presents the main characteristics of our visualisation system, whose prototype has been used as a base to represent data from different spaces. In this chapter we also introduce a usability test that helped us to validate the radial tree as the main visual metaphor of the system. Moreover, we introduce a new technique which aims to extract meaningful contextual subgraphs that maximise the amount of information shown within a specific problem while preserving the structural features of the main graph.

Chapter 4 provides most of the contributions of this thesis, and it is dedicated to the development of new approaches for exploring website data, aimed at supporting the assessment of the usability of websites. More specifically, we introduce techniques that enable the representation of structure and usage data of a website.

Chapter 5 presents a long term study based on the customisation of our tool to assess the evaluation of Virtual Learning Environments. We will present the main differences in the analysis of these kinds of Web spaces compared to classic Web analysis approaches, as well as to characterise the main problems that teachers and instructors must face in order to improve their online materials.

Chapter 6 goes deep into the characterisation of flash forums, the new type of online asynchronous debates that are populating the Web. We introduce the problems they involve, and explain how our visual system may assist readers and social researchers to explore conversation threads of a well known discussion website.

We conclude this dissertation with Chapter 7, where we discuss the main contributions of our work, pointing at future research directions and final remarks regarding the current state of the topics covered in this thesis.

Chapter 2

Background and State of the Art

In this chapter we discuss the state of the art of the research fields involved in this dissertation: Information Visualisation and Visual Analytics, Data Mining, Web Mining and Web Analytics. The InfoVis process diagram which describes the basic stages in the visualisation process will be used to guide the presentation of such fields.

2.1 Information Visualisation and Visual Analytics

The core of this dissertation focuses on the use of techniques from the research fields of Information Visualisation (InfoVis) and Visual Analytics (VA). InfoVis was defined for the first time by Card et al. as "the use of computer-supported, interactive, visual representations of abstract data to amplify cognition" (CMS99a). To provide a more detailed description of what this discipline is about, we will focus on the different parts of this definition as Few made (Few09):

- *Computer supported*: the visualisation is displayed by a computer with a screen.

- *Interactive*: the visual representation can be manipulated and should respond to user actions to provide ways for exploring the data.
- *Visual representations*: in a broader sense, an InfoVis system can be any system that makes extensive use of human senses to aid the user in the generation of a mental map of a dataset. Nevertheless, this discipline mainly focuses on the development of visual abstractions of data. Ware (War04) highlighted the main advantages of visualisations:
  - provide the ability to comprehend huge amounts of data.
  - allow the perception of emergent properties that were not anticipated.
  - enable problems with the data itself to become immediately apparent.
  - facilitate understanding of both large-scale and small-scale features of the data.
  - facilitate formation of hypothesis.
- *Abstract data*: data that represents physical objects has a direct visual representation either in 2D or in 3D, such as the case of geographical data. However, quantitative data, processes or networks do not have a natural way of being represented. Therefore, visual metaphors have to be used to resemble real forms that may help provide a shape to such data.
- *Amplify cognition*: while visualisations help convert data into information, interacting with them helps users to grasp such information, enabling them to see it from different perspectives that might provide insight.

VA is "the science of analytical reasoning facilitated by interactive visual interfaces" (TC05). We now describe the main building blocks of this discipline that can be seen in Figure 2.1.

- *Analytical reasoning*: VA is about incorporating evidence to influence the human brain, to allow it to apply judgements that will lead to conclusions.
- *Data representations and transformations*: InfoVis techniques must be applied to transform data into meaningful and comprehensive structures.

Figure 2.1: The building blocks of Visual Analytics (TC05).

- *Visual representations and interaction technologies*: Generated visualisations must allow the interaction and manipulation of the data.
- *Production, Presentation and Dissemination*: One of the most important concerns of VA is to communicate the analysis results in a comprehensive way.

Although InfoVis and VA are intertwined fields, VA can be understood as a broader field than InfoVis, as it represents "an integrated approach combining visualisation, human factors and data analysis" (KMSZ06), and focuses on exploring heterogeneous sources of data that usually are large and complex.

Figure 2.2 represents a simple example of the power of visualisations. While numbers on the left do not provide any clues about the data, the simple line chart on the right helps us to rapidly identify a growing trend in the stock market.

The reason why Information Visualisation and Visual Analytics techniques are so effective is due to the use of preattentive features such as colour or size, among others, that our brain is able to process rapidly. Such features are used extensively by our working or short-term memory, shifting the load of processing information from the cognitive system to the sensory system.

1-Apr-10	10,857.31	10,983.14	10,851.57	10,927.07	4,006,870,000	10,927.07
31-Mar-10	10,907.34	10,933.19	10,802.22	10,856.63	4,484,340,000	10,856.63
30-Mar-10	10,895.02	10,968.56	10,837.21	10,907.42	4,085,000,000	10,907.42
29-Mar-10	10,849.23	10,954.43	10,827.91	10,895.86	4,375,580,000	10,895.86
26-Mar-10	10,841.29	10,934.85	10,801.84	10,850.36	4,708,420,000	10,850.36
25-Mar-10	10,837.51	10,985.26	10,818.77	10,841.21	5,668,900,000	10,841.21
24-Mar-10	10,887.62	10,925.48	10,788.77	10,836.15	4,705,750,000	10,836.15
23-Mar-10	10,787.18	10,906.06	10,752.41	10,888.83	4,411,640,000	10,888.83
22-Mar-10	10,741.00	10,836.68	10,672.23	10,785.89	4,261,680,000	10,785.89
19-Mar-10	10,780.00	10,869.55	10,665.35	10,741.98	5,212,410,000	10,741.98
18-Mar-10	10,733.44	10,821.49	10,686.89	10,779.17	4,234,510,000	10,779.17
17-Mar-10	10,686.36	10,787.86	10,658.10	10,733.67	4,963,200,000	10,733.67
16-Mar-10	10,642.53	10,717.42	10,594.16	10,685.98	4,369,770,000	10,685.98
15-Mar-10	10,623.41	10,680.32	10,540.65	10,642.15	4,164,110,000	10,642.15
12-Mar-10	10,611.77	10,690.90	10,561.66	10,624.69	4,928,160,000	10,624.69

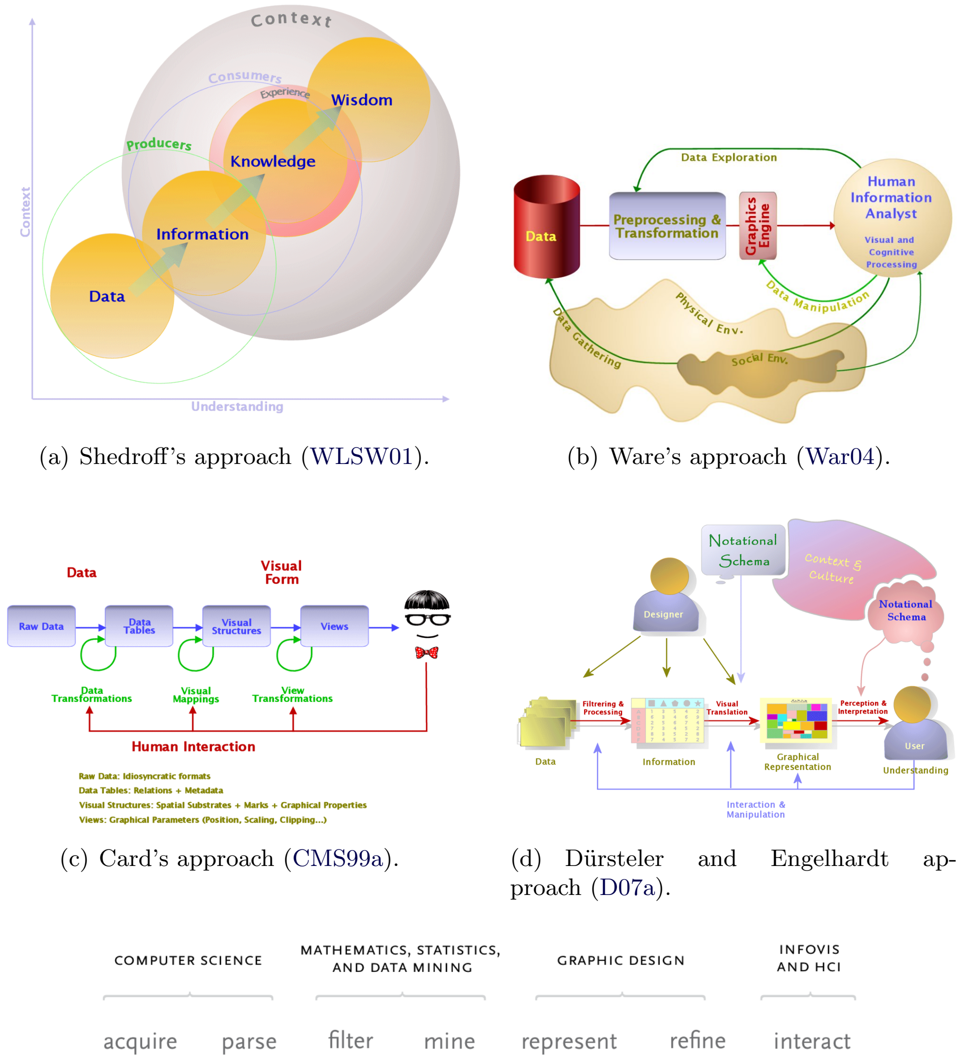
Figure 2.2: Values from the Dow Jones index (left), and a plot of the same values showing trend during 2009 (right).

Form now on, we will use the term InfoVis/VA to refer to the set of intertwining techniques embraced by the Information Visualisation and Visual Analytics areas.

2.2 The Process of Information Visualisation

The understanding of the basis of data transformation into insight, known as the *the process of InfoVis*, is crucial for developing effective strategies that help users to reach their informative goals. Several conceptual approximations to such a process have been presented, as illustrated in Figure 2.3. However, all of them converge in the definition of three main steps, specifically named by Dürsteler and Engelhardt (D07a):

- *From Data to Information*: Once relevant data regarding a target problem has been collected, it has to be structured and organised in order to be transformed into information. Three tasks can be identified concerning such a conversion:
  - the *collection and storage* of raw data relevant to the object of study
  - the *processing and transformation* of such data to filter errors. This process implies the deletion of irrelevant and redundant records, and the creation of derived magnitudes.
  - the *usage of metadata* to build data tables organised according to their meaning.
- *From Information to Visual Representation*: With the data already transformed into information, this step deals with the conversion of



(a) Shedroff's approach (WLSW01).

(b) Ware's approach (War04).

(c) Card's approach (CMS99a).

(d) Dürsteler and Engelhardt approach (D07a).

(e) Fry's approach (Fry04).

Figure 2.3: Diagrams of the process of InfoVis. Despite there being different perspectives, all the approaches show three main steps over four stages. Images have been extracted from (D07a) and (D07b).

information into a perceptual representation, mainly in a visual form. This perceptual representation has to display the information in such a way that the underlying patterns and structures have to be easily identified. A notational schema, which is a particular visual language that maps information into graphics, must be applied to take advantage of previous knowledge or experience of the user. There are two kinds of representations, the arbitrary conventional and the sensory. The former is the one learned over time by the receptor and its main characteristic is that it does not have any perceptive base. The latter is based on symbols and visualisation aspects that use the perceptual capacity of the brain without any previous knowledge or experience. To let the visualisation produce a perceptual impact to the user, the designer has to take into account both sensory and arbitrary conventional representations. Moreover, visual perception, cognitive psychology and even linguistics must be considered in order to provide a pleasant and understandable visual experience.

- *From Visual Representation to Understanding*: Once the visual representation has been built, it has to be given to the receptor. In order to help gain insight and build knowledge, the visualisation must allow the user to interact with it and empower the analytic discourse. In this regard, Shneiderman introduced the visual information seeking mantra: "overview first, zoom and filter, then details-on-demand" (Shn96). This mantra suggests the need for providing a general overview of the data,that can be filtered, zoomed and modified at any time so the user can get a deeper insight.

We will cover each of these steps in the following sections, in an effort to define the most important concepts as well as the state of the art of the main disciplines involved at every step.

2.3 From Data to Information

The main goal of InfoVis is to provide an effective way to generate knowledge about data. The first step to reach such knowledge is to make data accessible, and to transform it into useful structures and patterns that facilitate its access, manipulation and representation. An illustration of this stage can be seen in Figure 2.4.

The disciplines of Information Retrieval (IR) and Data Mining (DM) play a key role in the first steps of the InfoVis process. While IR deals with the representation, storage, organisation of, and access to information items (BYRN+99),

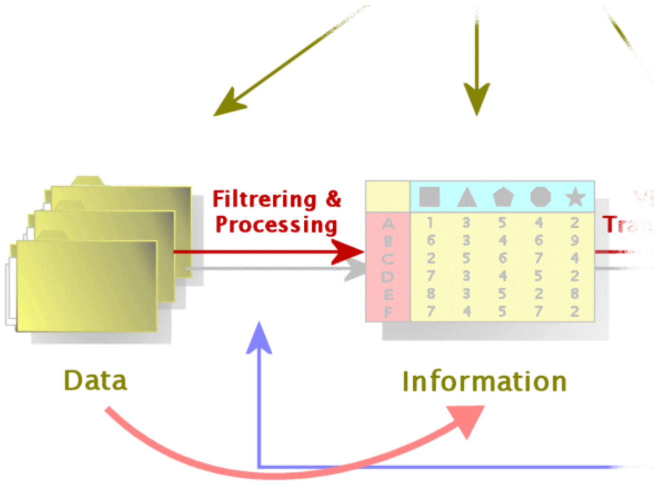


Figure 2.4: The process of converting raw data into information requires its organisation, the generation of derivative attributes and the addition of metadata to better explain it. Image extracted from (D07a).

DM is a mechanised process for identifying and discovering useful structure in data (FGW02).

From a philosophical point of view, IR and InfoVis can be understood as parallel disciplines, as both seek the acquisition of knowledge. Nevertheless, IR is more focused on the data treatment, storage and process; while InfoVis is centred on generating interactive visualisations to amplify cognition and communicate information. At this point, collected data by IR processes is analysed using DM techniques. DM comprehends techniques from artificial intelligence and machine learning aimed at providing meaningful structure to the data as well as derivative metrics to aid its organisation and classification into patterns, models or relationships.

For DM to provide knowledge, it is important to include the user in the data exploration process to combine the flexibility, creativity, and general knowledge of him/her with the enormous storage capacity and the computational power of today's computers. Infovis/VA techniques may then become crucial in providing this interaction layer which assists the user in the manipulation of the data.

We will firstly review some of the most important aspects of graph theory.

Afterwards we will explain how we can consider Web spaces as webgraphs, as well as introducing the disciplines of Web Mining and Web analytics that will provide an understanding of the type of data that can be analysed.

2.3.1 Representing Relations between Data: Graphs

A graph G can be formally described as the combination of two sets: $G = (V, E)$, where the elements of V are called vertices or nodes and the elements of E are called edges or arcs. Each edge $e \in E$ has two vertices $(u, v) \in V$ associated to it, which are called its endpoints. An edge is said to join its endpoints. An *undirected* graph is made up of undirected edges, which means that the pair of connected vertices (u, v) is not ordered. On the contrary, a *directed* graph, also known as *digraph*, is made up of directed edges, which means that the pair of connected vertices (u, v) is ordered. In this case, the first vertex is called source and the second is called target. A *weighted* graph contains edges that have an associated weight, which represents the strength of the connection between the nodes. Moreover, a *subgraph* can be defined as a graph $G' = (V', E')$, where $V' \subset V$ and $E' \subset E$. Finally, the concept of graph *walk* or *path* can be described as a sequence of connected vertices.

Let's now define some of the most important structural measures of a graph that capture important attributes according to (SNS06). Such measures can be computed locally (*i.e.* according to every node in the graph), or globally, (*i.e.* for an entire graph or subgraph). Local measures define how important a node is in the graph, and are:

Node degree corresponds to the number of edges adjacent to it. This measure can be divided into in- and out-degree. The in-degree of a node corresponds to the number of incident edges, and the out degree to the number of outward edges.

Size of the k-neighbourhood corresponds to the number of nodes within a distance less or equal to a given k value, as can be seen in Figure 2.5.

Closeness or status is the result of the summed up lengths of all the shortest paths from a node v to every other node in the graph.

Contrastatus is the sum of finite distances from all other nodes to a node v.

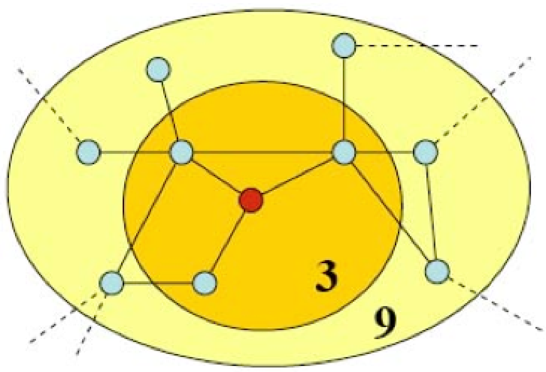


Figure 2.5: The 1-neighbourhood of the red node has 3 nodes while the 2-neighbourhood has 9.

Eccentricity is the maximum length between all the shortest paths from a node v to every other node.

Node betweenness has been defined as the number of all the shortest paths that pass through a vertex v.

Node prestige is the difference between the status and the contra status.

Cliqueishness measures how much a node appears to belong to a clique. A clique is defined as a group of nodes that is highly connected internally but loosely connected to nodes outside the clique.

The local attributes defined above can be averaged per all nodes in a graph to generate global attributes. These measures can be used to describe and provide an overview of the whole graph. Other specific global measures are:

Diameter corresponds to the largest eccentricity value, contrary to the **radius**, which corresponds to the smallest eccentricity value.

Compactness or density defines how well connected the graph is. A high compactness value means that each node can easily reach other nodes of the graph.

Tree-likeness measures the structural resemblance of a graph to a tree. As defined in (SNS06), a graph is called (p, k)-treelike if it has no more than k cross-edges and the fraction of cross-edges with respect to all edges is less than the percentage p.

An important feature of graphs, and therefore of webgraphs, is that they can be easily converted into hierarchies or trees. A tree is a mathematical structure that can be defined as an acyclic graph. That is, a graph with no walks that start and end at the same vertex. A *spanning tree* of a graph is a tree with vertices $V' = V$, having $n-1$ edges $e \subset E$.

There are many algorithms for extracting spanning trees from graphs. *Breadth First Search* (BFS) is one of the most used ones as it assures that every node will be located at the lowest possible level. It consists of a level-order traversal which starts at a given root, that is expanded to reach its *children*. Children nodes are stored in a cue and sequentially expanded in an iterative process. The complexity of this algorithm is $O(|V|+|E|)$. Another approach is the extraction of trees from weighted graphs, generating *minimum spanning trees*, which is a tree that minimises the sum of the weight of the selected edges. Prim's (Pri57) and Kruskal's (KJ56) algorithms are two of the most commonly used algorithms that deal with this problem in undirected graphs. They are greedy algorithms that run in polynomial time. For directed graphs, Edmond's algorithm (Edm67) is one of the most famous solutions. This algorithm selects the entering edge with the smallest cost for each node other than the root. If such edges generate a cycle, the algorithm tries to replace edges which have the minimum extra cost to eliminate the cycle. The order of this algorithm is $O(EV)$.

2.3.2 Characterising Web Spaces

Turetken and Sharda (TS07) defined the concept of Web space as "a collection that consists of pages that are linked to each other", *i.e.* a website, or collections more loosely connected such as the results of a web search. Thus, any Web space can be understood as a graph, also known as *webgraph* in this context.

We detail below how a Web space can be organised and understood to resemble a graph following the taxonomy presented in (TS07), which proposes three types of organisations:

- *Web space organisation based on connectivity*: This kind of organisation can also be considered as explicit, as the most common webgraph is the one formed by the hyperlink structure that exists between a set of pages. It could be, for instance, the connected pages within a website, the result pages of the search engine results, or the relation between the comments existing in online forums.

- *Web space organisation based on semantic content*: Contrary to the webgraphs formed by connected pages, other possibilities contemplate the usage of implicit relations among a set of web pages based on content similarities. The foundation of these techniques is the representation of each textual document as a vector of term frequencies, whose projection and dimensionality reduction can help to discover 'close' pages in terms of contents.

- *Other Web space organisation*: Other kinds of Web spaces consider a mixture of implicit and explicit relations. For instance, query logs can help in discovering relations based on the usage of query terms as well as browsing patterns, extracted from the pages clicked by the users when performing a query in a search engine. The resulting webgraph contains strong relations between those pages visited by many users who used the same query.

As we will see later on, in this dissertation we will mainly deal with Web spaces based on their explicit connectivity and usage.

2.3.3 Web Mining

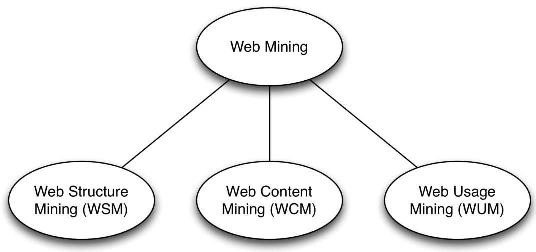


Figure 2.6: The three different types of Web Mining.

Web mining is the use of data mining techniques to automatically discover and extract information from Web documents and services (KB00). The resulting growth in on-line information required the development of powerful yet computationally efficient data mining techniques suitable for the characteristics of the Web. Web Mining techniques provide statistics and measures called web metrics that describe the characteristics of a site. These metrics can be understood as mathematical approaches to conceptualise such characteristics. Web Mining can be categorised into three main areas of interest (KB00), based on which type of data to mine from the Web: Web Structure Mining, Web Content Mining and Web Usage Mining (see Figure 2.6).

Web Structure Mining (WSM)

Web Structure Mining is the name given to the discipline of Graph Mining in the Web context. It deals with the hyperlinked structure of Web spaces, providing structural summaries in the form of metrics based upon graph theory (see section 2.3.1). Given a collection of interconnected web documents, interesting and informative facts describing their connectivity in the web subset can be discovered.

The basic tool for collecting structural information of a Web space is called crawler. Also known as spider, it is a software agent capable of traversing the Web, starting by visiting the URLs of a starting list, called seeds. As the crawler visits these URLs, it identifies the hyperlinks existing in the HTML code, and stores them in a list of URLs to visit, called the crawl frontier. URLs from the frontier are recursively visited according to a set of policies, such as URL pattern or maximum reachable depth.

Web structure metrics reflect the structural organisation of the hypertext and hence determine the readability and ease of navigation. For instance, poorly organised web sites often cause user disorientation and the so-called lost in cyberspace problem. As we will see in this dissertation, there are several approaches from the perspective of WSM that can inform webmasters or web operators of the structural organisation of their site.

Most web spaces, especially websites, are intrinsically organised in a hierarchical manner. Think about an online store such as the one illustrated in Figure 2.7. It has a home page (the root) that links to sections such as books, movies or electronics (these pages represent nodes at the first level of the hierarchy, below the root). Each category links to several detailed product descriptions (nodes at level 2).

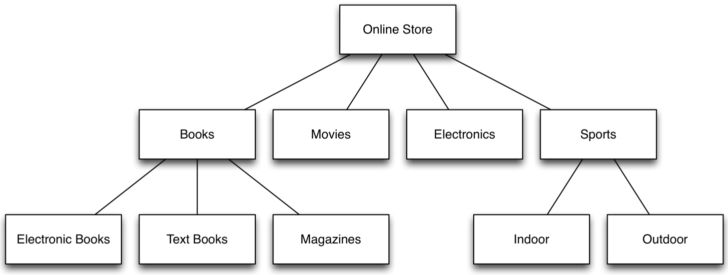


Figure 2.7: Example of the hierarchy of a typical online store.

Botafogo et al. identified in (BRS92) two main tasks to be performed in order to find an inner hierarchy of a web space:

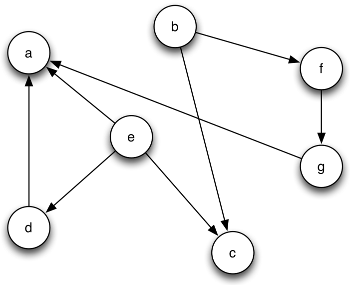
- *Identifying the root*: The most important criterion for selecting a root is by choosing a node that has a path to every node of the webgraph. Following this guideline, the home page is usually a good root, as it is commonly designed as the main entry point to the website. However, current web analytics practices also define several landing pages from a commercial point of view, which are also good candidates for representing the root of the hierarchy. In other web spaces where the root page is less clear, Botafogo et al. proposed a metric named *Relative Out Centrality*, as a measure to decide whether a node is suitable for acting as root or not. This measure is based on the *Out-Distance* measure (OD), which represents the sum of the distances from one node to the rest. Considering the distance matrix in Figure 2.8, the Out-distance corresponds to:

  $$OD_i = \sum_j C_{i,j}$$

  The Relative Out Centrality is then calculated by normalising OD by the sum of all pair-wise distances between nodes

  $$ROC_i = \frac{\sum_i \sum_j C_{i,j}}{\sum_j C_{i,j}}$$

  A high ROC coefficient may suggest that the node might be a suitable root of the hierarchy.



(a) A simple directed graph.

	a	b	c	d	e	f	g
a	0	∞	∞	∞	∞	∞	∞
b	3	0	1	∞	∞	1	2
c	∞	∞	0	∞	∞	∞	∞
d	1	∞	∞	0	∞	∞	∞
e	1	∞	1	1	0	∞	∞
f	2	∞	∞	∞	∞	0	1
g	1	∞	∞	∞	∞	∞	0

(b) Distance matrix.

Figure 2.8: An example of a directed graph and its associated distance matrix.

- *Distinguishing cross-reference links from structural links*: Structural links are the ones that clearly identify a hierarchical structure in a web space (*i.e.* the visible links in Figure 2.7). Cross-reference links are those commonly used for easing the navigation, linking relevant content. For instance, cross-reference links may link two categories or pages, or allow the user to return to the home page. Although distinguishing between these two types of links is a difficult problem to be tackled with only structural information, Botafogo et al. (BRS92) proposed the usage of a Breadth First Search Algorithm as a first approximation. Selected nodes by the algorithm are good candidates to represent structural nodes as they are located at the shortest distance to every node from the root.

Web Content Mining (WCM)

Web content data consists of unstructured data such as free texts, semi-structured data such as HTML documents, more structured information such as data stored in databases, and multimedia files. The usage of text mining techniques has been one of the most widely used approaches, allowing the analysis and further classification of contents covered in web spaces. For instance, mining textual content may give a crucial understanding into which topics are covered within a web space.

An example of a current trend in the analysis of text is opinion mining and sentiment analysis (PL08). This area of WCM has recently captured the interest of researchers. It makes extensive use of machine learning and artificial intelligence techniques for categorising pages according to the sentiment they express, such as happiness, sadness, or even frustration. This data can be of crucial importance for companies in order to measure the satisfaction of their customers through the analysis of public forums.

Content mining has often been combined with WSM. For instance, (PBY06) proposed a hybrid approach of such fields to discover pages of the same topic that are not linked within the same website, suggesting errors in the hyperlink structure.

Web Usage Mining (WUM)

Web usage mining deals with automatically generated data of trails left by users while using and navigating web spaces. This branch of web mining has been an important focus of attention during the last decade due to the explosion of the Internet and the growth of eCommerce, as we will see in the next section. eCommerce websites have pushed this area forward in an effort to improve user satisfaction through the understanding of what the users are looking for, as well as their behaviour in the site in order to discover if the hyperlinked structure is easy to navigate.

#	IP	Id	Acces	Time	Method/URL/Protocol	Status	Bytes	Referer	Agent
1	165.182.168.101	-	-	16/06/2002:16:24:06	GET p1.htm HTTP/1.1	200	3821	out.htm	Mozilla/4.0 (MSIE 5.5; WinNT 5.1)
2	165.182.168.101	-	-	16/06/2002:16:24:10	GET A.gif HTTP/1.1	200	3766	p1.htm	Mozilla/4.0 (MSIE 5.5; WinNT 5.1)
3	165.182.168.101	-	-	16/06/2002:16:24:57	GET B.gif HTTP/1.1	200	2878	p1.htm	Mozilla/4.0 (MSIE 5.5; WinNT 5.1)
4	204.231.180.195	-	-	16/06/2002:16:32:06	GET p3.htm HTTP/1.1	304	0	-	Mozilla/4.0 (MSIE 6.0; Win98)
5	204.231.180.195	-	-	16/06/2002:16:32:20	GET C.gif HTTP/1.1	304	0	-	Mozilla/4.0 (MSIE 6.0; Win98)
6	204.231.180.195	-	-	16/06/2002:16:34:10	GET p1.htm HTTP/1.1	200	3821	p3.htm	Mozilla/4.0 (MSIE 6.0; Win98)
7	204.231.180.195	-	-	16/06/2002:16:34:31	GET A.gif HTTP/1.1	200	3766	p1.htm	Mozilla/4.0 (MSIE 6.0; Win98)
8	204.231.180.195	-	-	16/06/2002:16:34:53	GET B.gif HTTP/1.1	200	2878	p1.htm	Mozilla/4.0 (MSIE 6.0; Win98)
9	204.231.180.195	-	-	16/06/2002:16:38:40	GET p2.htm HTTP/1.1	200	2960	p1.htm	Mozilla/4.0 (MSIE 6.0; Win98)
10	165.182.168.101	-	-	16/06/2002:16:39:02	GET p1.htm HTTP/1.1	200	3821	out.htm	Mozilla/4.0 (MSIE 5.01; WinNT 5.1)
11	165.182.168.101	-	-	16/06/2002:16:39:15	GET A.gif HTTP/1.1	200	3766	p1.htm	Mozilla/4.0 (MSIE 5.01; WinNT 5.1)
12	165.182.168.101	-	-	16/06/2002:16:39:45	GET B.gif HTTP/1.1	200	2878	p1.htm	Mozilla/4.0 (MSIE 5.01; WinNT 5.1)
13	165.182.168.101	-	-	16/06/2002:16:39:58	GET p2.htm HTTP/1.1	200	2960	p1.htm	Mozilla/4.0 (MSIE 5.01; WinNT 5.1)
14	165.182.168.101	-	-	16/06/2002:16:42:03	GET p3.htm HTTP/1.1	200	4036	p2.htm	Mozilla/4.0 (MSIE 5.01; WinNT 5.1)
15	165.182.168.101	-	-	16/06/2002:16:42:07	GET p2.htm HTTP/1.1	200	2960	p1.htm	Mozilla/4.0 (MSIE 5.5; WinNT 5.1)
16	165.182.168.101	-	-	16/06/2002:16:42:08	GET C.gif HTTP/1.1	200	3423	p2.htm	Mozilla/4.0 (MSIE 5.01; WinNT 5.1)
17	204.231.180.195	-	-	16/06/2002:17:34:20	GET p3.htm HTTP/1.1	200	2342	out.htm	Mozilla/4.0 (MSIE 6.0; Win98)
18	204.231.180.195	-	-	16/06/2002:17:34:48	GET C.gif HTTP/1.1	200	3423	p2.htm	Mozilla/4.0 (MSIE 6.0; Win98)
19	204.231.180.195	-	-	16/06/2002:17:35:45	GET p4.htm HTTP/1.1	200	3523	p3.htm	Mozilla/4.0 (MSIE 6.0; Win98)
20	204.231.180.195	-	-	16/06/2002:17:35:56	GET D.gif HTTP/1.1	200	3231	p4.htm	Mozilla/4.0 (MSIE 6.0; Win98)
21	204.231.180.195	-	-	16/06/2002:17:36:06	GET E.gif HTTP/1.1	404	0	p4.htm	Mozilla/4.0 (MSIE 6.0; Win98)

Figure 2.9: Example of a log file in ECLF.

Every time a user clicks on a link in the Web, the web server responsible for the linked content receives a request that is recorded in a log file. Therefore, log files contain trails from users activity that can be studied in order to understand their behaviour. One of the most common log file standards is the CLF (Common Log Format), which stores data such as the URL of the requested page, an identification of the user (*i.e.* the ip of its machine) and information on how the page was requested, such as the status code of the response and the method used to make the request. A more detailed format is the Extended Common Log Format (ECLF) that can be seen in Figure 2.9. It is an extended version of the CLF that also contains information regarding the page where the user came from after visiting the requested content; and detailed description of the agent, which is the OS and browser of the user.

Raw data from log files require some preprocessing to convert it into useful information. Cooley (CMS99b) defined the steps involved in this process, that can be seen in Figure 2.10:

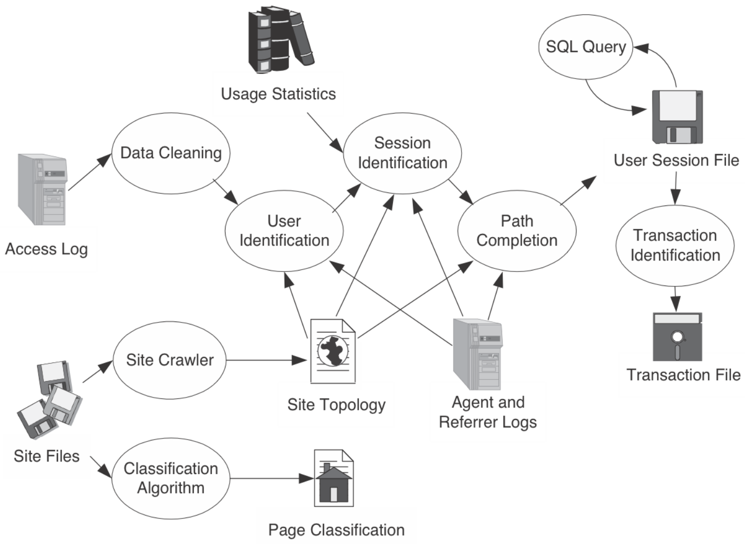


Figure 2.10: The WUM preprocessing process proposed in (CMS99b).

- *Data cleaning*: This step requires the removal of useless data existing in log files. The HTTP protocol requires a separate connection for retrieving embedded content in a web page. However, although in specific cases, users' behaviour analysis does not need information regarding requests to images or multimedia content. Therefore, such records are usually deleted.

Furthermore, to avoid misinterpreting users' behaviour, it is important to remove records regarding web crawlers (TK02). Many of these automatic agents cache web content to feed web search engines databases. They can usually be detected through the access of many pages in a very short amount of time. Moreover, some crawlers provide data that can be used in the agent field of the log file. In this regard, there are lists of repositories of agents that identify web crawlers[1].

- *User and session identification*: The stateless nature of the HTTP protocol presents a set of problems related to the user and session identification. First, users can not be uniquely identified as dynamic ip services provided by ISPs may give the same ip to more than one user. Furthermore, the presence of proxies may also leave trails with the same ip to more than one user. There are several heuristics to overcome this problem: the first one assumes that two users with the same ip may be distinguished through their user agent. Another approximation is the study of the paths of such users using site topology. Therefore, if there are several accesses to unconnected pages, it is likely that these accesses come from different users. Once users are identified, it is also important to split their trails into sequences of pages that compound sessions. Split web accesses from users into sessions requires the understanding of when they left the site. In that sense, time heuristics are used, considering a time gap of about 25 minutes (CMS99b) to split page sequences into different sessions. However, the reader should note that the diversity of web pages have dramatically changed the way a site is navigated as well as the amount of time needed to consider that two accesses belong to different sessions. Hence, every website should investigate for this particular heuristic for the WUM methods to be effective.

- *Path completion*: Once user sessions have been identified, they should be analysed in order to check their completeness. Again, due to the stateless nature of the HTTP protocol, and to the existence of browsers caches and the back button, there might exist user sessions with a sequence of visits not connected to each other. Therefore, the website structure can be used to check the validity of the path and to infer possible backtracks performed by the users. According to (CMS99b), it is very unlikely that these non sequential visits may

[1]http://www.user-agents.org/

come from a user using bookmarks or writing the direct URL in the browser.

- *Formatting*: Finally, processed data in the form of user sessions with path completion should be stored in a proper manner for the WUM algorithms to work efficiency.

Some of these problems can be overcome, however, with the use of cookies. Cookies are a javascript-based technique that leaves light text files in the user's computer that are detected every time the user visits a site. Using a unique id, cookies enable the identification of the user providing more realistic data about users behaviour. Moreover, another advantage of this technique is that web operators may customise the information stored in such files according to the analysis needs of the site, such as incorporating information from commercial campaigns. However, this methodology fails when users clean the browser's cookies, or when more than one user actually uses the same computer. Moreover, one of the main drawbacks is that every single page or template, on a site must be tagged in order to incorporate the cookie code. It has been observed that the more frequently a website's content changes, the more prone the site is to missing page javascript tags (LT07).

Once usage data has been collected, preprocessed and formatted, there are many useful statistics that can be computed by counting existing records in the clean logs. For instance, the number of page visits or unique sessions may provide valuable information on the most relevant contents existing in a website. These metrics can be used for many purposes. Facca et al. identified four main interrelated applications for WUM (FL05):

- *Personalisation of web content*: Personalisation techniques analyse website activity in order to predict and anticipate user behaviour. Hence, through the analysis of previous visits the system may automatically provide links with related content (or products) consumed previously by the user (WR09; PPPS03).

- *Pre-fetching and caching*: By predicting and anticipating the most important contents in a site during a certain time or period, web operators may develop pre-fetching and caching strategies to reduce the server response time (CC02; LBOT00).

- *Support to the design*: Usability is one of the main issues in the design of websites. WUM techniques and measures may help to better understand users' behaviour in order to evaluate if the overall hyperlink structure facilitates the navigation enabling users to reach what they are looking for (SP01). Current examples analyse frequent sequences (Ber02) as well as propose new and improved hyperlink structures according to usage patterns (HSW07).
- *Support market strategies*: Business specific strategies can be supported through the extensive usage of WUM techniques to measure customer attraction, customer loyalty or cross sales (Kau07; LT07). In fact, most of the outcomes of research in WUM are mainly applied in this community,

2.3.4 Web Analytics and Search Engine Optimisation

The disciplines of Web Analytics and Search Engine Optimisation are quite new disciplines co-existing in the so-called Online Marketing. These techniques take advantage of research efforts of the Web Mining community to improve the success of websites and online campaigns.

The exploitation of WUM metrics and algorithms has been mainly applied in the Web Analytics domain, whose commercial interests have pushed forward the application and development of new techniques to measure the success of websites and marketing strategies.

The Web Analytics Association[2] (WAA) defined web analytics as *"the objective tracking, collection, measurement, reporting, and analysis of quantitative Internet data to optimise websites and web marketing initiatives"*.

There are three types of metrics: counts, which tend to be WUM metrics that sum up data, like number of visits of the site; ratios, which better help to identify trends such as bounce rate; and Key Performance Indicators (KPIs), which are counts and ratios infused with business strategy and, therefore, are specifically targeted to evaluate the success of online strategies. Web metrics can be considered in an aggregate way, taking into account all the users together; segmented, considering only a subset of the users; or individually, considering individual users. Counts and ratios turn into KPIs when applied to very specific universes of users and in specific types of web. Some examples of widely used KPIs are:

[2]http://www.webanalyticsassociation.org

- *Conversion Rate*: It is the percentage of visitors who reach the goal of the website, like generating a lead, selling a product, or registering a user for a newsletter.

- *Unique Authenticated Users*: It is the number of actual individual people who visit at least one page on a site. More effort should be made to avoid multiple counting of single users registered more than once as well as multiple users using the same registration.

- *Impression*: It is the number of times an advertisement or promotion is delivered to a site visitor.

Search Engine Organisation (SEO) is a parallel discipline to Web Analytics that also takes part in the wide domain of Online Marketing. SEO techniques pursue the increase of web presence of a website through its positioning in the Search Engines Results Page (SERP). Web search engines provide results according to a user query in an ordered manner according to a Page-rank algorithm (BPMW98). In this algorithm the Web structure as well as the textual content of every single web page on the Web is taken into account to provide relevant results to a given query.

SEO's techniques adapt website contents and hyperlink structure to increase their page-rank in order to appear in the highest position possible, as the higher a website appears in a SERP, the more possibilities there are to being visited (ESSF09).

Since the definition of the concept of page-rank (BPMW98), every web search engine (Google, Yahoo!, Bing, ...) has modified the original algorithm in order to improve its results. One important approach for improving the accuracy of the results is the usage of website structure to better understand the offered topics. The automatic understanding of the categories and subcategories of web page (see example of Figure 2.7) provide detailed information of contents offered in a website. However, the continuous changes in these algorithms make SEO a very vivid discipline, forcing experts to continuously test new techniques to satisfy search engine page-rank policies and therefore, appear in higher positions in results lists.

There is also a version of the SEO discipline applied to the positioning through paid strategies. This discipline is called Search Engine Marketing.

2.4 From Information to Visual Representation

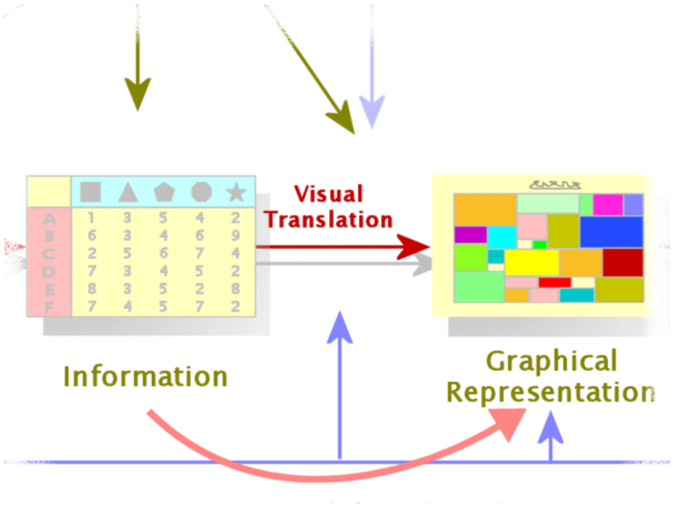


Figure 2.11: The process of converting information into a representation requires a visual translation from information to a graphical representation. Image extracted from (D07a).

Sight or 'vision' is the human sense most closely linked to cognition (War04). Accordingly, the increasing need for understanding large datasets has also intensified the interest in developing new visual translations to assist in the conversation of information into an intuitive graphical representation, expressed by the schema of Figure 2.11. Abstract data requires new approaches that are different from techniques used in geographical information. In this section we present some of the most important aspects of graph drawing and describe some of the most relevant visual metaphors for visualising hierarchies, which will be extensively used in the following chapters.

2.4.1 Graph Drawing

Graph drawing algorithms have had a great impact in the InfoVis community, as there are many problems that can be expressed as a set of connected items. Research in graph drawing deals with the automatic placement of nodes in the space in an aesthetic way. Some standard aesthetics criteria are the minimisation of edges crossings, the avoidance of node overlapping and the minimisation of edge bends (WPCM02).

Graph drawing techniques can be classified according to the type of layout they generate. We will follow the taxonomy presented in (DBETT98) to define them:

Orthogonal layouts. Orthogonal layouts (Figure 2.12) rely on polylines that map a graph's edges into horizontal and vertical segments. Eiglsperger et al. (EKS03) proposed a methodology called topology-shape-metrics, whose generation process consists of three steps: planarisation, orthogonalisation and compaction. The planarisation step inserts dummy nodes into the system which represent crossings to convert non-planar graphs into planar ones. The orthogonalisation step determines the angles and the bends in the drawing using only multiple of 90 angles. The compaction step removes the dummy nodes and tries to minimise the length of the edges to reduce the drawing area.

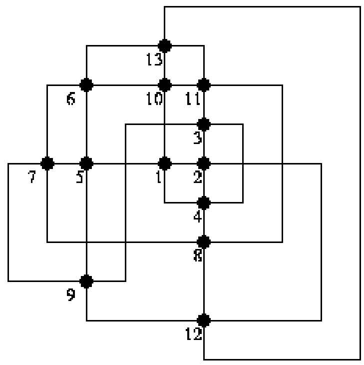


Figure 2.12: Example of orthogonal drawing.

Force directed layouts. Force directed layouts (Figure 2.13) are one of the most popular layouts for drawing graphs. They use a physical model where nodes behave as repulsive forces and edges between nodes behave like springs. Hence, nodes are placed in the space by iteratively calculating forces that apply to every node, making this kind of algorithm very time consuming for large graphs as the complexity tends to be n^2. However, this iterative approach enables the user to create smooth animations that show the movement of every node to the destination point. The most common algorithms for calculating force directed layouts are Fruchterman-Reingolds (FR91) and Kamada-Kawai (KK89). Besides the computational complexity of these algorithms, their main drawback is that they are not deterministic, leading to different configurations at every run of the algorithm.

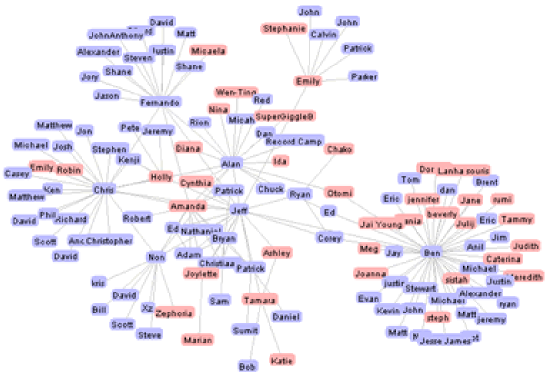


Figure 2.13: Example of a Force Directed Layout from the Prefuse toolkit (http://prefuse.org/).

Hierarchical layouts. Hierarchical layouts (Figure 2.14) compute node positions for directed acyclic graphs. Sugiyama introduced in (STT81) his method based upon three main steps: determine the layering for the nodes of the graph according to its topological properties, finding nodes order within each layer to reduce edge crossings, and assigning final node positions.

Circular layouts. The circular layout (DMM97) (Figure 2.15) locates nodes of the graph in the edge of a circle. The main drawback of this visualisation is the number of edge crossings that it generates. However, this method has become popular for the visualisation of social networks and usually incorporates interaction techniques to stress connections between selected nodes on the perimeter.

Topological layouts. The topological layout introduced in (AMA07) computes topological features of the graph to detect possible existing different substructures. The system applies the best possible layout for

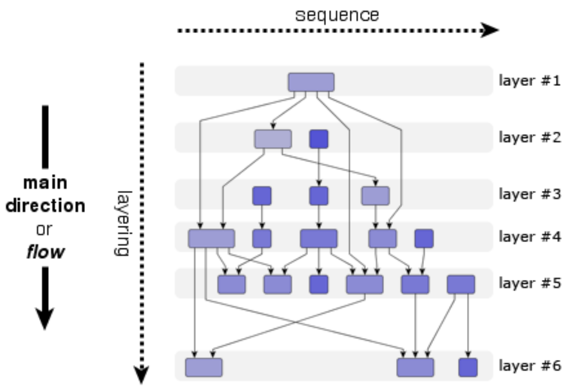


Figure 2.14: An example of the Sugiyama layout showing the different layers detected by the algorithm.

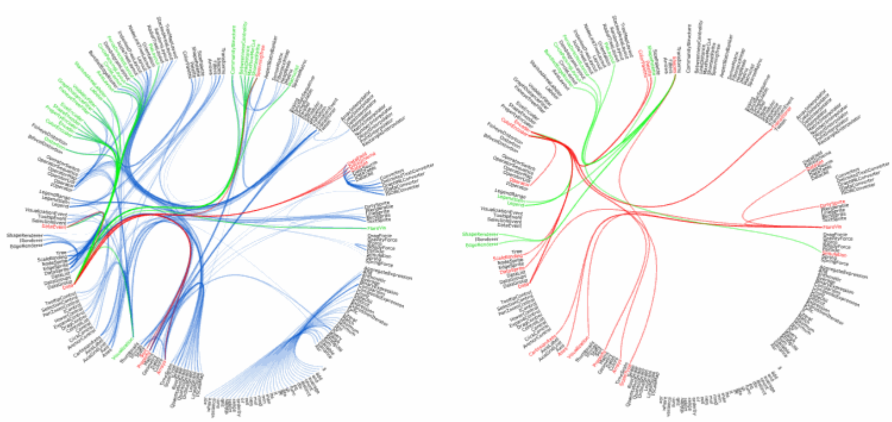


Figure 2.15: Example of a circular layout from the Flare toolkit (http://flare.prefuse.org/). Relations are usually hidden (right image) to stress specific data relations.

depicting every substructure. The main drawback of this techniques is that it needs to analyse the graph beforehand.

Tree layouts are also part of this taxonomy. Due to their importance to this work, we will discuss them in detail in the following section.

2.4.2 Tree Layouts

In the previous section we have introduced hierarchical layouts. This kind of visualisation displays graphs in a tree-like form, placing nodes in a vertical or horizontal manner and minimising the edge crossings by ordering the nodes. However, this technique generate many edge crossings in very dense graphs, as can be seen in examples from (STT81).

Following the discussion introduced in Section 2.3.1, there are several approaches for converting graphs into trees, based upon the extraction of meaningful or structural edges from the graph. Extracting trees from hierarchies reduces the complexity of the algorithms as trees are acyclic and planar graphs that can be displayed without edge crossings. We will now review some of the most popular and relevant techniques for laying out hierarchies:

Classic tree drawings. Algorithms for drawing classical trees are the most popular methods for drawing hierarchies, as they resemble typical vertical trees appearing in many examples, such as in organisation charts. Reingold and Tilford (RT81) proposed an algorithm for drawing ordered binary trees upwards from bottom to top. The x coordinate of each node is set to an arbitrary value, while the y is determined by its depth on the tree. The position of every parent is then defined by the average of the x coordinate of its children. Finally, each subtree is drawn independently. After this process, the right subtree is shifted so that it is placed as close as possible to the left one. Walker (WI06) extended the Reingold and Tidier algorithm to trees with an arbitrary degree that computes well balanced layouts (Figure 2.16). Nodes are traversed from left to right, while corresponding subtrees are placed and shifted as done by the binary algorithm. In a second step, the nodes are traversed from right to left taking average positions of the subtrees.

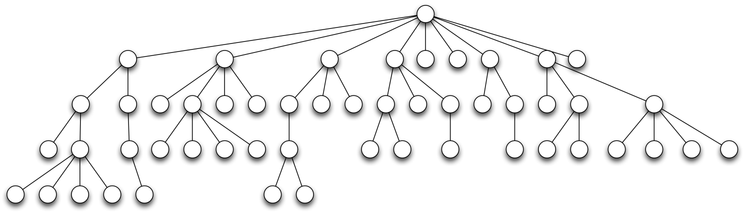


Figure 2.16: Example of a classic tree layout.

The main drawback is that as the number of nodes increases, classic tree layouts use log(n) vertical and n-squared horizontal screen real estate (Eic01).

Space filling techniques. Johnson and Shneiderman (JS91) introduced one of the most famous algorithms for hierarchical drawing: the *treemap* (Figures 2.17 and 2.18). The main feature of treemaps is that unlike most other methods, they use a space filling approach that occupies all the available drawing area. The initial rectangle represents the root node of the hierarchy. This rectangle is partitioned into as many rectangles as children nodes the root has. The algorithm continues partitioning the rectangles recursively, until reaching leaf nodes. Figure 2.17 is an example of how to convert a hierarchy into a treemap.

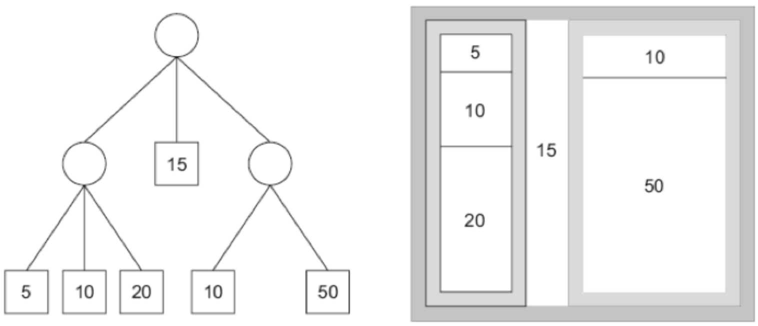
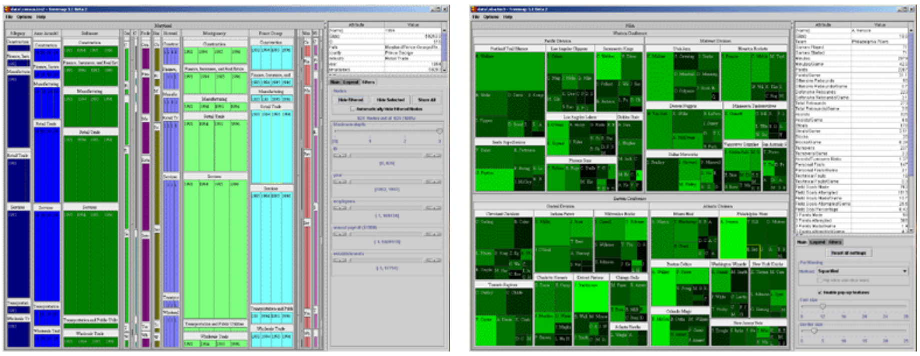


Figure 2.17: Converting a hierarchy into a treemap.

Usually, partitioning algorithms use the size of each rectangle according to a defined weight attribute of the representing node. There are two main algorithms for partitioning the rectangles: Johnson and

Shneiderman (JS91) presented the "Slice and Dice" algorithm (Figure 2.18(a)), which proposes to change the direction of the partition of the rectangles alternatively between vertical and horizontal slices. However, this approach leads to very thin and long rectangles in deep branches of the tree.

Bruls et al. (BHVW00) proposed the "squarified treemap" method (Figure 2.18(b)) in an effort to improve the slice and dice algorithm. These algorithms generate rectangles which are more square-shaped making partitions that are neither horizontal nor vertical, but a combination of both.

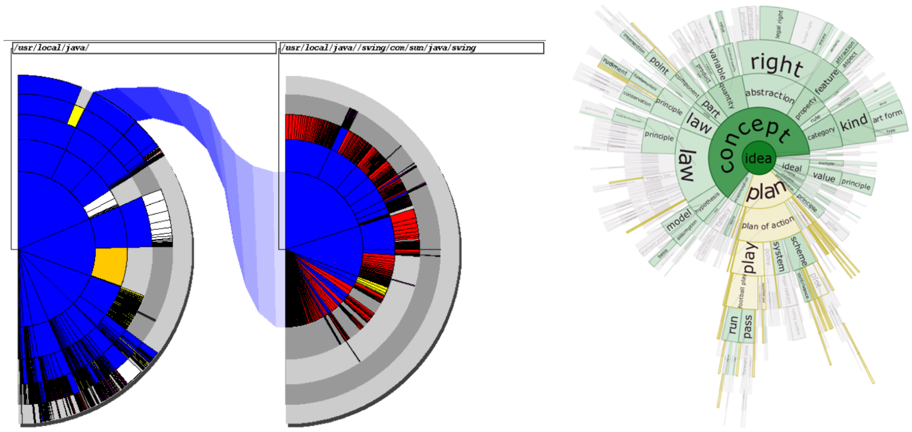


(a) Treemap with the slice and dice algorithm. (b) Treemap with the squarified algorithm.

Figure 2.18: The slice and dice approach may produce less comprehensible representations as it generates.

Information slices (AH98) (Figure 2.19(a)) is another space filling technique presented by Andrews and Heidegger. This technique uses one or more semi-circular discs to compactly represent hierarchies in a two dimensional space. Each disc represents several levels of the hierarchy. Deeper levels are shown using cascading series of discs. At each level of the hierarchy, the children are fanned out in the available space according to the total size of each child. The space dedicated to each node of the graph depends directly on a weight defined by the user, like in the treemaps. This technique also allows the navigation through the hierarchy by clicking on inner nodes. Clicking on a node fans out the children of the node on the adjacent disc.

Another space filling technique is *sunburst* (Figure 2.19(b)), presented



(a) Example of the information slices tree visualisation.

(b) Example from the software DocuBurst showing a Sunburst visualisation from Wordnet

Figure 2.19: Radial space filling visualisations.

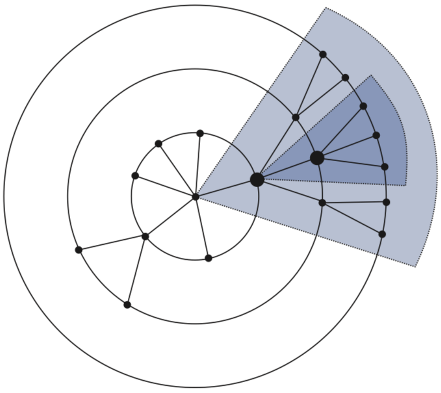
by Stascko and Zhang (SZ00). Their approach uses a radial layout, having the root node at the centre of the visualisation and deeper nodes placed further away from the centre. Analogously to the treemap, the angle of every node is defined according to a defined node property. Two main benefits of this technique are that it provides more space to show information about non-leaf nodes than the traditional treemap display does, and it enables us to easily discover leaf nodes. An implementation of DocuBurst was used to improve the usability and utility of the WordNet database (CCP09).

Finally, Schulz et al. (SHS10) surveyed the most important space filling techniques in an effort to characterise the design principles that lay behind some of these techniques, stressing three main strategies:

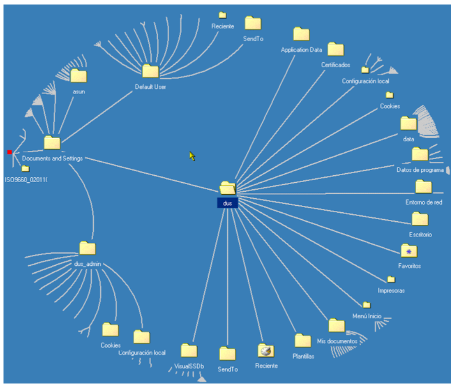
- seek unexplored regions within the existing design space.
- establish new connections between different regions of the design space
- find a novel parameterisation of an otherwise fixed design

The authors state that, by mixing and parameterising such strategies, any space-filling technique can be created.

Radial techniques. The *radial tree* was introduced by Eades in (Ead92). A focus node (generally the root of the tree) is placed in the centre of the display. The remaining nodes are placed on concentric rings around it, according to their depth in the tree. A subtree is then laid out over a sector of the circle. The angular position of a node on its ring is determined by the sector of the ring allocated to it. Each node is allocated a subsector of the sector assigned to its parent, with size proportional to the angular width of that nodes subtree as can be seen in Figure 2.20(a).



(a) Sectors assigned to different nodes in the radial tree algorithm.



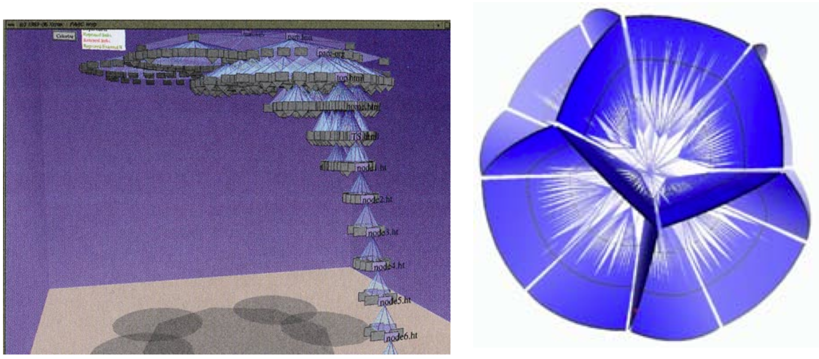
(b) Example of a hyperbolic tree.

Figure 2.20: The radial tree (left) provides a clear and easy way to interpret structure, but cannot handle large graphs. On the contrary, the hyperbolic tree (right) provides a more obscured structure, while enabling the visualisation of very large graphs.

The main drawback of the Radial Tree is that it does not take advantage of the available space in the display, with the possible generation of areas with a high density of nodes, and others with no density at all.

Sunburst (Figure 2.19(b)) is a space filling extension of radial trees.

3D approaches. *Cone trees* (Figure 2.21(a)), introduced by Robertson et al. in (RMC91), represented the first approach to representing hierarchies in using three dimensions. The tree is built top down: the root is placed at the top-most centred position, while its children are located in a deeper level in a circular way, creating a cone shape. Cone trees were conceived with an interaction technique that rotates the cone according to user clicks, bringing the selected node to the front of the display. Rotations at each substructure are made in parallel using animations to avoid the user's confusion. Illumination techniques are also used to increase the realism of the visualisation. In that sense, shadows are projected on a floor placed under the tree, where darker shadows imply denser branches in the hierarchy.



(a) Example of a cone tree.

(b) Example of a polyplane drawn with the dodecahedron polytope (30 subplanes).

Figure 2.21: 3D approaches for visualising hierarchies.

This polyplane visualisation (Figure 2.21(b)) was proposed by Hong and Murtagh (HM04) and allows the visualisation of large and complex networks in three dimensions. The algorithm uses the concept of subplanes which are defined using regular polytopes. Every subplane contains a subtree of the whole graph. Example from Figure 2.21(b) represent a polyplane built with the dodecahedron polytope.

Focus+Context techniques The most representative example of focus plus context techniques for drawing hierarchies is the *hyperbolic tree* (Figure 2.20(b)). This technique was first introduced in (LRP95) and extensively studied by Tamara Munzner (MB95). The hyperbolic tree helps to represent large trees, using distortion techniques to modify the space, so a focus area is given more space and, therefore, can be seen in more detail. Taking into account a graph drawn on a hyperbolic disk, the space grows exponentially with the distance to the centre, providing more space to represent large graphs. Segments in the hyperbolic space are exponentially smaller in the Euclidean plane when approaching the perimeter.

The main drawback of this technique is that it requires interaction methods for exploring and navigating the graph, as it does not provide a clear overview of the whole structure.

2.5 From Visual Representation to Understanding

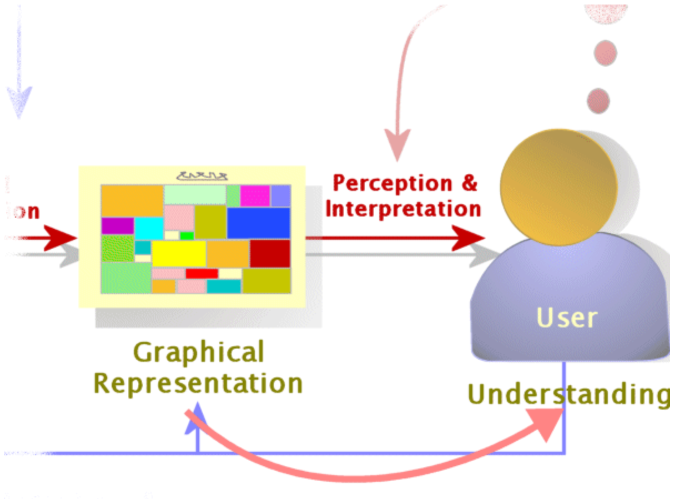


Figure 2.22: The process of converting the visualisation into understanding. Image extracted from (D07a).

The last step of the InfoVis process deals with the generation of insight. While the generation of a visualisation helps to conceptualise a dataset in a visual manner to take advantage of perception capabilities of our brain, interaction techniques enable the user to manipulate the data, generating new perspectives that may even solve problems the user never thought he/she had. Figure 2.22 represents an schema of this stage of the InfoVis process.

2.5.1 The Interaction Process

One of the main characteristics of the information visualisation field is that visual representations are accompanied by interaction techniques that enable the user to explore the represented data with the intention of generating new views that help him/her to discover outliers or relevant patterns. Interaction techniques may assist the navigation through large spaces, filtering undesired data or select an adequate level of detail.

Ware (War04) pointed out the importance of the role of interaction in visualisations to aid the generation of knowledge and identified three main loops that take place during the interaction process:

Data manipulation loop. In this loop, objects are selected and moved using the basic skills of eye-coordination. For instance, placing the mouse on top of an object may provide extra and relevant information. This action is called “hover query”.

Navigation and exploration loop. Several tasks to browse the information can be accomplished in this loop. The navigation through a 3D space, the usage of zoom, focus plus context techniques, distortion techniques or linking and brushing are methods that provide visual feedback that allow us to get insight into the data.

Problem solving loop. This loop represents the highest level actions. In this loop, the user forms hypothesis about the data and refines it them by repeating the two previous loops.

Shneiderman also proposed the InfoVis mantra (Shn96), which has become one of the most followed rules used in current InfoVis systems: “overview first, zoom and filter, then details-on-demand”. This mantra suggests that any visualisation system should first provide an overview of the data to first locate potential patterns or outliers. Then zooming and filter interactions may help the analyst to focus on a specific part of the data to end up by gathering details of display items.

Proper interaction techniques with low time response may be very beneficial in terms of answering user questions.

2.5.2 Interaction Techniques

As a previous step to define interaction techniques, it is important to distinguish them from distortion techniques. On one hand, and according to Keim (Kei02), interaction techniques allow the data analyst to directly manipulate the visualisations and dynamically adapt them according to the exploration objectives. On the other hand, distortion techniques provide a means for focusing on details while preserving an overview of the data. Therefore, distortion techniques can only be considered interaction techniques when they are used in a dynamic form.

We hereafter present some of the most popular and more widely used interaction techniques:

Direct manipulation Perhaps one of the most popular interaction techniques, was introduced by Shneiderman in (Shn81). Direct manipulation tries to resemble a physical world enabling users to select or drag display items producing an immediate visual feedback. This technique is usually integrated in node link diagrams, allowing users to drag nodes around in order, for instance, to separate a relevant set of data.

Scrolling and panning Visualisations can be understood as viewports to a visual space. Scrolling lets us move a horizontal or vertical scroll bar that moves the viewport, allowing the visualisation of other parts of the space. Panning provides the same effect, but instead of moving scrolling bars it lets us drag the whole visual space directly.

Interactive filtering This technique enables the user to dynamically select the important data segments to be displayed in the visualisation. Hence, it enables us to simplify large datasets by hiding non-relevant information. Relevant data segments can be directly selected by the user, or expressed with dynamic queries (AWS92). Typical dynamic query objects are sliders that can be set accordingly to satisfy a data range to be visualised. Visual feedback is provided by showing or hiding visual elements that match the range selected by the user at the same time that the user moves it.

Interactive zoom This technique can be divided into geometric zooming and semantic zooming. Geometric zooming graphically scales the visual item, providing more screen space to observe it. This is particularly interesting when having very overlapped or dense layouts. Semantic zooming may also graphically scale the display items, but modify their visual properties according to semantic properties. For instance, semantic zooming may provide textual information inside nodes when reaching a certain scale, or may even modify object shape to depict values from other object attributes.

Distortion techniques The concept of distortion techniques relies on the deformation of visualisations to improve the readability of certain parts by dedicating more display space to them and compacting the rest. Some distortion techniques are usually called focus+context techniques as they allow the user to have a very detailed view of part of the visualisation while maintaining the context. The hyperbolic tree (MB95) is an example of a visualisation whose usefulness depends of these kinds of techniques.

Linking and brushing Linking and brushing enables the user to combine different visualisations of the same data providing multiple coordinated views. Any changes performed in one of the visualisations should immediately affect the rest of the coordinated views, allowing the user to examine the dataset from different perspectives. Highlighting is one of the most popular techniques used in linking and brushing, which stresses selected elements in all the available representations.

2.6 Evaluating Interactive Visualisations

The evaluation of interactive visualisation systems is one of the top unresolved problems of InfoVis/VA (Che05). While the use of visual and interactive techniques have been shown to be valuable at enhancing the acquisition of knowledge (War04), most of them are still too novel for many users who are still struggling to use simple business graphics effectively (Pla04). The main goal of evaluation is to understand how and why a specific visualisation helps users to gather insight, providing information to improve them and present actionable evidence of measurable benefits that will encourage more widespread adoption (Pla04).

Capturing and understanding the exploratory nature of InfoVis techniques is one of the main challenges in assessing InfoVis/VA systems. Therefore, evaluation methodologies must go beyond the mere understanding of the ease of interpretation of a representation, and must support the discovery of working practices that lead to insight. At the same time, they have to help to understand the context where each technique must be applied.

Usability testing has been a fertile research area of Human Computer Interaction that has broadly influenced current evaluating practices in InfoVis/VA. Several methodologies exist that may also be applied to the InfoVis/VA environment. However, usability issues of InfoVis have tended to be addressed in an ad-hoc manner and limited to particular systems (Che05). Andrews (And06) classifies testing methods into three categories:

Formative testing. Also known as Thinking Aloud Testing, this qualitative analysis consists of asking separately a small set of users to use the visualisation and verbalise their thoughts while interacting with the system. This test can be useful in order to rapidly identify misleading artefacts and bugs on the visualisation. Nevertheless, thinking aloud tests are rarely generalisable to other visualisations because they are heavily focused on specific tasks provided by the software. The main drawback of this technique is that users thinking aloud change their behaviour and are slower (ES93), discouraging the use of this methodology for measuring usability attributes such as effectiveness or task completion time. Other evaluation methodologies considered formative include cognitive walkthroughs (TM04), expert reviews (TM05), and heuristic evaluations (ZSN+06).

Summative testing. Also called formal experimentation or quantitative evaluation, this methodology is about collecting statistical measures that define how well the user performs a set of tasks. This methodology is usually applied when comparing different interfaces or designs. In this case, the number of users is a key point for the reliability of the results. Summative evaluations can be designed in two ways: by using independent groups of users (also called between groups experiments), each one testing a system, or having only one group of users that will use all the available interfaces (also called within groups experiments). The main drawback of between groups experiments is the possible bias introduced due to possible differences between the groups. Regarding within groups experiments, their main drawback is the possible learning effect that the use might accumulate due to the fact that they repeat the same tasks with different interfaces. This methodology is especially useful when conducting evaluations at the component or system levels (TC05). The former involves the evaluation of specific interface designs, interaction techniques or new visual metaphors. The latter involves the integration of several components, and is targeted at discovering learnability and utility of a system.

Usage Studies. These kinds of studies are performed at the working environment level (TC05) and started to gain importance with provoking articles such as (Pla04), when Plaisant challenged researchers to reformulate their evaluations to capture the exploratory nature of InfoVis systems. Lately, Plaisant and Shneiderman formalised the concept of In-depth Long-term Case Studies (MILC) (SP06), in an attempt to overcome some of the main issues involved with traditional HCI methods presented above. This methodology enables us to conduct more naturalistic evaluations with qualitative assessments based on the observation of real users engaging with real tasks using the target tool during a long period of time. However, this methodology is very time consuming and requires very robust and usable systems which goes beyond research prototypes. There are not many ethnographical studies using the MILC methodology. Perer and Shneiderman showed the benefits of the Social Action system for analysing social networks in (SP06; PS09).

Munzner identified in (Mun09) a nested model made up of four levels at which an InfoVis system can be evaluated. Such levels are:

1. *Domain level*: evaluations should aim to identify if the target problem is, in fact, a real problem to users.

2. *Abstraction level*: this level refers to problems related to the abstraction model of the problem. Thus, it involves the assessment of the data types and operations related to the main problem identified in the domain stage.

3. *Encoding and interaction design*: this level deals with the assessment of the visual and interaction techniques available, assessing its usefulness and adequacy to the problem to solve.

4. *Algorithm level*: once the main problem to be solved has been identified, as well as its abstraction model and visual encodings, the developed algorithms to create the visualisations and the interactions have to be evaluated to assure their usefulness and scalability.

One of the goals of this dissertation is to report evidence of the usefulness of the visual and interactive approaches of our interactive system, to support the explorative analysis of Web spaces. To do so, we have conducted several studies to evaluate our tool at the four levels mentioned above. In each evaluation, we will stress which stage has been covered to help the reader to follow the whole assessment of our system.

Chapter 3

WET: A System to Support Visual Data Mining

This chapter introduces the architecture of an information visualisation system that has been used as a platform to develop and test visualisation and interaction methods to enable the exploration of Web spaces. We will introduce its architecture as well as present the main features of its user interface.

3.1 System General Overview

The Website Exploration Tool (WET) was originally conceived as a platform to evaluate some of the most relevant research developed regarding the visual analysis of web data (see Section 4.2), aimed at improving state of the art methodologies and providing a further evaluation. Nevertheless, we ended up developing a complete Infovis architecture that supports the easy prototyping of new visual (and mainly hierarchical) approaches, that can be applied to a wide range problems.

The proposed architecture, presented in Figure 3.1, is built upon two main building blocks: the Data Management and Mining System (DMMS) and the Visualisation System (VS).

The DMMS is responsible for gathering, preprocessing and mining raw data to generate derivative magnitudes that can be explored afterwards with the

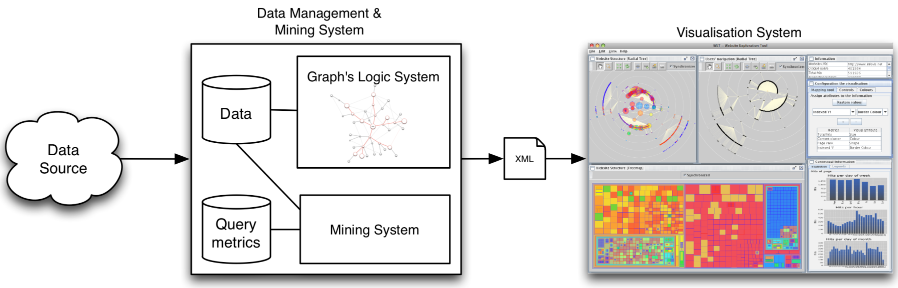


Figure 3.1: The WET's pipeline is based upon two main blocks, the Data Management and Mining System and the Visualisation System. The first is in charge of processing the data extracted from the data source to generate a portable xml file that can be interpreted and visualised by the latter.

VS. As seen in Figure 3.1, this system is made up of two databases. The first one contains the information of the graph structure, by populating two tables (nodes and links), describing the connections between the nodes. Also, any metadata associated to the nodes can be integrated in the system, by referencing the node ids, stored in the first database, allowing the system to link both data sources.

The main characteristic of the architecture of WET is that the data attributes (metadata) are calculated using queries stored in a second database. Such queries, formatted in a proper way, are automatically executed in the main database. The queries database is made up of three main tables, each one containing queries able to extract metrics from either nodes, links or regarding the whole graph. The Mining System is responsible for managing the database that contains those metrics, and executing them in the main database as illustrated in the Figure 3.2. It outputs a XML file formatted with the GraphML format[1], that contains the graph structure plus the metadata created with the queries. Moreover, the Mining System provides classes for accessing the main database with the data, to allow the development of more advanced mining methods difficult to be expressed with SQL statements.

As we have seen, the main feature of this system is that it can easily integrate any database containing relevant data to the problem, with the only requirements of filling the nodes and links tables, as well as the query met-

[1] http://graphml.graphdrawing.org/

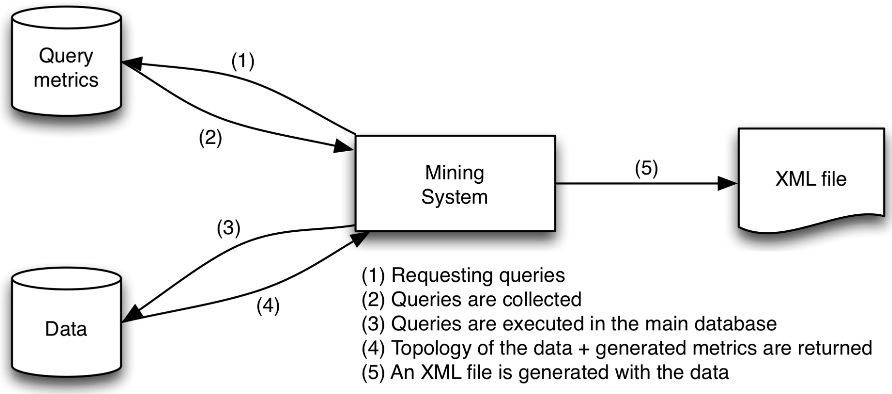


Figure 3.2: The web mining system of the DMMS is based on a Java based GraphML builder that applies SQL queries to extract web metrics over the data stored in the DM database.

rics tables for calculating derivative data attributes. As we will see in the following chapters, use cases demonstrate the usefulness of this architecture to ease the data integration and derivative metrics generation. However, it is very important to notice that the system requires a fine tuning for handling large datasets, based upon indexes that speed up the execution of queries.

Regarding the visualisation system, it was conceived as a platform to visualise and explore data, enabling the customisation of the visualisations as well as providing a set of rich interactive tools to empower the analytic discourse (TC05), which is a *dialogue between an analyst and this information to produce a judgement about an issue.* We will further discuss the visualisation elements in Section 3.3.

3.2 Dealing with Large Graphs: the Graph's Logic System

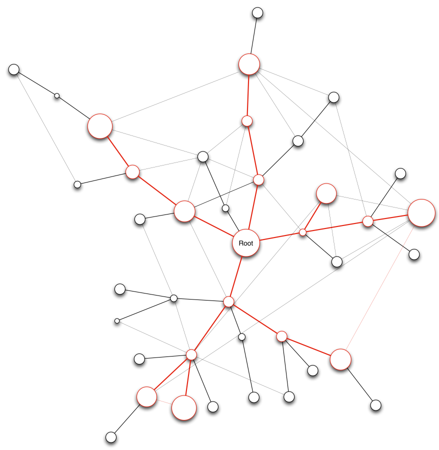
One of the main issues that needs to be addressed in the development of a reusable system capable of visualising a wide range of Web spaces, is the problem of representing very large graphs aiming at discovering topological patterns or outliers. However, algorithms for laying out such networks tend to generate computational overloads and, more importantly, create layouts that present readability problems due to the limited screen space available. Even if an algorithm is capable of placing all the nodes in the display so they do not overlap, annoying edge crossings may clutter the visualisation, compromising again the readability.

The most common techniques to overcome this problem apply layouts that use the screen space more efficiently, such as matrix representations (HF06); or generate clusters from similar or highly connected components in the graph (MG05) aimed at compacting them, and hence reducing the amount of space needed to represent the graph. However, these kinds of clusters often derive groups of nodes with no clear semantics.

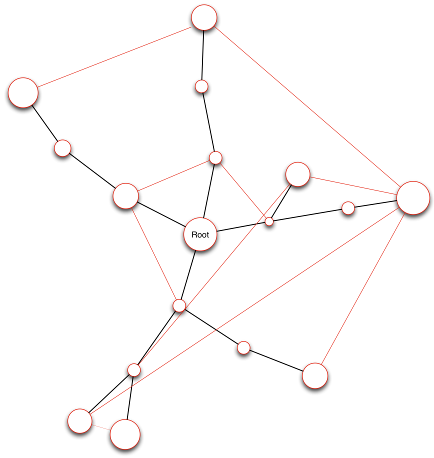
Despite this, we claim that in some cases users might prefer to understand something from a specific portion of the dataset rather than obtaining global patterns of the complete structure. Hence, another approach to deal with large graphs is the extraction of representative and meaningful subgraphs that comprehend a set of relevant nodes according to data attributes that provide valuable information to the end user according to a specific information need.

In an effort to extract meaningful hierarchies from very big trees, Furnas proposed the concept of *Degree Of Interest* (DOI) (Fur86) as a numerical value that can be calculated for each node in a tree. This approach allows the user to create compact abstracted views by selecting nodes above a certain level of interest. Such a concept can be calculated as a two part function made up of an A Priori Interest function (API) that defines the importance of a node in the structure, and a distance function D that depends on the distance of the node y to a focus node x. However, this approach only works in hierarchical structures, where the DOI of a parent is always bigger than the DOI of its children (HC04). Otherwise, disconnected graphs may appear when applying the threshold, hiding the implicit structure of the graph.

In our case we applied a similar approach to what Van Ham and Perer proposed in (vHP09) to our Graph's Logic System (GLS) extending Furnas' technique to graphs. To do so, we first generate an ordered list of candidates that contains the nodes that satisfy some weighted criteria based on the analyst needs (big nodes in Figure 3.3). Such criteria can be defined according to specific data attributes of the dataset, or to general topological metrics such as centrality measures that capture topological interest of the nodes. Furthermore, the system also enables the user to define a threshold,



(a) Initial graph. In red, the traversed links and nodes that follow the rules according to the information needs of the user.



(b) Complete subgraph containing links among all the selected nodes.

Figure 3.3: The GLS selects relevant nodes (big ones) as well as irrelevant nodes that are part of the structure (highlighted in red).

expressed as a percentage of the global value of a specific attribute, in order to assure the collection of high percentage of nodes. This can be useful, for instance, when trying to collect the most visited pages in a website.

Once the candidates list has been generated, the system computes a spanning tree of the graph, calculating the shortest paths from a focus node x to every candidate. This tree is traversed, storing the visited nodes in a final nodes list. The nodes not included in the nodes list are discarded, assuring a minimum substructure. The system halts when it reaches a maximum predefined number of nodes N, or completes a certain desired amount of levels L. Finally, the GLS retrieves the existing links between the nodes in the final list that were not previously collected in the traversing process, to ensure the acquisition of a complete portion of the graph (thinner links in Figure 4.5(b)).

Considering that the amount of final nodes is limited and smaller than the number of vertices in the graph, the complexity of our methodology gets reduced to the complexity of the algorithm that computes the spanning tree. As we only concentrate on nodes, edges are considered to have the same weight, which allows the use of a breadth first search algorithm that has a time complexity of $O(|E| + |N|)$.

The main benefit of our approach is the possibility to segment large networks according to the analysts needs while preserving its structure, generating contextual subgraphs that reduce the amount of information to be shown while keeping the same amount of relevant data. Furthermore, assuming that the candidates belong to the same connected network, this methodology assures in all cases the existence of a connected subgraph.

Nevertheless, the main drawback of this technique is that it cannot control the number of irrelevant nodes that are finally collected, as they depend on the distance between the selected focus and the relevant nodes, providing subgraphs with a few interesting items. To overcome this problem, users may be assisted by the system, which might suggest a list of relevant nodes based upon their eccentricity value.

3.3 Visualisation System

The visualisation system enables the visual exploration of graph-structured data. The main approach of the system is to provide hierarchical visualisations that generate less cluttered visualisations with a clear structure. Thus, the first problem was to decide which visual metaphors to use. We reviewed the most common approaches for visualising hierarchies (see Section 2.4.2), and concluded to use a space filling approach based on a treemap, which is a contrasted and studied visual metaphor (Kob04; AK07), and a node-link diagram visualisation. The radial tree seemed a good approximation, as it is easier to understand than other approaches such as the hyperbolic tree, which generates difficulties in users due to its distorted layout (Kob04). The radial tree provides a similar metaphor to the classic tree, which is one of the most well known representations, but making a better use of the screen's real state. To evaluate the usefulness of this visual metaphor, and its capabilities for providing insight from the topology of the data, we conducted an experiment comparing it with two baselines.

3.3.1 Evaluation of the Radial Tree Layout

The main goal of our study was to focus on the potential topology-discovering benefits of the radial tree layout (sometimes called disk tree), in order to discover if it is a visual metaphor that is easy to use and understand. We selected the classic (Figure 3.4(b)) and the indented tree (Figure 3.4(a)) as baselines to compare with, as they are very well known and easy to use visualisations (AK07).

We developed the indented tree using the JTree class from the Java SDK[2]; and adapted examples of the Prefuse toolkit (HCL05) to provide the classic and radial trees.

Our main hypotheses were:

- H1: The classic and radial layouts will perform better, in terms of time and errors, than the indented tree in the discovery of topological features of hierarchies.
- H2: Users will need the same effort in terms of time and errors to perform tasks with the classic and the radial tree, as they are very similar visualisations.
- H3: Tasks performed with a large dataset will be easier to perform with the radial tree, as it makes a better use of the screen's real state.

The null hypothesis was that the radial tree may be significantly worse for enabling topology-based tasks.

In order to prove these hypotheses, two datasets were selected and presented with the three different visual metaphors to a set of subjects in a mixed design: each subject completed all the tasks twice with two different visualisations and different datasets to avoid learning effects and lengthy experimental sessions. Response time per task as well as number of errors were measured and analysed through an ANOVA test (more details below). Users were given a maximum task completion time of 4 minutes in order to suggest that they performed the tasks in the minimum amount of time, but with the maximum comprehension about the data.

Part of the experiment was conducted in a lab and some users were tested in-house with their own computers. An evaluator was present in all of the tests, giving five minutes training with the three interfaces with a demo dataset. The training consisted of an explanation of the visualisations and the available interactions, and a few minutes to allow users to perform random tasks with the system. Tasks were also introduced to the participants to avoid any misunderstandings.

[2]http://download.oracle.com/javase/tutorial/uiswing/components/tree.html

Considerations taken on the visualisations

One of the most important considerations taken in this evaluation has been the idea of being impartial among the visualisations. Our goal was to ascertain which layout provides the most insight on the topology, for this reason, we provided the three visualisations with the same interactions and visual features:

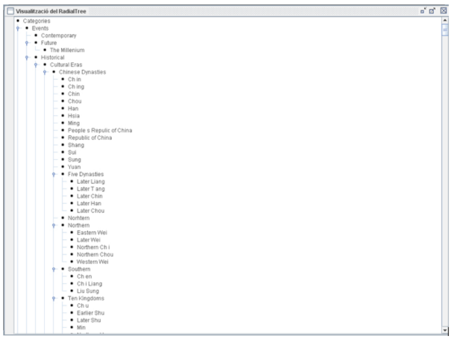
- Expand and collapse: expanding and collapsing nodes is a typical interaction provided by in file browsers of most operating systems that use the indented tree visual metaphor. Since this is an interaction that allows reducing the amount of information visualised both, the radial and the classic tree were also provided with this interaction.
- All nodes expanded by default: the three visualisation were, by default, presented fully expanded at the beginning of every task.
- Node sizes: a preliminary pilot test revealed that users expected the size of nodes to be representative of their number of descendants. To overcome this problem, users were explicitly aware that there two sizes of nodes were provided to all the visualisations. The small size was used to point out the node that was expanded, and the large one to represent a collapsed node.

Experiment's datasets

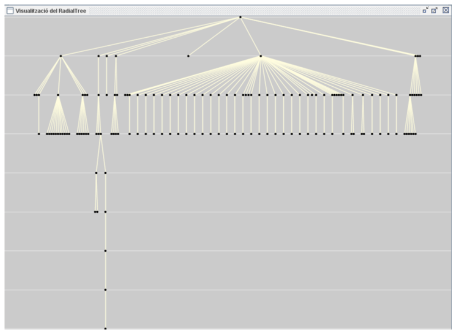
Two datasets of different sizes were used to compare response times and errors. The datasets information may be considered irrelevant, as the test was focused on topological tasks rather than on browsing tasks. While the browsing tasks need a consistent hierarchy with semantic information, topological ones just need node names to allow the user to identify and write down the answers to the questions. The small dataset was unbalanced, it had 142 nodes, 8 levels and a relevant outlier (Figure 3.4(b)). The large dataset was also unbalanced, it had 1101 nodes and also 8 levels of depth (Figure 3.4(a) and 3.4(c)).

Subjects

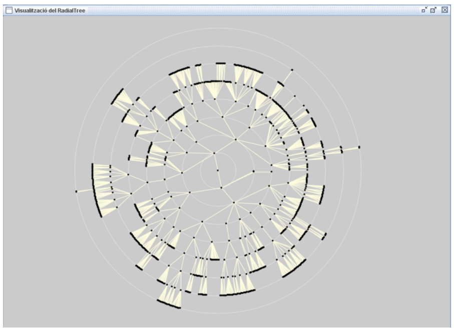
Thirty subjects participated in the experiment, all of them being computer scientists. Taking into account that each user performed the experiment



(a) Indented tree layout with the big dataset.



(b) Classic tree layout with the small dataset



(c) Radial tree with the big dataset

Figure 3.4: The three visualisations used in the experiment. The indented and the radial trees show the big dataset while the classic tree represents the small one.

twice (developing all the tasks with two visualisations), and that there were three visualisations and two datasets (this is six different experimental settings), thirty subjects assured the availability of ten measures per experiment condition.

Tasks

The experiment was divided into two parts. In the first part users had to solve the six tasks (presented in Table 3.1 with one visualisation, and a dataset. The second part consisted of accomplishing the same tasks but with a different visualisation and the other dataset. Hence, we avoided learning effects.

Task	Description
T1	How many nodes/dots do you think there are in this hierarchy?
T2	What is the maximum depth of this tree?
T3	Is the tree balanced or unbalanced? (We consider a tree balanced if the difference between any leaf on the tree is not higher than 2)
T4	Which node in the first level has the biggest number of nodes under it?
T5	Find 3 nodes with 6 direct descendants.
T6	Find all the ancestors of a node placed in the deepest level of the hierarchy.

Table 3.1: Tasks used in the experiment with their type.

The first three tasks may be considered 'overview tasks', while the rest were related to details of the hierarchy. After each task, users were told to rate their level of confidence with the visualisation, scoring it from 0 to 10.

Results

To analyse the results of the test, we conducted a two-way ANOVA analysis over time and errors per task.

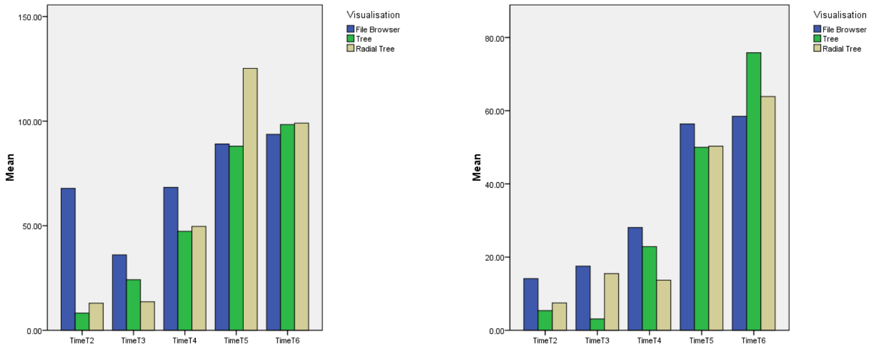
The aim of T1 was to see if any of the three visualisations provided a more accurate perception of the number of nodes in the hierarchy. To do so, we calculated the deviation between the number of nodes available in each hierarchy and the one estimated by the users. However, these tasks were too difficult and did not provide relevant results due to the variability of the responses.

In terms of the rest of the tasks, we found significant differences in time response of T2 ($F(2,21) = 16.5$, $p < 0.001$) and number of errors in T6 ($F(2,27) = 5$, $p = 0.014$) with the small dataset.

Regarding the big dataset, we found statistically significant differences in the response time of T4 ($F(2,17) = 4.6$, $p = 0.019$).

In all the cases mentioned above, a Bonferroni post-hoc test revealed that the radial tree was significantly better that the indented tree, and that there were no significant differences with the classic tree. These results can be explained due to the ability of the classic and the radial tree to provide a full and understandable overview, supporting our hypothesis. However, it must be stressed that, as happened in similar experiments like (BN01) and (AK07), there were no big differences among the three visualisations.

Figure 3.5 shows mean response time for tasks T2-T6 revealing some of the differences mentioned before between the classic and radial visualisations compared with the intended one.



(a) Small dataset.

(b) Big dataset

Figure 3.5: Mean times for tasks T2-T6

In terms of the subjective scoring asked after finishing every task, users ranked evenly the three visualisation with averages of 8.62 (indented tree), 8.72 (classic tree) and 8.83 (radial tree). However, when asked explicitly about the radial tree all of them agreed that it was an interesting visualisation, very similar to the classic tree and, hence, very easy to understand.

Discussion

As we have seen, no major significative differences were found between the radial approach and the two baselines, supporting H2. However, a few tasks reported statistically significant differences in favour of the radial tree, when compared with the indented tree. As argued before, such differences may be a consequence of the overview provided by the radial metaphor. Nevertheless, we also argue that the default interactions of zooming and panning provided by the Prefuse toolkit affected the possible differences between the classic and the radial tree, as users easily overcame the problem of using too much horizontal space presented by the classic tree.

With this experiment, we partially supported our hypothesis and rejected the null one, validating the usage of the radial tree layout as its structure does not provide a bigger cognitive load than the baseline. In addition, this visual metaphor makes better use of the screen real state, and does not provide a distorted layout that may be more difficult to interpret (Kob04).

3.3.2 User Interface and Interaction Design

The WET user interface loads XML files formatted with the GraphML standard, and is basically a coordinated multipanel visualisation that provides different types of panels, that can be divided in three types:

- *Visualisation panels*: such panels contain interactive visualisations that allow the exploration and manipulation of the data.
- *Information panels*: there are panels dedicated to the textual representation of the data.
- *Control panels*: such panels enable the user to control and customise the visualisations, facilitating the creation of specific data mappings, or expressing dynamic queries that modify the amount of data to be visualised.

According to the definitions proposed by Engelhardt (Eng04), the WET interface (Figure 3.6) is a multipanel made up of a set of representations with different syntactical meanings. The different panels available in WET are:

- The *visualisation area* (A), provides a set of coordinated visualisations using a highlighting interaction that allows the user to choose whether to highlight the same node in all the active visualisations, or to highlight the path from a hovered node to the root of the hierarchical layouts. Furthermore, analysts may change the focus of interest at any time by dragging any node to the centre of the hierarchical visualisation. This action causes the recalculation of the hierarchy using the selected node as the new root, affecting all visualisations. In our

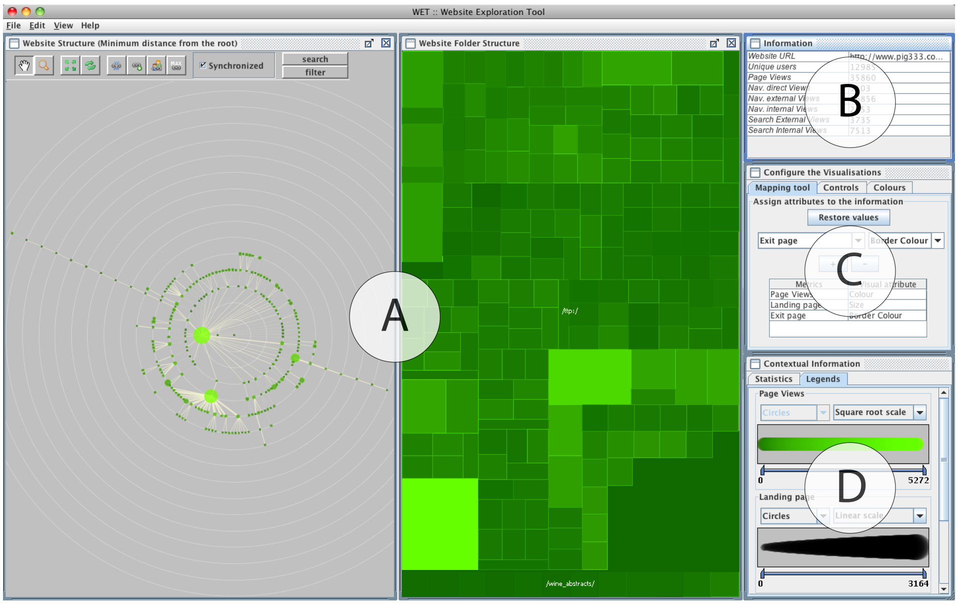


Figure 3.6: The WET user interface is a coordinated multipanel made up of four main parts: the visualisation area (A), the information table (B), the mapping tool (C) and the legends panel (D).

main visual metaphor, the radial tree, WET performs an animation between the old position of the nodes and the new ones following the algorithm proposed in (YFDH01). Such animation provides a smooth transition between the old and the new hierarchy, assisting the users in the process of understanding the change of the layout. Figure 3.7 illustrates a sequence of the keyframes in such animation process the root.

It is also important to stress that, due to the modular approach followed in the development of the system; WET provides programming interfaces that enable the user to easily integrate any other visualisation capable of supporting the data available in WET.

A contextual menu is also available for the nodes, which are the most important containers of information. Such a menu allows the user to show in- or out-links not shown by default, and can also be customised to provide more advanced features, such as opening in a browser the URL associated to the node, if available.

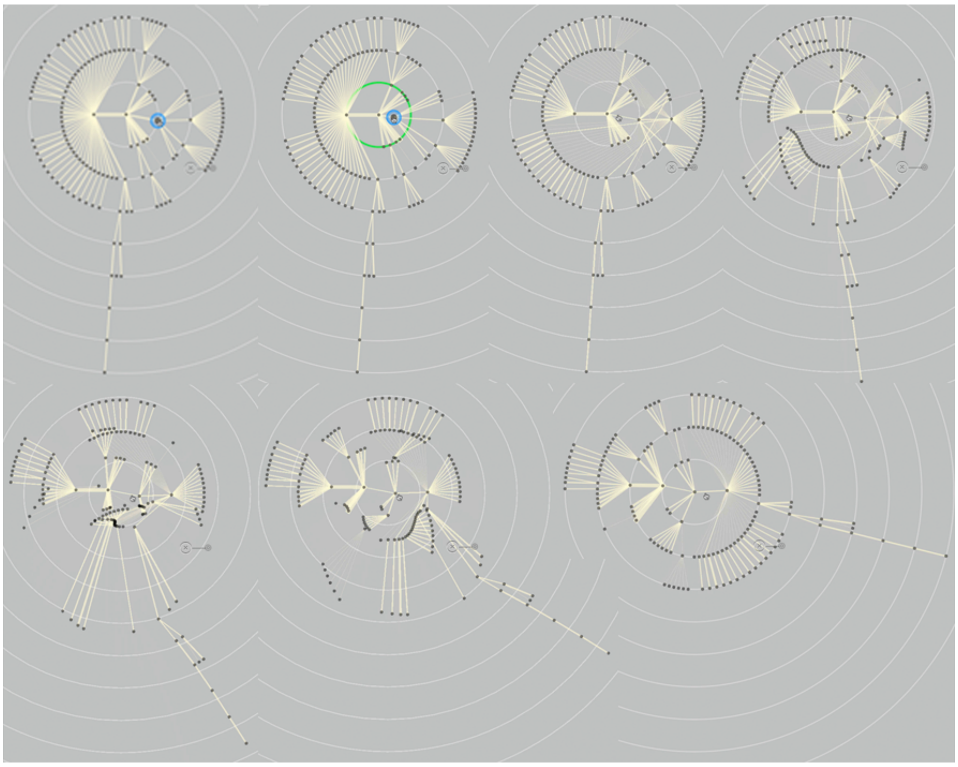


Figure 3.7: Sequence of images showing the animation performed when changing the root node of the hierarchy.

- The *information table* (B) provides detailed textual information regarding the whole graph, or specific nodes that have been clicked. This is the place where global metadata that is probably difficult to represent in the visualisation can be shown.

- Below this panel, the *mapping tool* (C) (Figure 3.8) allows the user to control and customise the visualisations through a set of tools distributed in different tabs, that enable him/her to define the maximum node size or the colours used in the different palettes. Nevertheless, the most important feature of this panel is a menu that allows the user to pair every available data metric with visual attributes such as colour, size, shape and location. The visual codings selected by the user are propagated in all the active visualisations, providing the analyst with the same information across the different visual abstractions

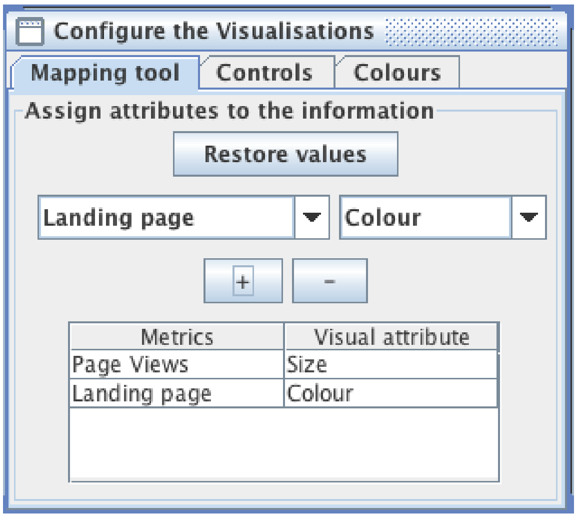


Figure 3.8: The mapping tool enables the user to assign the visual attributes that will represent the selected metrics.

- Finally, the *legends panel* (D) displays a legend for every binding created between a metric and a visual attribute. Figure 3.9 shows an example of two legends, one representing a categorical metric (content cluster) and the other a numerical one (number of page views). One of the most important characteristics of such legends is their interactive nature, which allows the user to express dynamic queries that helps to filter the data shown in the visualisations. To do so, a double slider can be adjusted to select the range of values to be displayed. Data items in the visualisation that are not in the selected interval are given a high transparency value at the same time as they lose their interactivity capabilities. Moreover, as can be seen in the image, legends also enable the user to change the mapping scale and shape of the nodes. The latter allows him/her to visualise node items in one dimension (using bars) or in two (using circles). Mapping the size of the visual elements in one dimension enables an easier comparison that is not that apparent with 2D representations.

The syntactical relationships existing between the panels are expressed in different ways. As has already been stated, visual metaphors existing in the visualisation area are coordinated using the linking and brushing interac-

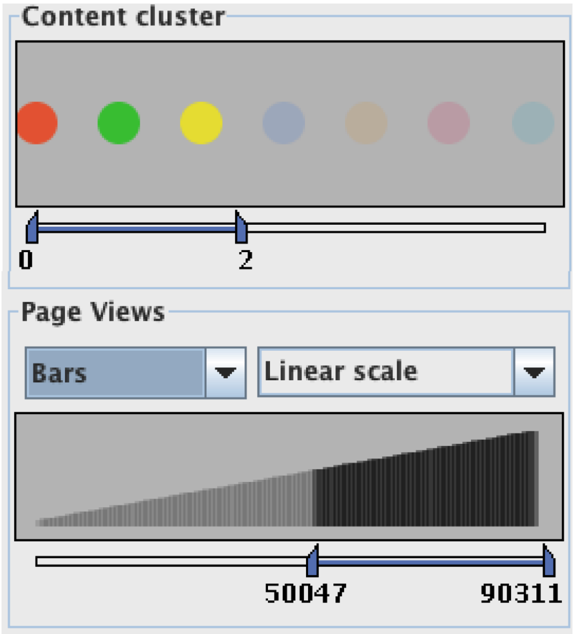


Figure 3.9: The slider makes legends more powerful, turning them into widgets that enable the user to express advanced queries that filter the information of the visualisations.

tion. In terms of the rest of the panels, every time a visualisation object is selected, detailed information is shown in the form of specific metadata in the information table and barcharts located in a tab of the legends panel.

The **interaction design** of WET was conceived as the vehicle that drives the analytic discourse (TC05), enabling the analyst to change the visualisations according to his/her information needs. The main interactive feature of WET is the Mapping Tool, which is the system that allows him/her to visually encode the available metrics with visual attributes such as size, shape, colour or location of the nodes. The visual codings selected by the user are propagated in all the active visualisations, providing the same information across different visualisations and information panels.

Each visualisation implemented in the system also contains, by default, a customisable toolbar that provides functionalities such as fitting the visualisation to the window, revealing hidden edges (in case they exist), and a dynamic search box that searches in real time the visible nodes that contain a specific piece of text in their metadata.

Figure 3.10: Example of a scented widget in WET. A histogram provides visual cues of the amount of information available. The double slider allows the user to select only the links between a range of data, from a defined links metadata. Information regarding the exact items matching the query is also provided.

In addition, as we have already mentioned before, the interactive legends enable the user to express complex queries in real time that allow him/her to filter non-desired nodes in the visualisation. Along that same line, WET provides scented widgets, which are "enhanced user interface widgets with embedded visualisations that provide information scent cues for navigating information spaces" (WHA07). In our case we have provisioned the node link diagrams with explicit edges with small histograms showing the usage of the visible links. Such histograms aim to provide visual cues on the number of edges that will be hidden after moving the sliders, and are provided along with a double slider that enables the user to specify the upper and bottom limits of the links to be displayed. Figure 3.10 shows an example of a scented widget, which reveals the typical distribution of link usage in a site, where many links are barely clicked while only very few are popular. As the reader can also notice, the widget has also been provided with textual information that informs the user about the number of edges matching the specified query, assisting the information foraging.

3.3.3 Implementation Details

The visualisation system of WET has been developed in Java, and uses different toolkits and APIs. Figure 3.11 represents the building blocks of WET, which makes extensive usage of Java swing libraries to manage the panels and the whole layout of the interface, and takes advantage of the reusable and InfoVis based architecture provided by the Prefuse toolkit[3] (HCL05).

The WET user interface has been developed using a very customisable architecture, which allows the user to modify the whole tool by manipulating variables of a properties file. Such a file allows the user to enable or disable any menu and panel in the system, defines the initial configuration

[3]http://www.prefuse.org

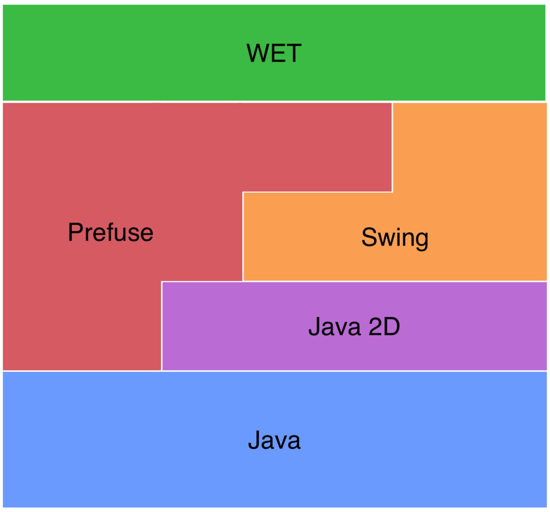


Figure 3.11: Building blocks of WET, which has been implemented in Java libraries. The Prefuse toolkit has been extensively used and adapted to create the visualisations and manage the user interactions.

of the system, and allows the user to configure the data attributes of the GraphML files that have to be considered as mappable metrics. Therefore, many different versions of the same tool can be deployed, aimed at tackling different visualisations needs according to the desired input file from several types of data sources.

Figure 3.12 shows the basic architecture of the system. The system can be deployed as a standalone application or as an applet, using the *WetApplet* or the *WetStandalone* class. The *Wet* class is a *DesktopPane* from the Swing library, which enables the user to convert the different panels of the system into floating windows. This class is also responsible for deciding which panels are available based on the data in the configuration file, instantiating the required visual metaphors. Any new visualisation must inherit from the *VisualMetahorDisplay* class, which provides a controller called *MetaphorController* that incorporates several interactions from the 'prefuse.action' package. Current visualisations adapt the existing layout in the 'prefuse.visualization' package, and are responsible for receiving a reference to the main data graph, and manipulating (e.g. generate a hierarchy) to create the most suitable data architecture to fit the needs of its

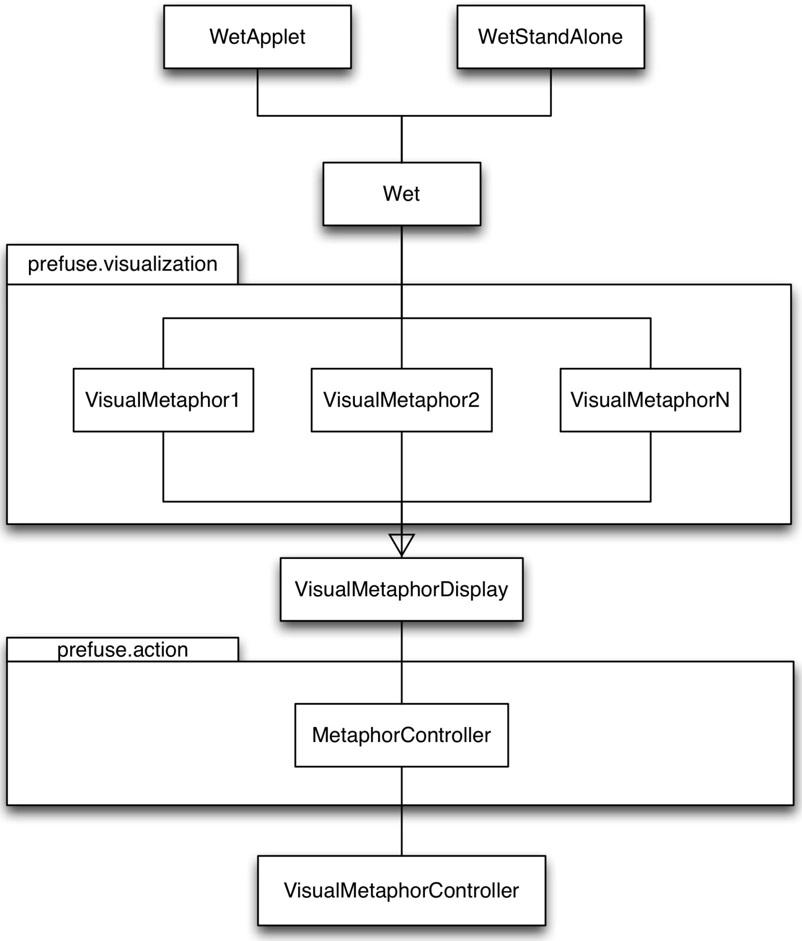


Figure 3.12: Basic WET class architecture.

layout. Finally, the whole system is controlled by the *VisualMetaphorManager*, which receives messages from all the visualisations, and decides which visualisations or panels should react to the user actions.

3.4 Conclusions

In this chapter we have presented the architecture and the different features incorporated in the WET system, which constitutes the main tool for our research. WET is a reusable system for storing, mining and visualising graph-based structures, aimed at supporting the analysis of Web spaces. In the design process of the system, we have proposed a new approach to tackling very large graphs which suggests the extraction of subgraphs that maximise the amount of information provided, based upon the interest of the analyst. The main advantages of our system is that it preserves the real structure of the graph, by only filtering nodes that are not relevant, maintaining the real structure of the graph.

We have also presented a study of the usefulness of the radial tree as a visual metaphor suitable for representing hierarchical structures. Such a metaphor will be used in several contexts as we will see in further chapters. Our experiment provided evidence that reveals that this visual metaphor is, at least, as easy to use as other very well known visualisations such as the indented and the classic tree.

Finally we have also introduced the interface and the interaction design of the visualisation system of WET, which enables the user to combine a set of visualisations on the same data in a coordinated manner. Furthermore, the system allows the user to customise the visualisations enabling him/her to visually encode the nodes according to their main attributes.

Chapter **4**

Visual Mining for Website Evaluation

This chapter presents the most important contribution of this dissertation, which is the application of the WET system to the visualisation of website data. We will study different approaches for supporting the analysis of website data in order to provide an explorative environment for assessing website usability. We will address problems involved in the visualisation and exploration of website structure and usage, providing new evidence of relevant methods for extracting meaningful inner hierarchies from webgraphs. We will conclude discussing the benefits and possible impact of our system, along with new research problems raised during this research.

4.1 Research Problem

The increasing importance of the Internet and its wide adoption require website operators to continuously improve their website. To accomplish this goal it is crucial to understand and evaluate its usability. Moreover, information architects, web analysts and website evaluators (from now on, analysts) play the role of interpreting web data, aiming to infer users' behaviour. However, although current Web Analytics tools have undergone a tremendous evolution, analysts must still examine large amounts of statistics that end up providing very little insight into website data. For instance, an example of a typical and straightforward result that one may come up with after a page view analysis (i.e. counting the number of requests that each page has received) may hypothesise that pages not visited embrace non relevant content while in highly visited ones the opposite occurs. As pointed out by Spiliopoulou (Spi00), this assumption may only be valid if the user perceives the site in the way the designer did or, as proposed by Baeza-Yates and Poblete (BYP06), because there might exist no clear way to reach such pages (either by browsing or by searching). Therefore, to make more suitable and realistic assumptions, we argue that there is a need to provide exploratory tools that support the examination of website data within a context, such as the hyperlink structure of a website or the paths performed by the users. Furthermore, current tools for web analytics lack proper support for analysts to explore and drill down their data. We claim that additional exploration capabilities would be essential for web analysts to discover information and correlate causes and effects based on user web navigation histories. As such, our goal is to provide a system aimed at filling this gap.

4.2 Related Work

The first visual representations of web data were devoted to easing the navigation of Web spaces, providing an interactive representation of the hyperlink structure to avoid the lost in cyberspace phenomenon (TS07). As a first example, Andrews proposed a 3D approach based on an information landscape metaphor tightly coupled with 2D visualisations to provide location feedback and alleviate disorientation (And95). However, main approaches for tackling this problem were based upon two dimensional node-link diagrams where pages are represented as nodes and links as edges among them. These approaches deemed the webgraph as a hierarchy, visualising structural links and hiding cross-reference ones (BRS92). Cone trees (RMC91) and the hyperbolic tree (MB95) represent two examples of early Web visualisations that followed such an approach. Nevertheless, Munzner concluded in her study (Mun98) that Web users do not need visual representations of such a hyperlink structure, as it adds cognitive load. However, the author argued that these techniques could benefit a specific target community such as webmasters or website operators, which represent the target users in our research.

Following research efforts focused on visualising users' paths to provide visual cues of their behaviour. Early works extracted navigation sequences from log files (PB94; CS99; HL01a) that were represented also using node-link approximations.

The main research trend in the late nineties considered mixing structural and navigational data at the same time, using website structure as a background where navigational data was overlaid (CPM+98; CS99).

Following this approach, Chen et al. (CZT+07b) proposed a visual data mining system with the ability to visualise multi-layer web graphs. Their approximation is based on the combination of layers that contain information from web usage overlaid on top of the website structure. More recently, the same authors also proposed a visual metaphor called Polygon Graphs (CZT+07a), which extends the concept of the radial tree metaphor by generating polygons that appear from the connection of parent nodes in the hierarchy with representative points in the edges calculated according to any usage metric of its children node. This approach generates visual artefacts within the hierarchy, facilitating the detection of outliers and patterns within the data.

In VISVIP (CS99), Cugini et al. proposed another tactic for visualising web navigational data using a simplified representation of a web graph laid out using a force directed algorithm where nodes were colour-coded to denote page type. Users' paths were then represented using smooth curves upon the graph. The time that subjects spent at each page was also represented as dotted vertical lines whose height was proportional to the amount of time spent in that node.

Fry also proposed Anemone[1] (Figure 4.1) (Fry08), a system for capturing the evolving nature of a website using the concept of Organic Information Design. The author extracted the hierarchical structure of a website and depicted users' activity by dynamically overlaying lines representing users' clicks and modifying node thickness to denote activity. Fry's approach represents a very creative and artistic approach.

To date, the application of visualisation techniques to the field of web analytics has been very poor. Analytics reports only provide information regarding web usage with simple representation methods such as tables or charts.

In the late nineties, Analog[2] and WebTrends[3] encouraged the adoption of

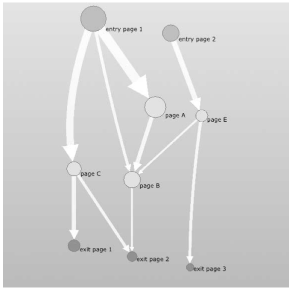
[1] http://benfry.com/anemone/
[2] http://www.analog.cx/
[3] http://www.webtrends.com/

Figure 4.1: Image of Ben Fry's Anemone.

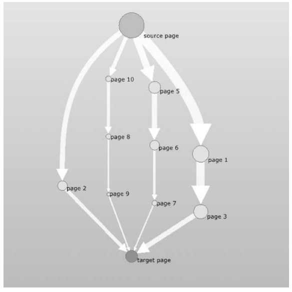
Web analytics with the first usage reports (Figure 4.3). Charts and graphics available in such reports are still used nowadays in commercial tools.

Perhaps, one of the most relevant contributions to current web analytics tools was made by Keahey and Eick, who decomposed the problem of understanding users' behaviour into a set of node link visualisations focused on representing the paths between predefined pages (KE02) through the use of Sugiyamas layout-like representations (Figures 4.2(a) and 4.2(b)). Another of their main contributions was the funnel charts (Figure 4.2(c)), which were proposed as a solution for representing dropout rates along a designated sequence of pages. This approach has been widely adopted by the web analytics community, and is available in most of the current web analytics solutions. Another technique that has been widely adopted in some web analytics packages is the click density visualisation, which superimposes semi-transparent colour coded rectangles over the links of a single web page, representing click stream. This approach has also become part of current web analytic tools, although it presents problems in pages with a large concentration of links.

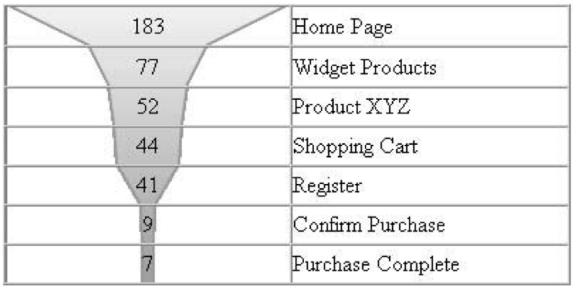
In one of the most relevant books from the emerging discipline of web analytics (Kau07), Kaushik proposed a new paradigm for the web analytics community based on the integration of clickstream data with data from all sorts of outcomes. Moreover, the author proposed a paradigm shift from the analysis of KPIs, to the usage of KIAs (Key Insight Analysis). Indeed,



(a) Paths between source/target pairs



(b) Paths between two pre-defined pages



(c) Funnel chart allows the user to find dropout rates along a designated sequence of pages

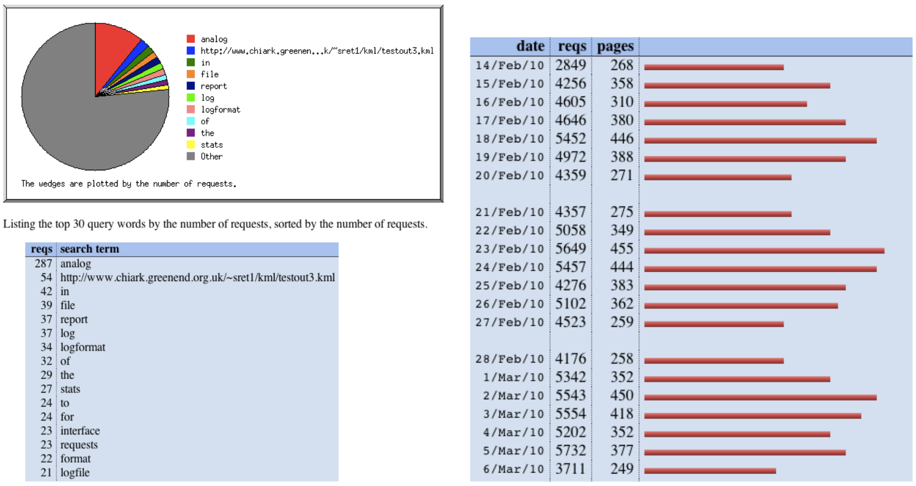
Figure 4.2: User's behaviour visualisation from (KE02).

most of the KIA proposed by the author are based on new visualisation approaches that provide data insight, such as click density analysis introduced before, and the approach used by software like CrazyEgg[4], which generates heatmaps (Figure 4.4(b)) on top of single web pages representing areas with the most activity (i.e. the most mouse clicks made by the users).

4.3 A Hybrid Approach for Visualising Website Data

In order to understand the rationale behind the working practices of web analysts, we actively participated in a monthly web analytics event for a

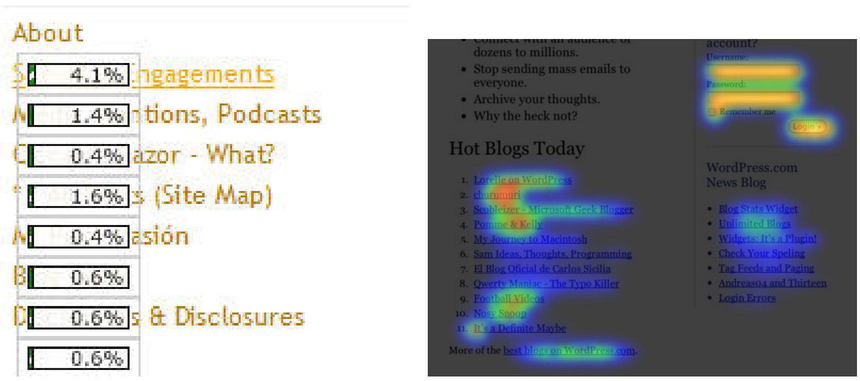
[4]http://www.crazyegg.com/



reqs	search term
287	analog
54	http://www.chiark.greenend.org.uk/~sret1/kml/testout3.kml
42	in
39	file
37	report
37	log
34	logformat
32	of
29	the
27	stats
24	to
24	for
23	interface
23	requests
22	format
21	logfile

date	reqs	pages
14/Feb/10	2849	268
15/Feb/10	4256	358
16/Feb/10	4605	310
17/Feb/10	4646	380
18/Feb/10	5452	446
19/Feb/10	4972	388
20/Feb/10	4359	271
21/Feb/10	4357	275
22/Feb/10	5058	349
23/Feb/10	5649	455
24/Feb/10	5457	444
25/Feb/10	4276	383
26/Feb/10	5102	362
27/Feb/10	4523	259
28/Feb/10	4176	258
1/Mar/10	5342	352
2/Mar/10	5543	450
3/Mar/10	5554	418
4/Mar/10	5202	352
5/Mar/10	5732	377
6/Mar/10	3711	249

Figure 4.3: First analytics reports from Analog showing keywords used (left) and number of page views per day (right)



(a) Click density report with bars showing on top of link representing their usage. Image extracted from http://www.kaushik.net/avinash/2006/06/tips-for-web-analytics-success-for-small-businesses.html

(b) The CrazzyEgg software uses a heatmap visualisation superimposed on top of a web page.

Figure 4.4: Examples of click density visualisations showing most clicked areas/links in a web page.

whole year, as well as conducting informal interviews. We observed that most analysts use several tools at the same time to enrich their analysis, combining systems such as Google Analytics or Omniture Site Catalyst (among others) for dealing with usage data, and tools such as Google Webmaster Tools or Yahoo! Site Explorer to track possible errors in the site structure, as well as analysing web search engines' incoming traffic. While analysts tend to work with statistics provided by those analytics packages, most of them end up developing their own metrics that satisfy the tracking needs of their businesses.

From the conversations with the analysts, it was interesting to see that although current web analytics tools do not provide ways to understand the structure of a website, the experts agreed that understanding such a structure is an actionable insight that cannot be tackled nowadays with existing tools. Furthermore, as expected, we also learned that most analytic tasks involve the careful examination of usage metrics from which analysts infer users' intentions and interests. When prompted, analysts stated that reports provided by current tools usually lack rich and appealing ways for visualising their data, as current table-based approaches (Figure 4.5) obscure the possible existence of data long tails, and provide data without a clear context to empower its judgment in the decision making process.

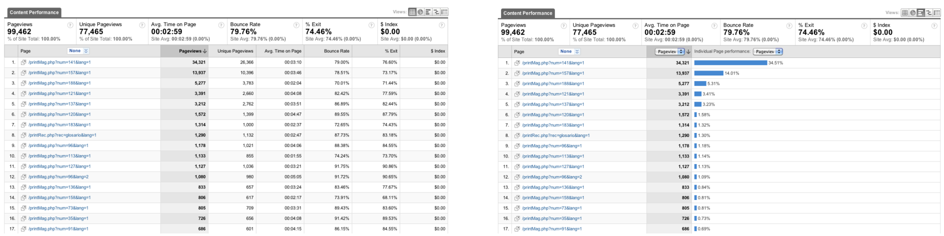


Figure 4.5: Classic Pageviews report from a web analytics package. Visual approaches can't provide a whole overview of the pages in the site.

From our understanding of web analysts working practices, we decided to make the most out of the WET architecture to develop a hybrid system capable of showing structural and usage data at the same time, aiming to tackle broad problems that may be easily generalised to more specific ones. Hence, our goals were to:

- Study new approaches for visualising website structure.

- Provide a hybrid and customisable visualisation of structure and usage.
- Enable the discovery of interesting and relevant links, such as broken links.
- Develop well known web metrics that may be examined within the system.

We developed a custom extension of the DMMS (see Section 3.1) of WET, in order to support the data collection and mining from website data. The very first specific research problem we faced was the problem of how to visualise the structure of a website, and how this visualisation can benefit its assessment.

4.3.1 Gathering and Preprocessing Website Data

As we have seen in section 2.2, the very first step for creating a visualisation is the data gathering process. In this case, we had to deal with structural information as well as usage data from websites.

In terms of the collection of the website structure, we describe and analyse below the two existing methodologies:

- Using a crawler, and inferring structure from usage data logs. A crawler is an automatic agent that accesses the site's pages following all the links starting at a given initial page or seed. The main advantage of crawlers is that they capture the whole accessible structure of a web space in a certain moment in time. However, this methodology depends too much on the response of the network and a website host's bandwidth requiring a limit to the depth of the pages to be crawled.
- Using access logs: Using access trails left in the web server logs. As we have seen in Section 2.3.3, access log files contain users trails, storing the URL of the requested content and the referrer page used to reach it. From this information, a whole webgraph can be collected based upon users' actions. The main drawback of this technique is that it does not detect non-visited pages, leading to an incomplete and unconnected structure. On the other hand, this method can be used offline, avoiding possible network problems.

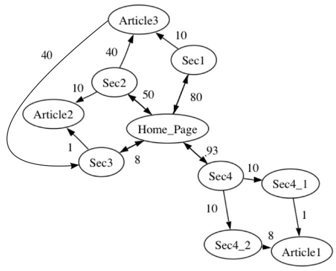
Both methodologies have specific characteristics that make them appealing for tackling different problems. Crawlers provide a more complete webgraph structure; however, they can only be used when aimed at obtaining the current structure of a site. If the goal is to obtain a site structure in a specific point in time, such an organisation may be obtained directly from logs, assuming that non navigated links will not be collected. Moreover, this approach also reflects a website structure influenced by users' behaviour, which we will see in Chapter 5 that can have its own importance in the data analysis.

Once the topology of the website has been collected, the Data Management (DM)) module extracts usage data by parsing access log files which are stored, indexed and logically linked to the pages of the webgraph already stored in the database. For now, our system supports both; an intrusive way which requires the usage of a Javascript tracker code that has to be added in the source code of all the pages in the site; and a non intrusive way which uses log files available in the web server provided by the user.

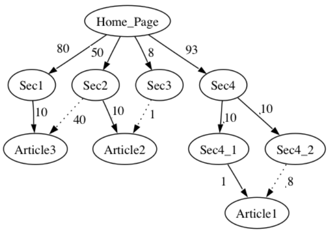
Finally, the DM system cleans up irrelevant records existing in the log file that may produce inconsistencies, such as entries related to embedded objects for example, multimedia files or scripts. Furthermore, usage data is also cleaned from entries generated by crawlers from search engines or spam agents. We are currently using two methods for doing so: the first one focuses on deleting entries with known user agents (such as the ones that identify themselves as bot or crawler); the second one uses a heuristic approach that takes into account the number of visited pages and the average time spent on them. After a set of tests, we are currently erasing sessions with more than 50 pages, with an average time of 10 seconds or less. Nevertheless, these parameters may be optimised to the type of usage of any specific target website.

4.3.2 Visualising the Webgraph Structure as a Hierarchy

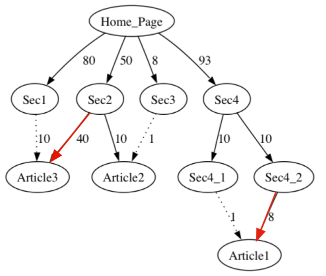
Websites are inherently hierarchical structures. As we have seen in Section 2.3.3, most websites are organised along sections and subsections that form a hierarchical structure. Hence, this feature can dramatically help in the representation of a webgraph, as hierarchical representations are more straightforward and do not have to deal with problems such as edge crossings. Therefore, representing a website as a hierarchy means that from all the existing links (up to $n(n-1)$ since web graphs are directed), $n-1$ must



(a) A graph representing a website



(b) Hierarchical tree from the BFS algorithm



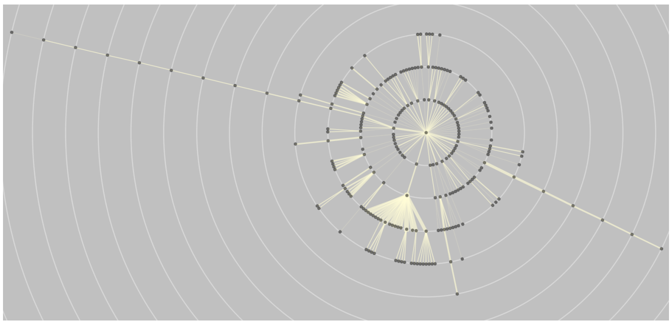
(c) Hierarchical tree from the W-BFS algorithm. New selected links are highlighted in red.

Figure 4.6: An example of the hierarchies extracted from the BFS and the W-BFS algorithms. Relevant links highlighted in red in Figure (c) are not visualised with the BFS approach.

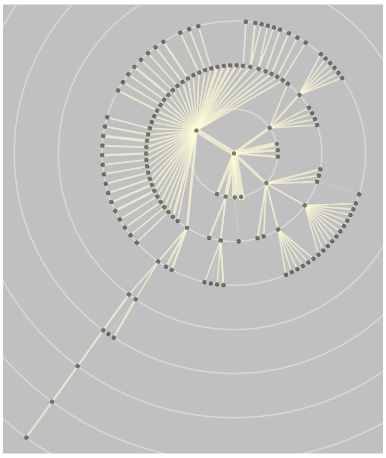
be selected with the condition that every node, except the root, must end up having only one incoming edge. Ideally, selected links in the hierarchy should be structural, avoiding the visualisation of cross-reference links. We describe below our approach for extracting such links.

A heuristic approach for extracting website structure

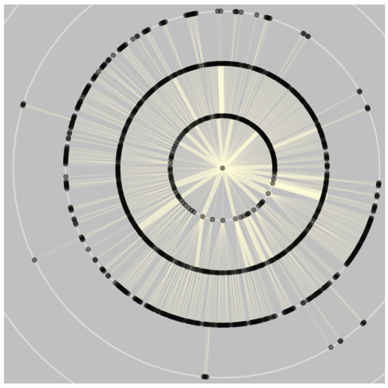
As it appears in the literature (BRS92), a common technique to extract the website structure is to use a Breadth First Search algorithm (BFS), which is the default algorithm implemented in the WET system. This algorithm selects the first non-expanded incoming edge per node that appears in the



(a) Structure of a news website.



(b) Structure of a personal website



(c) Structure of a blog.

Figure 4.7: Images showing website structure from different types of websites. Images from websites (a) and (b) helped web designers to discover relevant outliers, while website (c) showed a very compact and homogeneous structure that came out of the highly connected topology of a blog.

data traversal. However, due to the nature of this algorithm, there might exist ambiguities when parents of a node coexist in the same level. While the BFS algorithm selects the very first parent found per node, we propose a heuristic driven improvement based upon the consideration of the frequency of usage of links in the site. Given that webgraphs are weighted directed graphs, whose edge weight corresponds to the usage of its corresponding link, we propose using this weight to disambiguate between parents coexisting in the same depth. Therefore, our approach is able to improve the number of highly used links visible in the tree. Figure 4.6 shows an example of website with the different hierarchies obtained with the different algorithms.

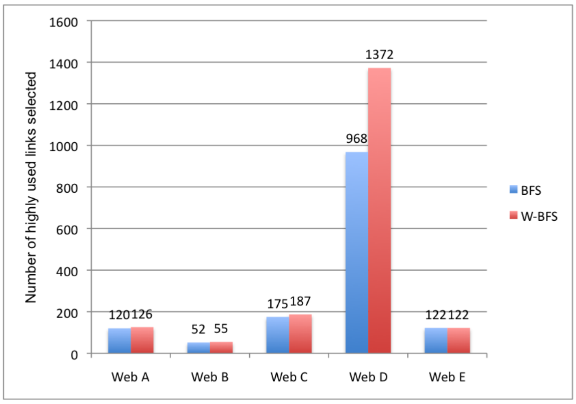


Figure 4.8: Average of the number of highly used links selected by the BFs and the W-BFS algorithms. The W-BFS always provides more relevant links.

The resulting hierarchy can be seen as an overview of the whole webgraph, where every page is located at the minimum distance from the selected root node. Therefore, such a structure allows the user to easily discover the depth of every page in the tree (i.e. the minimum number of clicks needed to reach it), supporting the discovery of outliers in the form of pages that need many clicks to be reached, which might suggest the existence of design inconsistencies. Figure 4.7 shows three examples of websites represented with our method. As can be seen, Figures 4.7(a) and 4.7(b) have relevant outliers while Figure 4.7(c) presents a more compact structure.

In order to model the W-BFS algorithm and compare it with the classic BFS in a practical way, we collected the webgraphs of four very different websites: a site containing collections of articles based upon newsletters (having about 1.000 pages and 5.0000 links), a personal site (having about 140 pages and 600 links), and two highly visited blogs (one having 1.500 pages and 60.000 links and the other with 7.600 pages and 30.000 links). We also used our graph logic system (see Section 3.2) to generate samples of webgraphs with 3.000, 6.000 and 7.000 pages.

With such webgraphs, we observed that although our approach does not guarantee the extraction of the real structure of a site, the existence of the

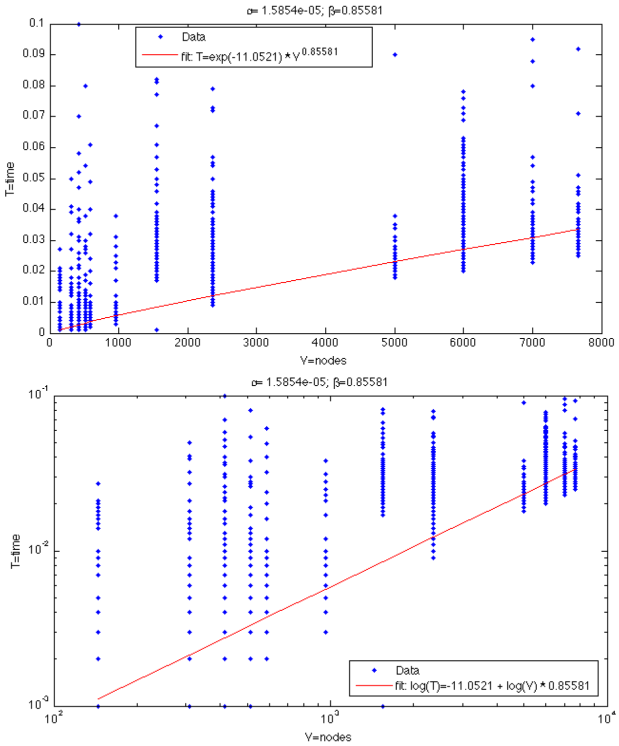


Figure 4.9: Time performance plot of the BFS algorithm with samples from seven different webgraphs. Bottom plot shows the fitting curve in the log-log scale.

usage heuristic guarantees the apparition of more relevant links in the final hierarchy compared to the traditional BFS algorithm. As can be seen in Figure 4.8, the W-BFS algorithm selects more highly used links than the BFS.

In an effort to compare the performance of both algorithms, we ran the BFS and the W-BFS with every page in the site with out-links. We used the least squares method to extract the fitting curve of our samples. The plots from Figure 4.11 shows the relation between the number of nodes of the webgraph versus the time needed by the algorithm to calculate the hierarchy in the linear (top) and the log log scale. Figure 4.10 represents the same information for the W-BFS approach.

As can be seen in Figure 4.9, the W-BFS algorithm performs slightly slower than the BFS as expected. Both curves are almost linear, and its sublinear

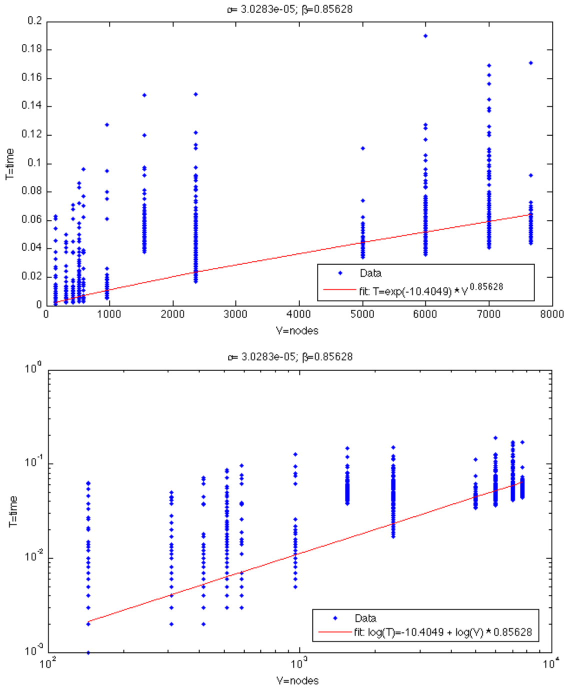


Figure 4.10: Time performance plot of the W-BFS algorithm with samples from seven different webgraphs. Bottom plot shows the fitting curve in the log-log scale.

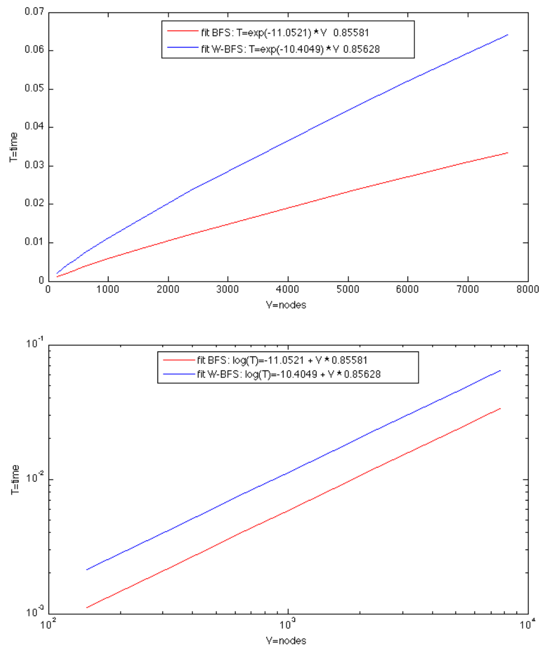


Figure 4.11: Time performance comparison between the BFS and the W-BFS algorithms with samples from seven different webgraphs. Bottom plot shows the fitting curves in the log-log scale.

behaviour can be explained by the distribution of the number of nodes of the data samples, whose mean is 3179 and its standard deviation is equal to 2735.

Results suggested, as previously expected, that our W-BFS approximation is scalable. In addition, our W-BFS approach might also be improved by using content similarity methods, which may inform our algorithm in order to decide which parent is more similar to the target page in terms of content.

4.3.3 Combining Usage Data and Website Structure

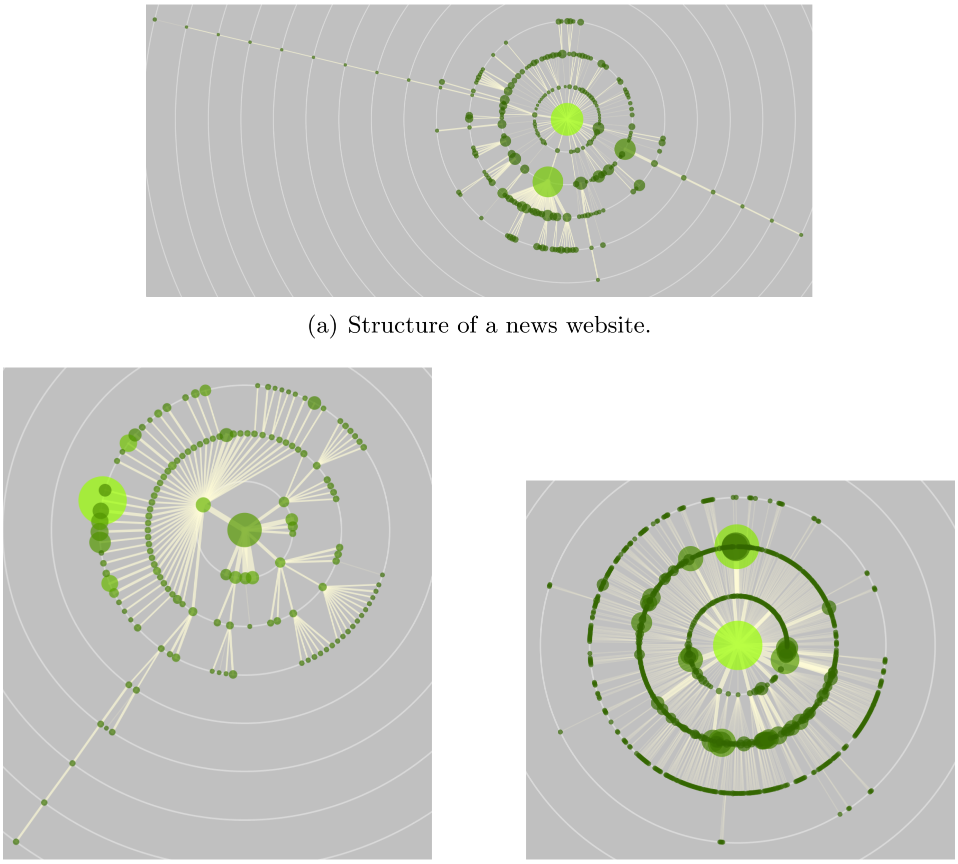
Our hybrid model for assessing website usability contemplates the utilisation of the website structure as the context where to show and interpret usage data. Such data may be related to the nodes in the site, helping to get insight into which pages are more visited; or to the links, helping to reveal the movements of the users in the site. We define below the exploratory strategies implemented in WET that enable the investigation of such data.

On the one hand, in terms of the visualisation of node-related statistics, the Mapping Tool enables the user to create a visual binding between a selected web metric and a visual attribute, enabling the integration of the structure with usage metrics.

Figure 4.12 shows images of the three websites introduced before with statistics of number of page views (size) and number of times that each page has been a landing page (colour), visualised on top of the structure. As expected, the images reveal that in the three websites there are not many visited pages at the outer rings of the site, as users tend to make very few clicks. From these visualisations, it can also be inferred that the most visited pages (big nodes), are the ones with most visits from search engines (bright green coloured nodes).

On the other hand, assuming that the extraction of the hierarchy may hide potentially interesting information (i.e. the links not selected by the W-BFS algorithm), we developed tools to assist the user in the process of discovering relevant links: a toolbar enables the user to first, show all the links in the graph (Figure 4.13(a)), and then use the scented widgets to filter non-relevant links. Most used links according to the dynamic query expressed through the widget become apparent, as can be seen in Figure 4.13(b).

Furthermore, specific links with metadata generated in the DMMS can also be highlighted. For instance, and following one of the typical website us-



(a) Structure of a news website.

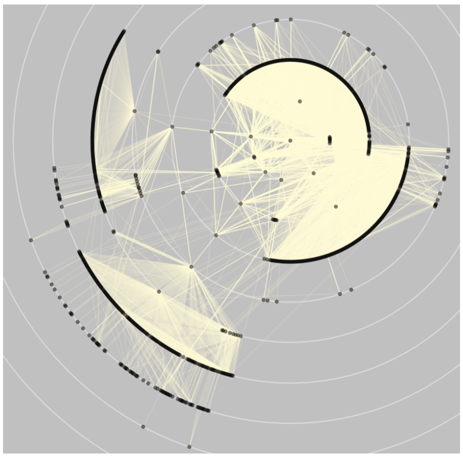
(b) Structure of a personal website

(c) Structure of a blog.

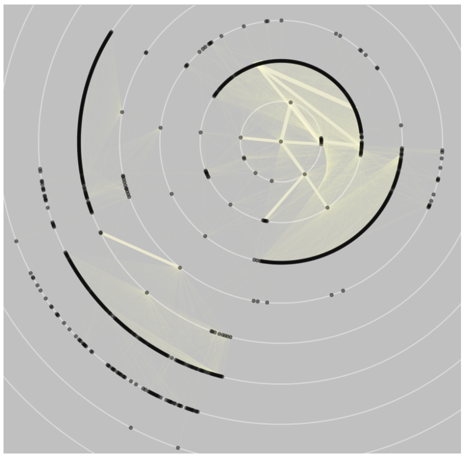
Figure 4.12: Image of the previously seen websites where size has been mapped accordingly with the number of page views and colour according to the number of times that each page has been a landing page.

ability reports, broken links can be superimposed in the hierarchy. Figure 4.14(b) shows an example that represents links that have been considered as broken by the DMMS visualisation provided by WET, contrasted with the same report in the Google Webmaster Tools[5] software. In our approach, red coloured links help to discover at a glance pages that contain such errors as well as their relative distance (in terms of clicks) to any other page in the site. With Figure 4.14(b), the webmaster of the site realised

[5]www.google.com/webmasters/tools/



(a) A graph representing a website



(b) Hierarchical tree from the BFS algorithm

Figure 4.13: Left image shows the messy visualisation with all the links of the site. Right image show most relevant links after using a scented widget.

that two scripts generated non-valid URLs, leading to a massive existence of broken links.

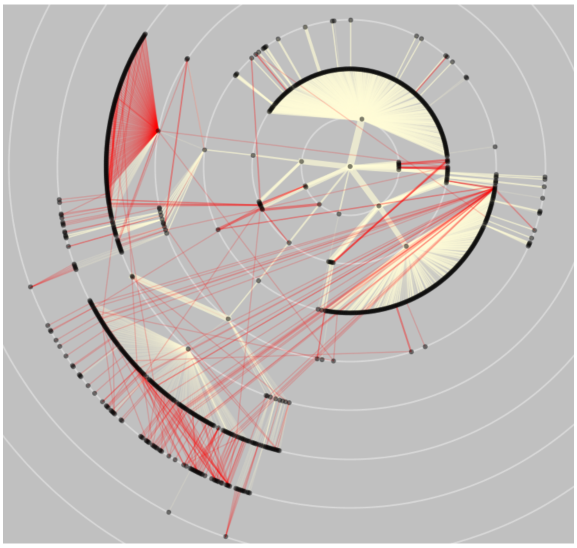
4.3.4 Interaction Design

We designed specific techniques to enhance the explorative experience with WET. The first one was related to overcoming the highly overlapped visualisations generated by the radial tree algorithm when handling dense graphs, as can be seen in Figure 4.12. While this fact does not provide problems when making an analysis at a high level, as the system sorts the nodes according to their size, locating bigger nodes always below small ones; an evaluation (see Section 4.5) revealed that overlapping represented a very important issue when performing analysis tasks. To overcome this problem we implemented a radial distortion algorithm that separates nodes near a distortion area built around the mouse, while maintaining the circular properties of the radial tree metaphor (Ant09). To do so, we used the approach proposed by Sarkar and Brown (SB92), with the differences that we do not modify nodes size, as we wanted to maintain their aspect (possibly modified according to a metric decided by the user). The algorithm finally projects

Show URLs: **Not found (128)** | Unreachable (2)

URL	Detail	Linked From	Detected
http://www.infovis.net/ http://www.infovis.net/applet/wet/login_sp.php	404 (Not found)	11 pages	May 29, 2010
http://www.infovis.net/\"	404 (Not found)	4 pages	May 23, 2010
http://www.infovis.net/\"printMag.php?num=157&lang=1%5C%22	404 (Not found)	5 pages	May 20, 2010
http://www.infovis.net/\"printMag.php?num=51&lang=1%5C%22	404 (Not found)	5 pages	May 20, 2010
http://www.infovis.net/Biblio/Glosario.htm	404 (Not found)	10 pages	May 31, 2010
http://www.infovis.net/Biblio/libros.htm	404 (Not found)	29 pages	May 22, 2010

(a) Part of the broken links report from Google Webmaster Tools.



(b) Broken links (in red), are visualised on top of the structure.

Figure 4.14: Top image shows a broken links report from the Google Webmaster Tools software. Bottom image shows the superimposed approach of broken links on top of the website structure, which facilitates the discovery of the most conflicting pages in this regard.

separated nodes onto the circle to keep the radial structure of our visual metaphor, as can be seen in Figure 4.15.

We have already introduced our methodology for extracting meaningful hierarchies from the webgraph of a site. However, there are several possible root pages to be considered in a site. Although the home page seems to be the best candidate, as it is usually considered as the main entry point to the site, we observed that one of the most important reports used by web analysts is the landing pages reports. Landing pages are pages where users *land*

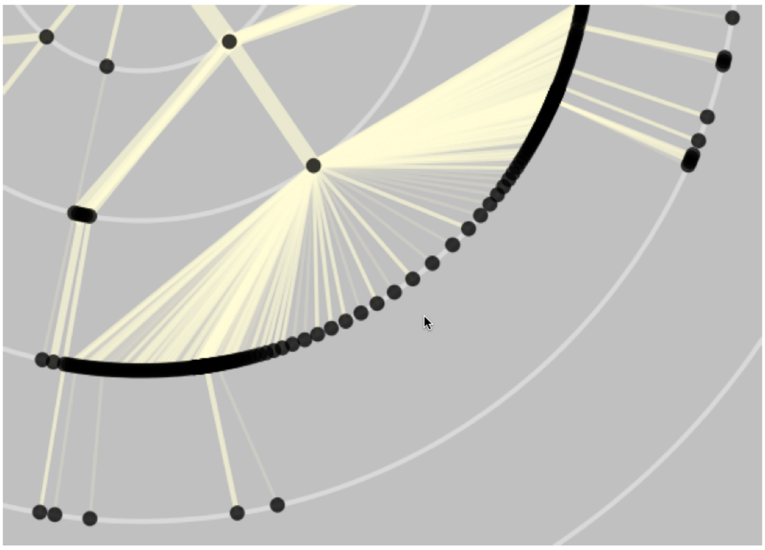


Figure 4.15: Radial distorsion applied to improve the overlapping of the nodes. Nodes are separated radially, maintaining the whole aspect of the radial tree as well as providing more space to nodes closest to the mouse.

on a website after issuing a query to a web search engine or following a link from any other site. Marketing strategies are often developed to generate specific landing pages, stressing the importance of analysing a website from the perspective of these very specific starting points. The dragging action to change the root node in the built-in hierarchy in WET (see Section 3.3.2) can be used in order to fulfil this need, enabling the user to modify the root node of the hierarchy and extending the exploratory possibilities of WET. Therefore, analysts are able to view the website structure from different perspectives. It is also important to remark that this technique enables the analysts to easily discover the number of clicks between the selected root and any other node in the site, becoming a powerful tool for designing the hyperlink structure of a website.

Finally, we have built a system to mark *conversion pages*. From our observations of web analysts working practices, we discovered that the main targets of their analysis are those pages, as well as those landings pages that surround them and bring traffic. *Conversion pages* are end points that, when reached by a user, mean business results. Using the contextual menu, the user can change the visual properties of the node, giving him a star shaped and greener look that makes it easy to identify in any visualisation, as can be seen in the screenshot in Figure 4.24).

4.3.5 Evaluation

In order to validate our hybrid model at the abstraction level (see Section2.6) and decide future steps in our research, we conducted a formative evaluation with eight real webmasters and website operators (from now on, analysts), all of them males. Websites owned by the analysts were crawled to gather their structure, which was combined with usage data of one month.

We interviewed the analysts before and after the test to better understand their current working practices, as well as to collect relevant feedback to improve our tool and to understand the usefulness of the hybrid approach proposed in WET.

Task	Description
T1	How many clicks are needed, at least, to reach the furthest page in your website from the home page?
T2	Discover the number of times that the last link of the previous path has been used.
T3	Use the mapping tool to visualise more than one metric at the same time..
T4	Identify the most visited page in your website. Which is the most important traffic source to this page?
T5	Identify the page with more in-links.
T6	Use the search tool to locate one of the pages of your website using its url. Now, visualise how the rest of the site is organised around it.
T7	Identify the page with the largest number of broken links in your site.

Table 4.1: Tasks performed by the users in the first evaluation of the hybrid model proposed in WET.

Analysts were given 15 minutes training with a sample website. We also asked them to perform seven tasks (see Table 4.1) with their own website while verbally expressing their thoughts and doubts. Tasks were specifically selected to guide them into the different features available in WET, while covering some of their daily tasks.

Two evaluators were responsible for taking notes on the session, annotating user comments and relevant reactions to discover usability problems of the tool as well as general user behaviour. Tasks were selected according to classic working practices of web analysts, and were designed to allow the user to try the main features of WET.

Results and discussion

Users reacted very positively in the very first moment they saw the structure of their own website, all of them stating that the structure resembled the mental model they had in their mind. Two of them also discovered at first glance design errors, finding obsolete links that pointed to private pages and broken links that were generated due to errors in php scripts.

We observed no major problems during the accomplishment of the tasks, it being interesting to see that the users solved all the tasks regarding the identification of nodes in the radial tree according to a specific metric using the size (preferably) and the colour visual attributes, stating that the other visual attributes (mainly shape and the stroke colour of the nodes) were less intuitive than fill colour and size.

During the final interview, users stressed the novelty of the tool, stating that it might be a very interesting add-on to current web analytic tools, especially to discover possible outliers in the web design. When prompted to rate the usefulness of WET from 0 to 4, all of them gave it a score between 3 or 4, averaging 3.3. The main factor that made the ratings drop below 4 was that users said that while the tool promises to detect structural errors, a more user behaviour oriented visualisation showing real user sessions would make the tool more appealing. Nevertheless, all of them stated that the tool might be useful to control the structure of their site.

This evaluation allowed us to redefine the user interface, modifying the position of some components such as the legend panel and the mapping tool, leading to the final design of the tool presented in Section 3.3.2.

4.4 Visual Analysis of Users' Behaviour

Although the hybrid approach proposed above was well rated by participants in our formative evaluation, users tend to request other approaches to better understand user sessions. While our hybrid approach shows usage metrics on top of the website structure, whose edges' thickness can help to understand relevant links, it is still complicated to understand the main trends regarding the behaviour of the users while surfing a website. Users' sessions provide such information.

The problem of visualising users' paths is a research topic that confronts some of the beliefs of web analysts. Kaushik, in his book (Kau07), states that the most repeated paths in a site are usually performed by less than 5% of the users, which does not represent actionable information to suggest a site improvement. Nevertheless, when prompted, web analysts do still believe that in many cases routes may provide a crucial understanding of relevant contents associated to users' information foraging strategies, as well as the existence of intricate hyperlink structures which are difficult to be navigated. Web analysts also agreed that part of the problem is due to the lack of new approaches for analysing in detail users' trails, as current techniques, such as the funnel chart (see Figure 4.2(c)), only provide specific data regarding the achievement of landmarks in a site, such as starting a purchasing order, or filling in billing data in an eCommerce website.

In this section we explore two approaches for examining users paths, aiming to provide explorative visualisations that enable the discovery of both main trends in users routes as well as specific areas where users converge into the same navigation patterns.

4.4.1 Identifying Users' Paths

We developed a new module for the Data Processing and Web Mining system which deals with the extraction of users' sessions. A session is an ordered sequence of pages visited by the same user. Following the approach proposed in (CMS99b), sessions are extracted from log files by grouping log entries that share the same IP and user agent, with time gaps no longer than 30 minutes (this value can be parameterised according to the characteristics of the website).

Due to the stateless nature of the HTTP protocol, the existence of proxies and caches, and the common use of the *back button* of the browsers, which provides cached pages to the user without noticing the server, log files generally contain incomplete sessions. Our system checks the continuity of all the sessions, using information of the topology of the site to infer possible uses of the *back button*. Hence, if a sequence of visited pages is 'A - B - C - D', in the simple website of Figure 4.16, the system detects a non-contiguous navigation, with a gap between the pages C and D, and infers a possible click to the *back button* from D until A. Hence, the final complete sessions will be 'A - B - C - A - D'. Contrary to the approach proposed by Cooley (CMS99b), we do not fill the gap with all the backtracking pages, as this may provide misleading information. Instead, we directly indicate a movement from page C to A with a virtual 'back link' that will be explicitly treated in the visualisation. Therefore, the existence of a visualisation of website structure might help the analyst to infer the real movement of the users.

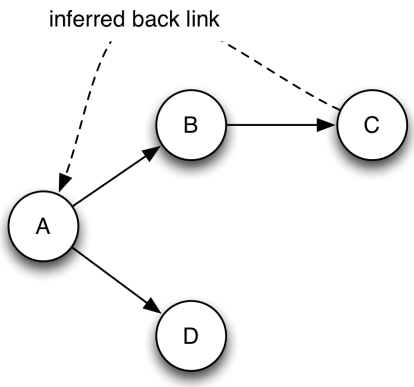


Figure 4.16: Example of the path completion algorithm. The navigation sequence A - B - C - D is transformed into A - B - C - A - D as the system detects that A is the first page already visited that links to D in the path of the user.

If there is no previously visited page that helps to fill the gap of a navigation sequence, the system splits the session into different paths. We interpret the resulting sequences as routes taken by a user to accomplish different goals, interpreting that a user session can be made up of several paths meaning different informational needs of the user, or different strategies to reach the desired information goal.

4.4.2 Visualising User Paths

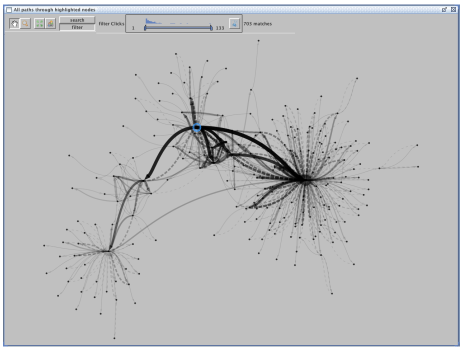
Up to now, we have presented ways of visualising a web graph by representing hierarchical abstractions based on the website structure and usage of links. However, as we have seen, the extraction of hierarchies implies hiding links that might be relevant. Therefore, we have developed a visualisation dedicated to the representation of users' sessions called Session Graph, which lays out a graph made up of all the routes performed by users who navigated through one or more selected pages in the target website. During the process of composing this graph, metrics such as the number of times that a link has been clicked on within this context, and the percentage of traffic that it is taking from its source node are displayed on demand.

The Sessions Graph is then represented using a force directed algorithm provided by the visualisation package Prefuse, which creates repulsion forces among nodes, and spring forces among links. The layout is iteratively calculated until the system reaches a stable state, where nodes' movement decreases. The algorithm running time is $O(max(O(VlogV), O(E)))$.

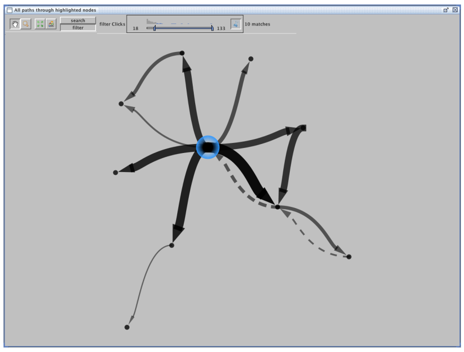
Edges of the graph are visually encoded using transparency and thickness, with a square root scale that avoids cluttering the visualisation, as can be seen in Figure 4.17(a). As we will see later in the evaluation of the system, the main benefit of our approach is that it enables the user to explore the navigated pages between several nodes of interest. To do so, WET provides a dynamic filtering system that enables the user to dynamically filter links according to their usage, making navigation trends apparent. Once links have been filtered the user can re-run the layout algorithm with the remaining links and nodes reaching more understandable visualisations as can be seen in Figure 4.17(b).

This approach has two main benefits compared to current methods used in web analytics packages such as the funnel chart:

1. It enables the user to understand the ecosystem of a page: we define the ecosystem of a page as the set of page visits by the users in the paths across a specific page. This definition came out of a discussion with an expert analyst (see Section 4.5), who stated that current tools barely provide such information, which might be of crucial importance in specific contexts such as analysing the navigated pages around a conversion page, which can be understood as the single resource which can deliver sales.



(a) The Session Graph



(b) The graph gets understandable and revealing after hiding non relevant nodes and recomputing the force directed layout.

Figure 4.17: Examples of the Session Graph, generated by superimposing all the paths that pass through one or more selected pages of the website.

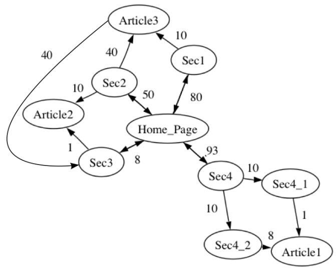
2. It enables the user to discover highly repeated subsequences: rather than focusing on the most used paths performed by the users, our approach allows the user to visually discover highly repeated subsequences of paths.

4.4.3 Characterising Users' Navigation as a Hierarchy

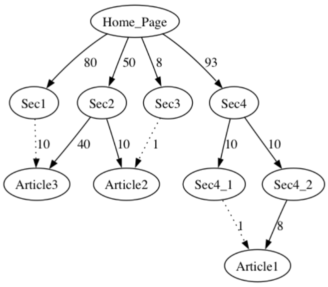
In previous sections we have shown how structural hierarchies can be extracted from a web graph, as well as our approach of depicting users' paths that can be dynamically filtered. Hereafter we will introduce a new approach for generating hierarchical structures from websites called "Usage Tree".

The generation of hierarchies to model users' paths was applied in previous research. For instance, Pei et al. (PHMAZ00) generated a hierarchical structure from users' sessions to facilitate the execution of their mining algorithms. Moreover, Dierbold and Kaufmann (DK01) presented a visual approach for depicting site maps based upon a Sugiyama layout applied to users' paths which was considered as a collaborative approach to distinguish possible interesting content.

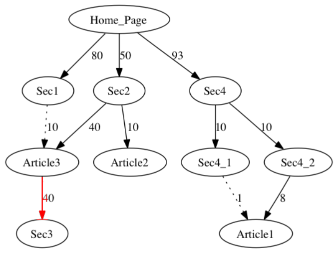
To present our approach, we will first define the concept of *Local Maximum*



(a) A graph representing a website



(b) Hierarchical tree from the W-BFS algorithm



(c) Hierarchical tree from the BM algorithm

Figure 4.18: An example of the visual abstractions generated by the different hierarchical approaches from a web graph.

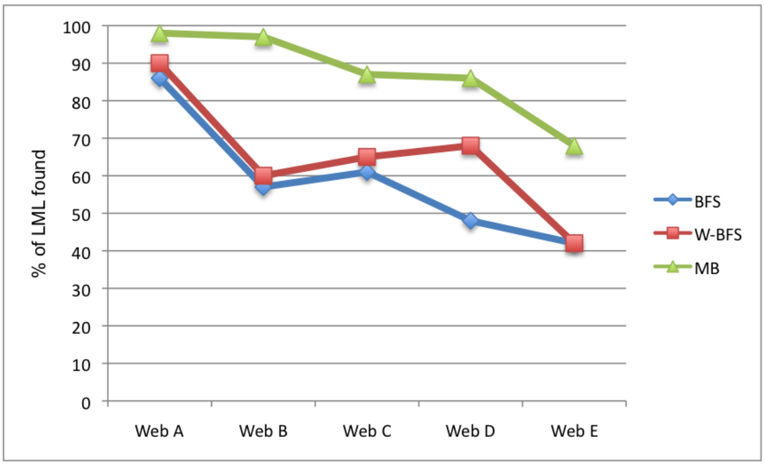
Link (LML) as a link that provides the highest amount of traffic to a node in the webgraph. Hence, each node in the webgraph, except the root (typically, the home page or any landing page on the site), will have at least one LML. Our goal was to evaluate if it is possible to generate a comprehensive hierarchy containing the maximum number of LMLs, in order to get a visualisation based on users' behaviour, rather than on its theoretical structure. Our approach uses a Maximum Branching methodology based upon Edmond's algorithm (Edm67), which collects the LML of every node in the site, and resolves the cycles that this process may produce.

Evaluation of the methodology

To evaluate our new technique, comprehended in the algorithm level of the nested model introduced in Section 2.6, we used a heuristic based on the

					BFS		W-BFS		MB	
website	#nodes	#links	#sessions	Total LML	$\overline{X}$	σ	$\overline{X}$	σ	$\overline{X}$	σ
A	143	565	5303	140	86%	10%	90%	12%	**98%**	0%
B	366	21865	2589	91	57%	5%	60%	5%	**97%**	.4%
C	374	11767	51518	291	42%	3%	42%	3%	**68%**	.2%
D	957	4898	47191	287	61%	1%	65%	1%	**87%**	.1%
E	2812	51807	160561	2017	48%	1%	68%	3%	**86%**	.1%

(a)



(b)

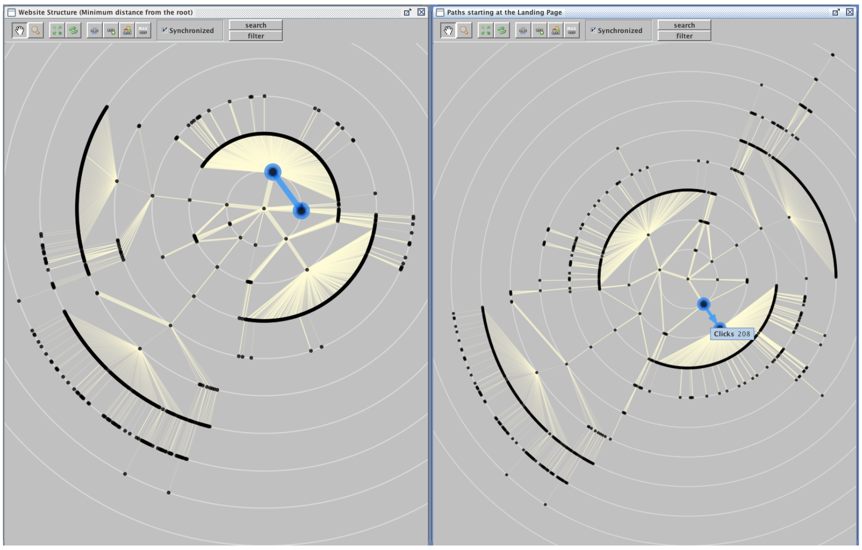
Figure 4.19: Characterisation of the websites used in the experiment and its results. Table (a) describes the different websites used in the experiment, and plot (b) shows the percentage of LML found by each algorithm. The green line represents the MB algorithm, which outperforms the BFS algorithms in terms of LMLs found.

number of LMLs selected by each one of the already presented algorithms for extracting a hierarchy from a graph: the Breadth First Search algorithm (BFS), the Weighted-Breadth First Search (W-BFS) algorithm presented in Section 4.3.2, and the Maximum Branching (MB) approach that can be seen in Figure 4.18.

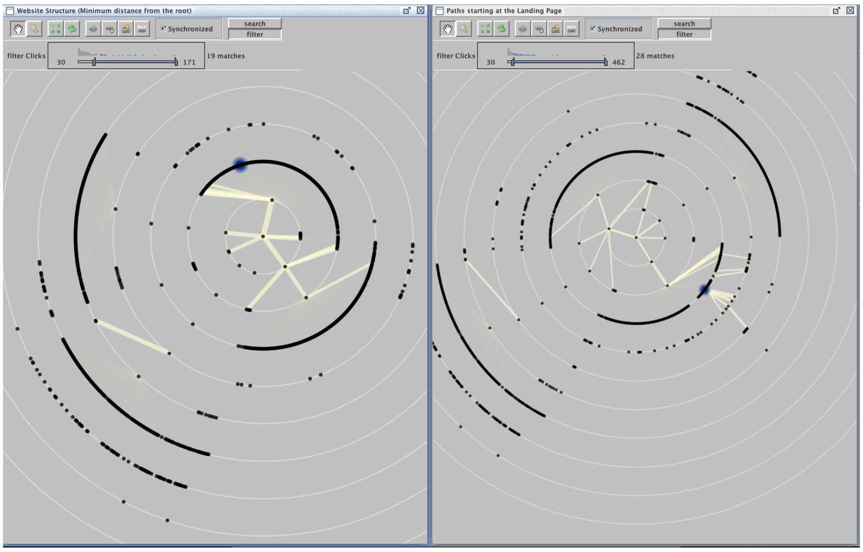
The experiment consisted of counting the number of LMLs selected by each algorithm computed in five different websites, each one rooted randomly at a hundred different nodes. From the records generated, we deleted those executions that were unable to generate a tree because the chosen root didn't have an outgoing link. The selected sites had different sizes as can be seen in the table in the top of Figure 4.19 and were selected according to their different structures, going from a personal website with a few hundred pages and links, to a highly connected blog with about three thousand pages and fifty thousand links.

Figure 4.19 shows the average percentage of LMLs found by each algorithm on each website. These values have been calculated according to the Total LML value, which represents the desired number of LMLs (ideally $(n-1)$) minus the number of non-relevant LMLs. We define a *non-relevant LML* as a link that points to a page that has never been part of a navigation sequence, although it could have been visited through a search engine or a bookmark. These edges are considered irrelevant, as none of them reported traffic to its target node. As can be seen in the table of Figure 4.19(b), results changed according to the connectivity of the websites. Nevertheless, it is clear that the MB algorithm always shows more LML than the Breadth First Search approaches, improving the results by an average of 20%.

From this experiment we can conclude that the MB algorithm produces hierarchical visualisations that improve the accuracy of other methods proposed in terms of number of LMLs shown, as can be seen in Figure 4.20(a), where the structural tree based upon the W-BFS algorithm and the usage tree are shown to visualise the same website. However, we analysed, with an expert analyst, a website using the explorative capabilities of WET, and realised that our approach had a very important drawback: as the system takes into account all users' sessions, it generates a tree containing all the nodes in the webgraph, locating them at their best position considered by the gathered LMLs. However, their depth in the tree does not reflect the number of clicks needed to reach the nodes from the root node. Hence, the hierarchy by itself is meaningless. Figure 4.20(b) illustrates this problem. The highlighted node is an important landing page in the site. However, this page is rarely visited from the root node of the image (in this case, the home page of the site). Thanks to this visualisation, we formulated a new hypothesis shifting the paradigm according to the sought outcome: the MB algorithm might be useful to represent the most common paths performed by the users that start their navigation at a specific page, which will be the root of the hierarchy. To prove our hypothesis, we ran another experiment that consisted of running the algorithm upon all the nodes in a site that were a landing page, at least once. This operation was repeated for the five different sites used in the previous evaluation. Results show that extracted



(a) The usage tree (right) reveals an important edge (in blue), that the linking and brushing system of WET allows the user to see in the structure tree (left). Such an edge would have been difficult to detect otherwise.



(b) The filtering system helped us to discover an important landing page (highlighted in blue), that distributes its traffic to 4 pages. The page, located at level 3 of the hierarchy, is not usually reached through the paths performed from the root node, which is the home page of the site. This fact suggested that we only consider paths that start at the root node, to reach more meaningful trees.

Figure 4.20: The exploration of the different visualisations helped us to evaluate the usefulness of the different algorithms for extracting hierarchies. Right images show that the algorithm reveals important links not visible with the W-BFS.

hierarchies provide an average of 99% of the LMLs existing on those navigation paths, which means that they are an accurate summary of the graphs made up of all the paths that started at every landing page.

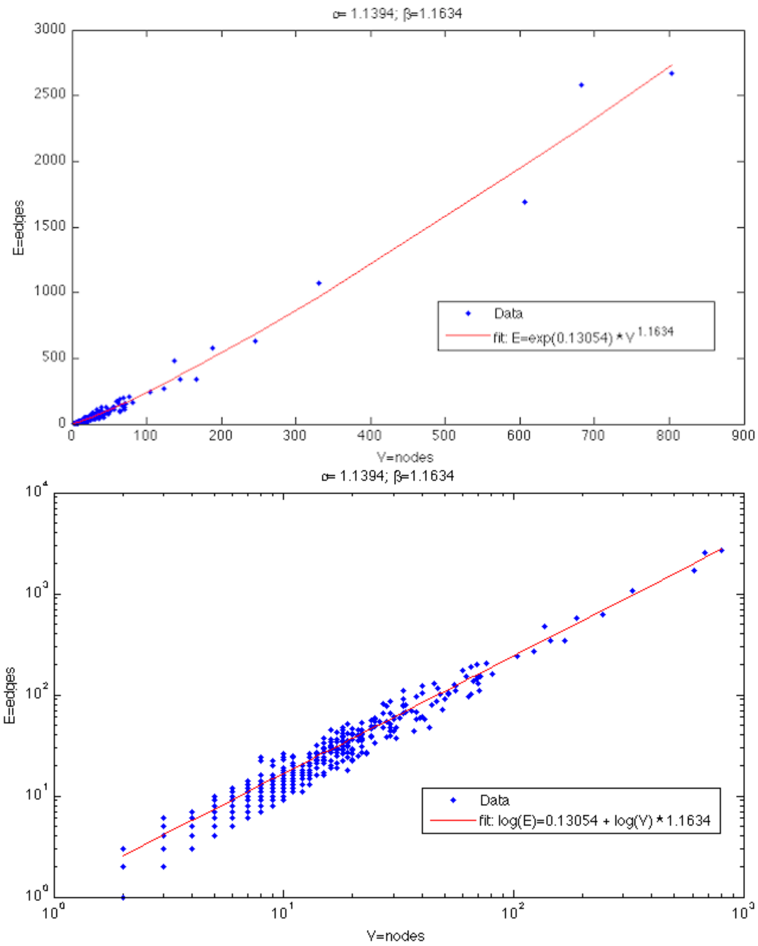


Figure 4.21: Relation between nodes and edges of the graphs to be handled by the Edmond's algorithm. The fitting curve represents the growth of graphs with more than 100 nodes. Bottom plot is in log-log scale.

Taking into account that the algorithm runs in $O(EV)$, we characterised such graphs to see if the algorithm is computationally feasible to get responses in a reasonable response time. We used usage data from six months of those already mentioned 5 websites. The extracted model presents a polynomial curve corresponding to the function $E = 0.47 \times V^{1.42}$ in graphs with less than a hundred nodes. However, the degree of the function decreases in bigger graphs (i.e. graphs with more than 100 nodes),the function being $E = 1.14 \times V^{1.16}$, showing a quasi linear growth, as can be seen in Figure 4.21.

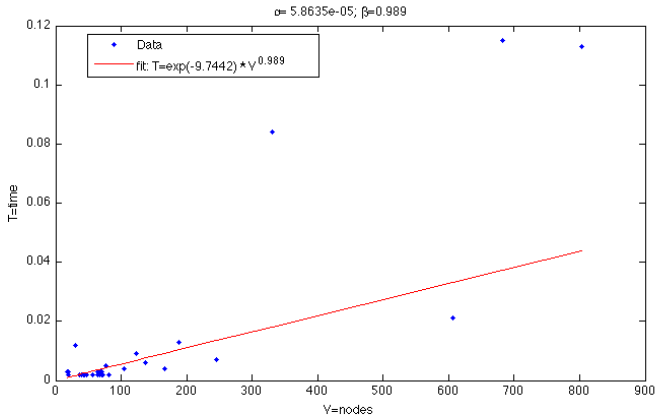


Figure 4.22: Time performance of the Edmond's algorithm applied to the graphs formed by users' paths starting at a specific page.

We finally characterised results of time performance of the Edmond's algorithm upon such webgraphs. As can be seen in Figures 4.22, once outliers have been removed, the algorithm presents a linear behaviour (being the fitting curve $T = \exp(-9.7) \times V^{0.989}$) among the available graphs which confirms its adequacy considering the size and topology of the graphs that it has to deal with.

Again, the linear behaviour described by the data samples of time performance show that, considering the size of the available webgraphs generated by user sessions in a wide and diverse set of websites (with a mean of 11.8 nodes and 24 edges), the scalability of such algorithm is acceptable.

The main benefits of our approach are:

- The generation of a summary of the most common users' paths from a specific landing page to every exit page, which, taking into consideration that the sessions' length in a site is fairly limited, generates small trees easier to layout and visualise.

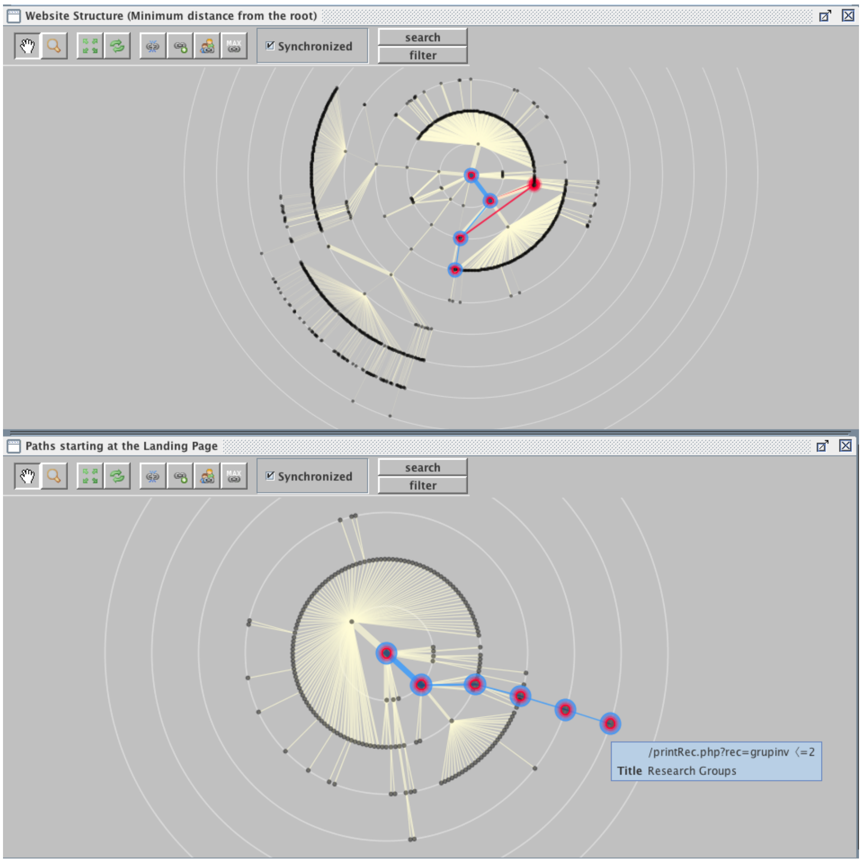


Figure 4.23: The coordinated highlighting enables the user to explore the shortest path to reach a node from a specific root (top frame) comparing it with the real users' behaviour (bottom frame). Considering the user sessions that started at a specific page (the root of the hierarchies), the usage tree reveals the most common paths performed from the route to every existing ending point in such sessions. This image reveals that users made five steps to reach a specific page, while the website structure reveals that the same node can be reached with only three steps.

- Node depth becomes meaningful, as it represents the average number of clicks that the users make to reach a page from the rooted node.
- The coordinated highlighted environment provided by WET enables the user to compare the shortest path to reach a node from a root page with the real path performed by the users, as can be seen in Figure 4.23. This feature was specially well rated by domain analysts that took part in the formative evaluation described in Section 4.5.

4.5 Evaluation of the System

We conducted a formative study with domain experts, aimed at evaluating both the domain level (stage 1) and abstraction level (stage 2) from the nested model presented in Section 2.6.

Andrews defined in (And06) the concept of formative evaluations as “tests that involve a small number of users using an interface to gain insight into which problems occur and why they occur”, and Munzner (Mun09) stated that this kind of evaluation is “intended to provide guidance to the designers on how to improve a system and answer the question ’can I make it better?”’. Munzner also stressed that this kind of assessment may support the evaluation of the domain and abstraction levels.

The test was conducted with five web analytics experts who analysed a website through a set of 10 tasks. The website used was www.infovis.net, which at the time of the experiment had nearly 1.000 pages, almost 5.000 links, and usage information from six months, having almost 50.000 user sessions. The dataset was especially interesting as it had a clear structure, having a mirrored structure due to the multilingual nature of the site. Users were given 15 minutes training with a real dataset, and spent almost two hours with WET analysing the dataset. Tasks were defined with the webmaster of the website, who used WET to analyse his own site, and were defined to allow the users to use all the visualisations provided by WET. Although this makes the results of test not generalisable, it helps to focus users’ attention on specific problems that ended up providing interesting feedback in order to validate the domain and the abstraction level of our approach. Tasks can be found in Table 4.2.

WET was initially loaded with the visualisations described in Chapters 3 and 4, which included two visualisations of the website structure (the radial tree and a treemap), the usage tree and the session graph. Users were asked to verbally express their thoughts and analytic strategies. The whole session was video recorded using screen cast software that recorded user actions as well as the analyst comments.

4.5.1 Evaluation Results

The test allowed the users to perform an in-depth assessment of a real website, enabling them to use the main features of WET in detail. From

Task	Description
T1	How many clicks are needed to reach the furthest pages from the home page?
T2	Locate the most visited page in terms of 'page views'.
T3	Have a look at the broken links existing in this site, and decide which page would be the first one you would fix.
T4	The website you are visualising has contents in Spanish and in English. Show the most used links in the home page of the English version of the site and discover what pages they link to.
T5	Which version of the website receives more visits, the English or the Spanish one? more in-links.
T6	Use the search tool to locate one of the pages of your website using its url. Now, visualise how the rest of the site is organised around it.
T7	Discover the shortest path to reach the page that the evaluator will give you from the main home page of the site, and discover if this is the main trend followed by the users, or if they make more complicated routes.
T8	The page XX is the main conversion page of this site. Explore the main paths performed by the users who reached it.
T9	Which is the most important landing page in this website? Discover how many clicks are needed to reach the conversion page from our landing page.
T10	Locate the page XX. Which is the most common path performed by the users that landed on this page that reached the conversion page?

Table 4.2: Tasks performed by the users during the final formative evaluation (URLs used in this study have been removed to preserve website privacy).

the recorded videos and in-situ observations, we saw that users understood all the visualisations and the data they represent. However, as the time and tasks completion increased, they tended to forget the data they were looking at, having doubts about how to perform the requested tasks. This issue suggests the need for more explicit information in the visualisations frame, as we observed that the titles of 'Usage Tree' and 'Sessions graph' were misleading, mainly in terms of the type of usage data they contain. As a result of this, we proceeded to change the title of the visualisation frames providing more detailed titles. In addition, a typical error was related to the fact of having the same visual metaphor (the radial tree) used in two different contexts: the website structure and the usage tree. From this, we learned that, in order to improve the results of the evaluation, it would have been more useful to provide predefined configuration settings regarding the type of analysis to be performed.

In T1, users stated that they do not usually take into account the number of clicks needed to reach a page from any other as current tools do not provide this information. However, they stated that this feature would be very interesting to discover the 'compactness' of the site and to visually discover those pages that need more clicks to be reached from every landing page than the average number of clicks performed by the users in the site, assisting in the discovery of pages that might never be reached.

In terms of T3, analysts stated that the superimposed visualisation of broken links upon the website structure (see Figure 4.14) allowed them to form more complex hypothesis which are difficult to formulate with current tools. While these tools provide only information about the number of broken links that each page has, our approach let the users incorporate more complex reasoning when deciding which page they would fix first. Besides from stating that they would solve the page with the most broken links, an analyst stated: "seeing the data in that way, I would rather start solving the pages nearest the home page, or near the conversion page rather than the ones with the most broken links". Moreover, analysts argued that the rest of the visualisations would help them to foster their attention into the most navigated routes, in order to start fixing the pages that would affect users' navigation.

T5 was really difficult to achieve, and denoted the need to incorporate analysis capabilities into the tool, such as tables delivering information regarding the visualised information. This particular task helped us to understand that, although WET might be very useful to locate particularly interesting information, it is very important to provide table based information which is easily exportable so further statistical analysis can be carried out.

One of the most well rated interactions available in WET was the linking and brushing feature between the structure and the usage trees,that was extensively used in tasks T7, T9 and T10. The combination of both visualisations enabled us to discover and compare the most efficient paths to the nodes with the real one performed by the users. According to the analysts, this information is very difficult to get hold of with current approaches.

T8 was usually tackled with the session graph, which was also very positively rated as, according to the analysts: "with this visualisation one can see the ecosystem of pages involved in the routes to a specific content". In a sense, the usage tree, the session graph and a funnel chart could be complementary visualisations as all of them provide solutions to different problems. On one hand, the funnel chart helps to reveal how many users go through a set of predefined steps needed to reach a desired goal, hiding information on what the users did before reaching every step. The usage tree, on the other hand, helps to reveal the most common routes performed by the users from a specific landing page, representing several funnels at the same time, mainly allowing the analyst to discover the average number of clicks made by the users to reach all the exit pages. The session graph, however, appears to be the most complex yet powerful approach. Its ability to visualise the paths between any given set of pages presents a new possibility for analysts, who do not currently have any tools for digging into users' paths in fine grain detail. The session graph may help to discover different routes that may converge to finally reach the same page, or to see how users that reach a specific page split to visualise other pages containing related information.

Overall, one of the best rated features of WET was the possibility to highlight the conversion page in the website, which according to the analysts, is the most important page in the site. Thanks to this interaction, the conversion pages pop out very easily, enabling the analyst to discover where it is located in all the different visualisations.

This evaluation brought up a new research question: do all the sites of the same type share a common structure? If so, it would be very useful to depict such a common structure to see the differences with the existing one, allowing the web designer to discover the main differences.

A further improvement of the tool should definitely consider the incorporation of the time dimension into the visualisations, enabling the user to discover changes in the data over time.

In the final post questionnaire interview, all the users stated that the interactive visualisations provided by WET would be very beneficial to their work, but they argued the need to integrate the tool with the data coming from analytic tools such as Google Analytics or Site Catalyst.

Usability issues Besides providing valuable qualitative feedback to understand the adequacy of the visualisations to the analysts needs, an important goal of formative evaluations is to aid the location of usability problems. Below we provide a list of the main usability problems we found in our system:

- *Node overlapping*: The overlapping in the Radial Tree was annoying for the participants. It often took too long to select a single node, or to realise that the selected node was the desired one.
- *Difficulty in remembering page position*: Although the animation performed when the root node of the hierarchical visualisations is changed, users tended to get lost when there were no metrics that visually helped them to identify a page in the visualisations. Therefore, we suggest that locating labels with the URL or the title of very specific and relevant pages may dramatically increase the location awareness of most of the pages in the visualisation.
- *Node selection*: We found an important usability issue that involved the unhighlighting of previously selected nodes when users performed the panning action. This issue could be solved by using a double-click approach for clearing the group of selected nodes.
- *Need of more visual clues*: As stated before, as the tasks got more complicated, users started forgetting the meaning of the representations. Hence, we believe that this could be solved with more visual cues that indicate what is being displayed in every frame.
- *Vocabulary problems*: The label associated with link usage was "Frequency". However users did not understand this term and suggested "Click" as a more appropriate word. Moreover, the name was confusing. For instance, "Navigation External Views" should be named "Organic Traffic".
- *Filtering System*: The filtering system changes the opacity and disables the interaction capabilities of items that do not match the filtering conditions. However, as they are still visible, some users got confused as they wanted to interact with them. One possibility would be to reduce the opacity further to better reflect that the item is not available. Moreover, if a link is active and one of its connected nodes are not interactive, this highlighting system will highlight both

of them. This issue confused users, letting them think that in fact both nodes where available for interacting.

These problems were addressed before conducting the next evaluation, which is covered in the next chapter.

4.5.2 Lessons Learned and Study Limitations

Nevertheless, the conducted thinking aloud protocol provided valuable feedback as domain experts could explore in detail the different visualisations provided by our system. In order to avoid big disparities on the gathered feedback we used the same dataset with all of them. While this fact benefits the comparison between the behaviour of the users with the tool, it prevents results from being generalisable. However, the fact that the dataset was real, and that users extracted the same results without the need of other analysis tools suggests that their reported benefits might be transferable to sites of, at least, a similar size.

As we have introduced before, WET was initially loaded with four different coordinated visualisations at the same time to evaluate if the users are able to discern the different possible benefits provided by each approach. This fact usually overwhelmed users, making them get lost with the different visualisations. We suggest that, at this level of evaluation in multivisualisation systems such as WET, the tasks should incorporate notes such as "use the X visualisation to discover...".

During the experiment, 15 minutes training was giving before a 2 hour session. In our experience, it is important to avoid user frustration as it decreases their confidence with the tool. Moreover, users tend to forget features of the tool not used in the very first tasks. To overcome this problem we suggest that longer training sessions should be accompanied with shorter experiment sessions which would benefit the final result.

We also learned that during the analysis sessions, it is very important to provide the user with information about the context of the data to enable the generation of more informed decisions. For instance, indicate the number of pages that are being visualised, or the percentage of the whole traffic that the visualised nodes hold. This will help to identify if the problems or inconsistencies detected through the exploration are really important or not.

We also suggest maintaining an as lengthy as possible discussion after the accomplishment of a very limited number of tasks with the users. It is also interesting to point out that some of the most useful user suggestions came from discussions made a few days after the test. At that time, users tended to remember what they considered to be the most important novelty in the system.

4.6 Conclusions

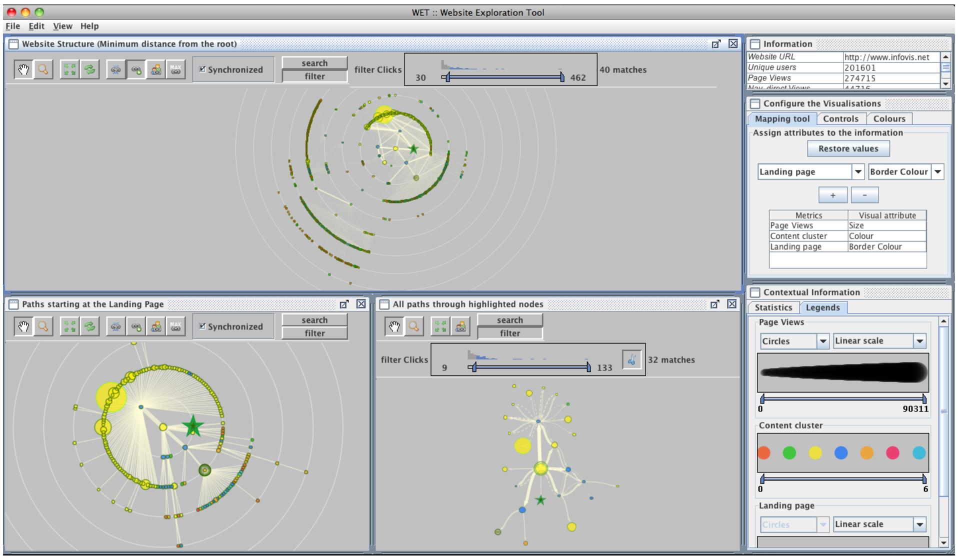


Figure 4.24: Screenshot of the WET visual system showing the three different visualisations available. All the visualisations use the same visual encodings: upper left window shows the radial tree that represents 'Structure Tree', bottom left shows the 'Usage Tree' and bottom right frame shows the 'Sessions Graph'. The green star represents the conversion page of the site, facilitating its tracking along the three visualisations.

Evaluating the usability of a website is a problem that goes beyond the analysis of statistics that mainly provide information about website usage. However, usability experts, information architects and web analysts are still struggling with tables full of statistics that end up providing little insight into how a website is used.

Contrary to current Web analytic tools, WET uses a context-based environment that enables the representation of web metrics on top of visual abstractions of the structure and usage of a website, encoding them as visual attributes from nodes. Moreover, WET offers a coordinated environment that provides different synchronised visualisations as well as interactive techniques to interact with the data.

In terms of the visualisation of the website structure, we have developed the 'Structure Tree', which is a radial visualisation whose novelty relies on the modification of the well known Breadth First Search algorithm. Our approximation is able to disambiguate coexisting parents of a node in the same depth, improving the number of selected relevant links to form a hierarchy. The system also incorporates interactions to recalculate the site's hierarchy starting at any page on the site, enabling the user to understand the number of clicks needed to reach the rest of the pages in the site. This approach may represent a major advantage for website designers as it enables them to visualise the structure of a website at a glance considering any possible entry point. We have also seen that this approach allows the user to easily discover possible design errors in the hyperlink structure of the site, such as the location of pages located at too many clicks from the desired root node.

In terms of the visualisation of users paths, we have developed two different solutions. The first one is a graph layout that depicts all the different routes that pass through one or more selected nodes in the site. Implemented dynamic filtering techniques available in scented widgets enable the user to filter non-relevant links, helping in the extraction of meaningful usage patterns that reveal main users trends. This visualisation has been especially well-rated by a group of domain experts who stated that our approach can complement current web analytics tools, providing a more detailed and compact view of users paths with data which is very difficult to interpret.

We have also presented our second approach for dealing with users' routes, which is based on a maximum branching algorithm that extracts the most common routes performed by the users, starting at a specific page. Domain experts stated that the combination of this visualisation and the 'Structure tree', with the filtering and mapping capabilities of WET, may benefit the comparison between shortest paths versus real performed routes.

Finally, according to the request of expert web analysts, we provided WET with interaction facilities that enable the web analysts to mark the conversion pages of the site. Hence, users can easily identify such pages visually and annotate the conversion page in order to locate it more easily while analysing the visualisations.

An evaluation with domain experts has helped us to understand the potential of our visualisations, and to validate them in the two very first layers proposed in the nested model presented by Munzner (Mun09).

Work from this chapter has been published in (PD07; PC08; PCBYD+09; PCBYD10).

Chapter 5

Visualising Virtual Learning Environments

In this chapter we present the integration of the WET system to support the analysis of Virtual Learning Environments. Such integration was conducted within two case studies, where real analysts used our tool to explore data from different learning platforms. Furthermore, we will describe the evaluation methodology that we followed, as well as the outcomes obtained by the analysts. We will finally report the lessons learned during these long term evaluations.

5.1 Research Problem

Ramsden (Ram03) suggested that integrated monitoring of student experience is an essential component and a 'minimum standard' of online learning provision. In fact, usage access data could be valuable for instructors to gain a better understanding into how students use the online resources and evaluate how effective they might be. However, in practice, it is rare that teachers use such data to inform their teaching or pedagogical goals, preferring traditional face-to-face interaction or feedback evaluation forms at the end of the teaching term. This is strikingly different from the growing trends in e-commerce in which the use of advanced analysis and visualisation techniques to explore usability and effectiveness of web material has become a cornerstone of business strategy to understand customers' behaviour.

The lack of applied examples of web usage mining in education is partially caused by the clunky interfaces implemented in VLEs to access and view data, and partially by the difficulty of finding the expertise in data mining among teachers and instructors. There is certainly a need to make available simple and intuitive tools to improve pedagogy and students' experience. Furthermore, typical web analytics approaches and applications to study web usage data are tailored for the business needs and highly focused on key performance indicators (KPIs) (LTT09). When considering a closed-access system within an educational setting, in which the VLE is either complementing the course material in a blended delivery of instruction or provides the interactive space in which teaching occurs, as in distance learning, the KPIs used in business models lose meaning as usability and students' experience are central.

WET has been introduced as a visual tool enabling the user to explore web data through interactive visual representations of the structure and usage of a site. The research presented in this section has two main goals: the first is to explore the potential and the benefits of a visual mining tool in educational online provision of content using standard VLEs. The second is to provide further evidence in support of the usefulness of WET in exploratory data analysis providing a detailed evaluation of this tool with two long-term case studies.

In this section we will first review the current techniques used for analysing. Afterwards we will discuss the evaluation methodology we used in this study to finally describe the two use cases where WET was applied along with a subset of the main insights gathered by our tool. We will finally conclude with a discussion on the benefits of WET in the assessment of VLEs and suggest improvements in the design of visual web mining systems.

5.2 Related Work

The web based nature of VLEs makes data mining techniques suitable for gathering information from web server logs that contain trails of student and teacher activity within the platform (RV07; CVN+07). Specifically, web mining techniques, which include the analysis of the structure, content and usage of a site, can be used to discover and model the success of pedagogical strategies and ease of use of the learning platform.

Early attempts to use web server logs to explore students' behaviour using online learning material have been reported in different studies with varying degrees of success. For example, Ingram (Ing99) used logs to evaluate utility and usability of instructional websites. Crook and Barrowcliff (CB01) carefully logged students' engagement to demonstrate the value of ubiquitous computing on campus. Hardy and colleagues (HBM+06) monitored students' access to material provided via a VLE across different disciplines to identify useful behaviour patterns. However, these studies provide a snapshot view of usage, and much like web analytics, give only a descriptive overview of activity.

There are many web analytics tools that generate a wide variety of metrics such as number of page views or average time spent by the users on the site that help to understand website usage. However, these statistics usually provide little insight into usage due to the lack of adequate and comprehensible visualisations that may potentially assist the process of making sense of the vast amount of data available. Information Visualisation (infovis) techniques have been applied in an attempt to overcome such a problem. The most common approach has been the usage of node-link diagrams where nodes represent content pages and edges represent links between them. Different layouts have been used for representing either the organisation of the website or the routes followed by the users when navigating through the site (CS99; HL01b; CPM+98; KE02).

Although there have been several approaches for providing visualisations to help with the understanding of websites usage (see Section 4.2), in fact, a few cases can be found demonstrating the applications of these techniques into the e-learning domain. For example, (MBC+05) used a file-navigation-like tree for digging into the details of student sessions. (SSW05) implemented the DMASC tool, which provided fine-grain details of the paths of individual users through the learning space.

5.3 Evaluation Methodology

To validate the domain and abstraction levels of our tool in the context of analysing VLEs we conducted two case studies with real scenarios in which WET was used by at least one analyst per project in the assessment of two e-learning platforms. As indicated earlier, the need for visualisation tools in education is demonstrated by the fact that in both cases, the first engagement followed an expression of interest from the project leaders in using visualisation tools to aid their assessment.

The exploratory and interactive essence of InfoVis systems makes it difficult to evaluate their effectiveness and the usefulness of the visual outputs with classical methods such as controlled and artificial experiments (Car08). In that sense, Perer and Shneiderman stated that "telling the users which tasks to carry out is incompatible with discovery" (PS08). In 2004 Plaisant (Pla04) challenged researchers to reformulate their evaluations to capture the exploratory nature of InfoVis systems and later contributed to the Multi-dimensional In-depth Long-term Case Studies (MILC) (SP06) in an attempt to overcome some of these issues. This methodology enables the user to conduct more naturalistic evaluations with qualitative assessments based on the observation of real users engaging with real tasks using the target tool during a long period of time.

Following the guidelines proposed in (SP06) and (PS09), the main steps that we applied in the MILC evaluation can be summarised as follows:

1. *Initial interview and experiment setup*: a first interview enables the user and the evaluator to define the goals of the analysis according to the capabilities of the system. Such goals will be used afterwards to assess the success of the analysis process.

2. *Training*: extensive training must be provided for the participants so they can understand the functionalities of the system.

3. *Early use*: participants install the tool in their workplace and get the required datasets formatted accordingly to the needs of the system. Continuous contact with the evaluator occurs to accommodate the tool to the needs of the user.

4. *Mature use*: it is expected that most of the outcomes of the analysis should come during this phase, where the participant is already skilful with the tool and has a better idea of the type of analysis that can be performed. Ethnographic types of observations should be performed.

5. *Outcome*: a final interview must be conducted to review the findings and compare them with the goals defined in the initial meeting. Participants should provide thoughts and conclusions about the potential of the system as well as suggest improvements for future designs.

During the analysis process, participants were encouraged to annotate and provide feedback about their findings or frustrations, as well as to capture images when discovery occurred. Moreover, we also had the opportunity to collect video recordings of analysis sessions in the AutoLearn project, as we had the opportunity to install screencast software in the analysts' computers. Such videos helped us in the process of reviewing the behaviour of the users with our tool, and contrast obtained results.

The interviews, communication and analysis tracking of the e-STUB project were performed entirely online through voice calls (Skype) and IM sessions using Google Wave. The latter was particularly interesting because it allows for a seamless combination of real-time and asynchronous exchanges.

Finally, in an effort to simplify and reduce the cost of analysing all the available videos, and to better understand the actions performed during the analysis we developed a logging system that keeps track of all the interactions performed with WET. Therefore, each interaction with the system leaves a trail of the form showed in Figure 5.1.

Figure 5.1: Parts of a record generated by WET's logging system.

We define below the information contained in each section:

- *Timestamp*: Records the time when the action was performed
- *User id*: Uniquely identifies the user of the system. This is especially useful when requiring a user name and password to access the tool.
- *Source*: Taking into account that WET provides a multiple coordinated environment, it is important to understand in which visualisation the action occurred.
- *Action Id*: We defined different types of actions to simplify its further analysis, identifying five different action types:
  - *Generic action*: These are a set of interactions such as zooming and panning that are available in all the visualisations.
  - *Node action*: When clicking to a node, WET shows detailed information in the Information Panel. Moreover, there are specific interactions that enable the user to highlight all the in- or

out-links of a node, or to set the node as the focus of the visualisations.

  - *Metric action*: One of the main characteristics of WET is that it enables the analyst to map a specific metric with any of the available visual attributes. These kinds of actions have been defined as metric actions, and may help to discover which have been the preferred mappings and their associated metrics.
  - *Session action*: This type of action registers the log in and log out actions to the system, to help define analysis sessions.

- *Action*: It contains the name of the performed actions such as 'pan', 'zoom' or 'click'.
- *Action parameters*: Each type of action may have a set of associated parameters. For instance, this section stores the pairs of 'metric/visual attribute' assigned with the mapping tool.
- *Visualisation*: This section of the log provides information on the state of every single visualisation in the system, providing information on which visualisations are actually visible and their size and positions.

Generated logs were analysed after the evaluation process and were used to guide part of the interview with the users.

5.4 Case Studies

The following sections are dedicated to the description of the main goals of the AutoLearn and the e-STUB projects. We will define its analysis needs as well as the data integration processes we followed to integrate project data to WET. Furthermore, we will report the different stages of the MILC evaluation process including early usage and mature usage of WET, to conclude with the obtained outcomes.

5.4.1 AutoLearn Project

AutoLearn is a VLE to learn English or German as a foreign language developed using Moodle; AutoLearn incorporates automatic correction facilities that go beyond spelling and grammar checking based on Natural Language

Processing techniques. The main goal of the project was to study the use of automatically generated feedback in real-life instruction environments, which adds machine-learner interaction to the existing teacher-learner and learner-learner interaction. In terms of our work, the project sought the evaluation of the usability of the platform to understand students' satisfaction and to assess the pedagogical and linguistic aspects of the courses.

AutoLearn was used in 7 institutions across Europe, managing a total of 28 courses taken by 610 students during two periods of 4 and 2 months respectively. The whole project was divided into two phases: the first testing phase was dedicated to the implementation of the learning system in the first 13 courses. During this stage analysts aimed to improve the system through the application of an iterative evaluation-development methodology.

WET was used as the main tool in the explorative analysis of quantitative data extracted from log files. The lack of experience of the analysts in the web mining field generated no primary hypothesis, which emphasised the need for an exploratory tool for the discovery of outliers and behavioural patterns.

Data integration

Although WET incorporates automatic tools for importing data from the most common web server log files, the existence of proxies in some schools prevented the user identification through classical methods of log file analysis (CMS99b). Hence, we ported the Moodle database, which incorporates its own usage data with identified users, into the WET mining system.

We generated a hierarchy based upon the main structure of the project that can be seen in Figure 5.2. The provided structure had nodes representing assignments in the outermost ring of the hierarchy, enabling a histogram like visualisation when mapping the nodes' height with exercise-based statistics. Such histogram gets visually integrated with the hierarchy of the platform, which enabled the discovery of patterns and outliers, as can also be seen in Figure 5.2.

The developed quantitative metrics were based on the time spent per exercise, number of visits and number of interactions per assignment. The last one was especially interesting as it provided a measure of the number of times that a user interacted with the automatic correction module. Regarding categorical data, the analysts identified every node based on its type, differentiating quizzes, assignments, resources and automatically corrected exercises.

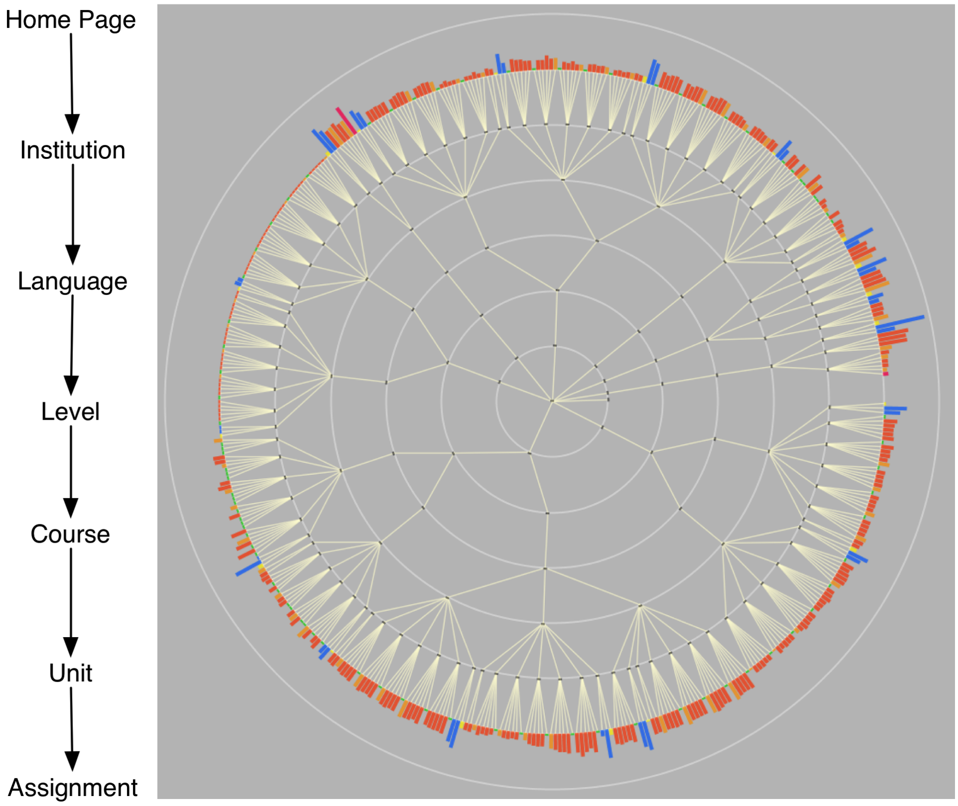


Figure 5.2: Hierarchy of AutoLearn structure. Leafs of the tree represent course exercises, that can be mapped according their specific quantitative and categorical metrics.

Early use

At the end of the testing phase, we provided the analyst with an XML file containing aggregated data from the 244 students that participated in 13 courses. WET incorporated a radial tree as well as a treemap to analyse the hierarchical structure of the platform, both of them with the same interaction capabilities such as dynamic filtering and visual mapping of metrics. We observed that analysts felt more comfortable with the radial tree than with the treemap as the hierarchical structure popped up more clearly. Moreover, comparing bar heights was easier than comparing the treemap squarified areas. At the time of this analysis there were no users' paths visualisations in WET, which limited the data exploration to the comparison of the different metrics with the visualisation of the structure of the project (see Figure 5.2).

During the analysis, each quantitative metric was mapped accordingly onto the height of the nodes, while colour was used to distinguish between categorical values such as level of the course (Intermediate or Advanced), language (German or English) or type of exercise (quiz, assignment, resource and automatically corrected exercise). Such mapping enabled the discovery of a usage pattern that revealed a decay in the visits as the lesson advanced, as can be seen in Figure 5.3. In a more detailed analysis, it was also discovered that there was a different slope considering students from advanced courses against intermediate ones.

This finding encouraged the analysts to develop and integrate more usage related metrics to compare in the second round of the project.

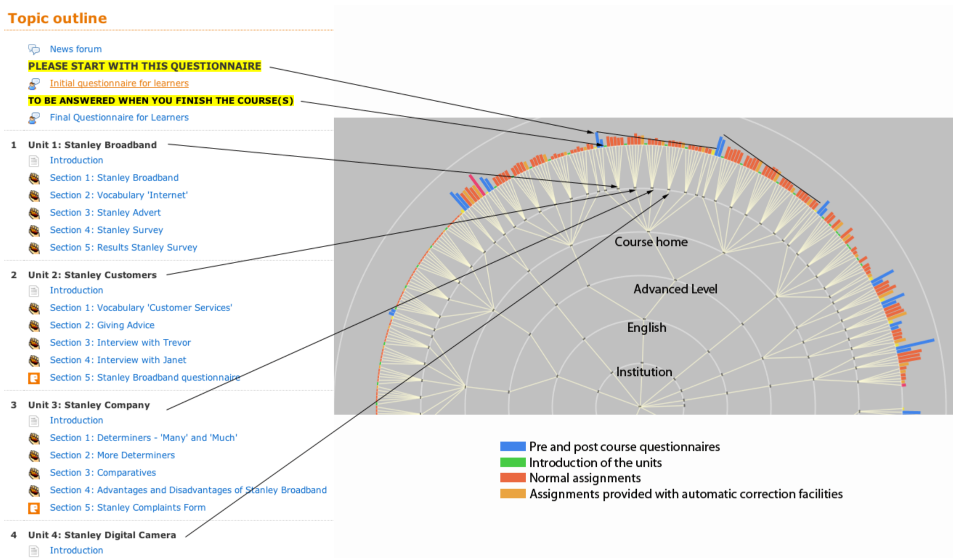


Figure 5.3: Detail of an AutoLearn course in the web interface (left) and in WET (right). Units and sections from the course are organised in clock-wise order. Colour coding shows the different types of sections within each unit, while size of the bars represent number of visits. The decay pattern has also been stressed in two different courses.

Mature use

The recorded analysis sessions allowed us to observe a dramatic improvement in task performance, proof of the learning curve of the analyst. When prompted, the analyst said: "I already knew how to use the tool, so I went straight to visualise what I wanted to see". We observed that the analyst developed a clear exploration strategy that followed the Information Seeking Mantra (Shn96): a very first overview of the whole structure using size and colour according to specific metrics helped to detect outliers and trends. Then, most relevant courses were filtered and explored in more detail.

In this phase analysts found more examples that supported the findings made in the early stage regarding exercise visits.

Outcome

While the overview obtained with the session graph visualisation (Figure 5.4) revealed the whole usage of the AutoLearn project, a closer look at the different courses revealed a pogo-sticking behaviour pattern that involved the absence of use of direct links between exercises within a course. The pogo-sticking term was introduced by Jared Spool in (HS09), and was described as "those times when a user jumps up and down through the hierarchy of the site, hoping they'll eventually hit the content they desire". Images from Figure 5.5 make apparent such patterns referred to by one of the analysts as the "flower visualisation". In the images, the main node represents the home page of the course, while surrounding coloured nodes represent the different exercises in the course. As can be seen, users tend to move back and forth between the exercises instead of moving from one exercise to the other. This pattern was interpreted as a sign that each activity was in principle self-contained and that all, or at least most of, the materials needed to answer the questions posed by the teacher could be answered without the need to navigate around. This interpretation is supported by a decision made after the first testing phase, which implied that all activities including reading and listening comprehension activities had to be implemented so that learners could have access to either the audio, or the video or the text as well as to the questions in one single screen.

In the final interview with the analysts, it was argued that the explorative capabilities of the tool and the "easy way to represent the hierarchical structure of the project" helped them in the process of generating new hypotheses

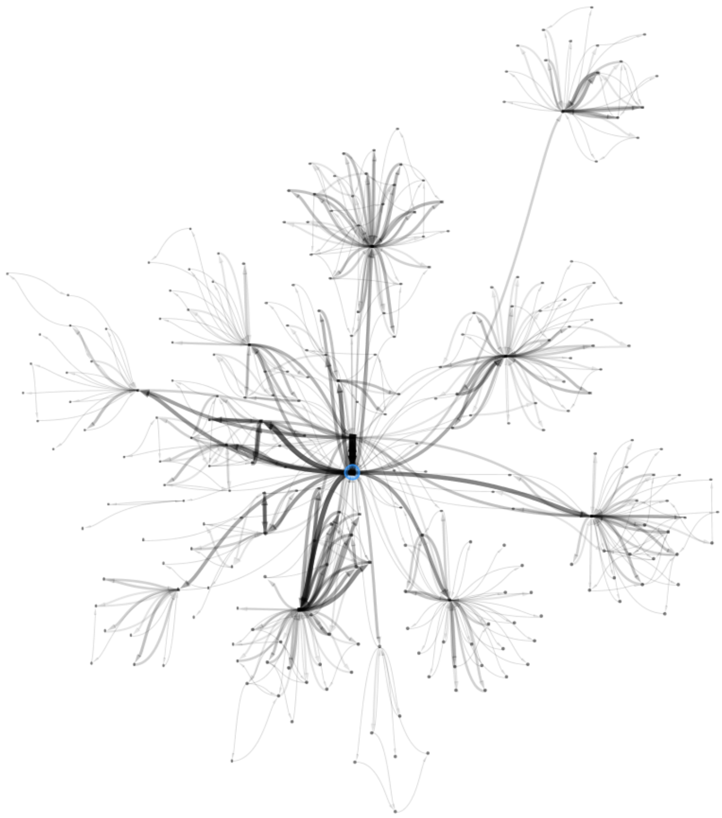


Figure 5.4: Visualisation of the sessions graph provided in WET, showing a graph corresponding to the student sessions in the season two of analysis. The visualisations reveal at a glance the different courses existing in the AutoLearn project. As expected, students can only navigate from the login page (central node) and the pages of the course he is enrolled in.

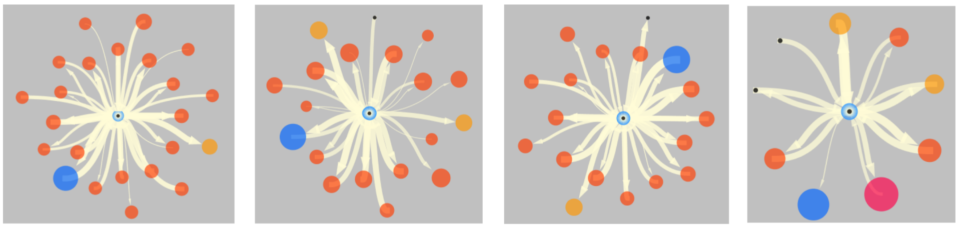


Figure 5.5: Aggregated users' paths from the 4 most visited courses of AutoLearn. The central node in the visualisations corresponds to the main page of the course which contain links to the different exercises, which are colour-coded according to its type.

which in the long term allowed them to perform a deeper analysis. Their main findings were also contrasted and confirmed with the qualitative analyses performed during the project.

We also noticed an extensive usage of image editing tools for annotating relevant screenshots after every discovery in order to incorporate them in the analysis report. This fact strongly suggested the need to incorporate basic annotation tools to the system, which at the same time may stimulate the collaborative analysis.

The final report of the project incorporated images extracted from WET to show the findings. More details about gathered insight from the analysis of AutoLearn may be found in (MEQ09).

Logfile analysis

The analysis of WET logs embracing 22 analysis sessions accomplished during the second analysis season helped us to discover how the visualisations were used.

As can be seen in the left chart of Figure 5.6, the structure tree was used way more than the other visualisations. The main factor that influenced this biased usage was explained by the main analyst as a lack of time to play with the usage visualisations. However, it is very interesting to notice that the session graph, only used 1% of the time, helped to discover the pogo-stick behaviour in the students.

Another interesting fact that can be seen in the right plot from Figure 5.6 is that only two visual attributes were used during the second analysis pe-

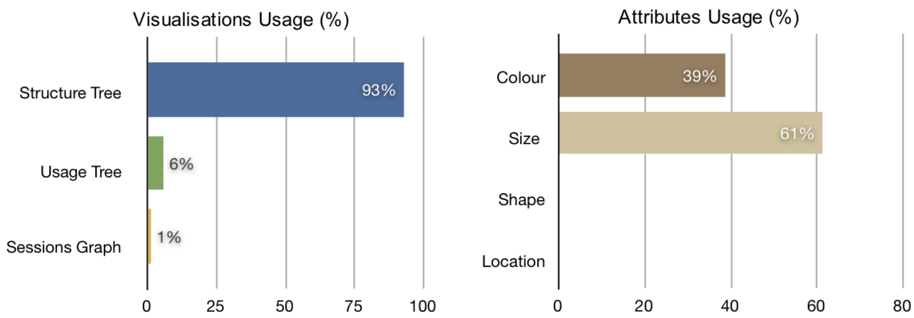


Figure 5.6: Statistics of the usage of the different visualisations (left) and visual attributes (right) in the AutoLearn use case.

riod. As we have explained before, we observed a dramatic improvement in the tasks performance, which limited the usage of features to those that satisfied the analysis needs during the first analysis season. In this case, the analyst preferred to only use size (mainly in bars of one dimension) to reveal numerical usage statistics while colour was used to distinguish between the different types of pages in the AutoLearn structure, using categorical metadata that helped in their identification as can be seen in Figures 5.2 and 5.5.

5.4.2 e-STUB Project

The main goals of the e-STUB project (exploration of Students' Tracking and User Behaviour) was to analyse in detail how students used online resources provided in blended courses via a VLE, and relate usage behaviour with both performance and a number of personality metrics.

In a prior quantitative analysis, Vigentini (Vig09) identified some compelling correlations between the amount of usage (characterised by frequency of access and time spent on resources) and academic performance. However, one of the key limitations reported was the inability to meaningfully visually represent the visitation patterns or attempting to define what a typical session would look like.

The dataset included over 2000 students enrolling on the foundation psychology courses over a period of 5 years at the University of Edinburgh.

This is an extremely heterogeneous sample in composition as the cohorts were followed from the first into the second year and the fact that not all enrolled students pursue a degree in social sciences, but take the psychology course as an outside subject (i.e. only 40% of the students are enrolled in psychology degrees).

Data integration

Web logs were obtained directly from WebCT Vista 6.0 (now Blackboard) using a powertool in the admin interface rather than the end user GUI. This generated a log of content units (pages with all the frames/graphics and scripts) organised by user and session. Such a format is an excellent starting point as it allows the user to skip some of the preprocessing steps normally required by the common log format. Each content unit was coded with metadata according to its function: this corresponds to tools such as content page, quiz, etc., actions such as 'view content page', 'search performed' etc., and abstractions such as core content, social activity, learning activity etc. Normally such information is not readily available in server logs, but the extra processing step to enrich the data gives a deeper understanding of usage behaviour. Users' goals cannot be easily extrapolated from the path taken, but by using the meta-data, intentions can be inferred based on the resources used. In fact, contrary to the concept of a conversion page in e-commerce, for an educational website, students could be driven by potentially competing goals within the same session (i.e. performing a self-test using a quiz as a learning goal as well as spending time on reading core content, but the intentions are different). Furthermore, although user data cannot be easily segmented with WET, the sample was stratified to provide single datasets based on the course year (i.e. first or second year) and performance (good performance, i.e. students with grades above 65%, and poor performance, i.e. between 35%-marginal fail- to 55% -satisfactory-).

Early use

We observed that the learning curve to attain a good proficiency with the use of WET was fairly rapid, as described by the analyst. This was predominantly because the project leader was already familiar with web analytics and some of the core concepts involved with the handling and processing of web log data in web usage mining. The steps to produce the data in the right format, however, were time-consuming and required a considerable amount of effort. The exchanges between the analyst and the developer were frequent and the aid of the communication tool (Google Wave) in storing and reproducing the exchanges was very valuable. After an initial live session in which the evaluator described the tool and its functionalities in detail, the work on graphs' generation took much more prominence and the analyst conducted the exploration using WET largely independently.

Mature use

Once the analyst had an opportunity to explore WET and its capabilities, the representation of meta-data and user information in the visualisation became central. The inclusion of categorical and user data required a further elaboration at the database level, before the generation of graphs. As the steps were standardised in the earlier stages of the project, the exchanges of data files and generation of graphs was considerably faster at this point, leading to an efficient workflow. WET needed a small upgrade to allow for the inclusion of all visual information available, but not previously used in a concurrent fashion; however this did not cause any side-effects in the data analysis allowing the analyst to focus on the exploration of data.

Outcome

Given the previous quantitative understanding of the patterns in the data, the analyst was mainly interested in the best possible visual representation of patterns. In study the analyst focused specifically on one cohort of students (students enrolled on the Psychology 1 course in 2007 and continuing into Psychology 2 in 2008) to demonstrate some useful insights generated from the use of WET.

For example, the analyst already knew that the students mainly focused on three core content pages (both in terms of time spent on resource and frequency of access): the home page, which is the first container displayed after login, the lectures hub (providing access to all lecture notes, handouts and core readings related to the teaching), and the tutorial hub (providing all material related to tutorials and practicals). These pages are prominent in the figures in which the size of items represents the frequency of use. The access to the social hub (i.e. discussion forums, internal mail system and chat) was less clear with aggregated data indicating uneven patterns across courses and years.

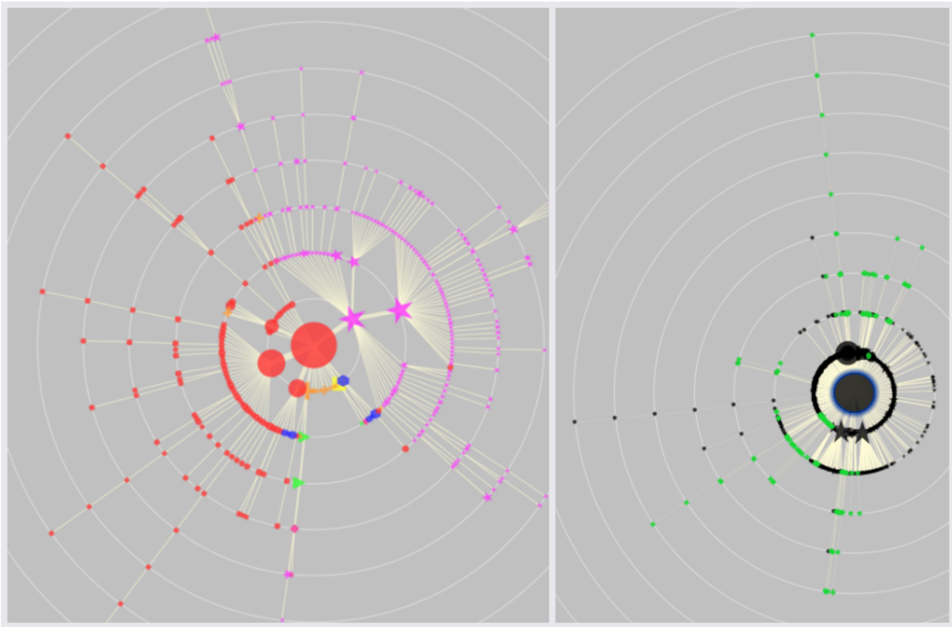


Figure 5.7: Paths and frequency of access in a 1st year course; most common paths from the homepage (left) and site structure highlighting shortest path to search results pages (green nodes).

The graphs (Figure 5.7 and Figure 5.8) demonstrate the expected patterns in accessing the core content. An interesting observation can be made in relation with the 'path to expertise' in using the system: students in year 1 (Y1, Figure 5.7), in general, display longer paths in their browsing sessions compared to Y2. Furthermore, as can be seen in Figure 5.8, low performers (right image) perform many more paths through social activity pages (coded as blue stars in the image), such as mail or chat ,contrary to what happens with high performer students.

In addition, longer visits in Y1 were expected from the quantitative data analysis as Y1 students tend to 'cram' visitation toward the end of the term, closer to the exams (see (HBM^{+}06)). Another possible demonstration of this 'novice', more chaotic approach, in using the VLE is offered by the access to the 'search' facility. The search is available from anywhere in the site and the results pages are dynamically generated appearing in the structure as if they were static. Figure 5.7 (right) shows that in the Y1 class there are a number of long branches (green nodes) indicating repeated searches beyond 3 clicks. Occurrences of the search results pages are a behavioural pattern rather than reflecting the structure of the site and its usability, because as explained in Section 4.3.1, the fact of generating the website structure based on access logs converts the structure tree into another visual abstraction for understanding usage patterns. A shallow occurrence like in Y2 (i.e. log in and find X) demonstrates strategic behaviour. In Y1 too many long branches and deeper occurrences in the clickstream could be interpreted as an index of disorientation.

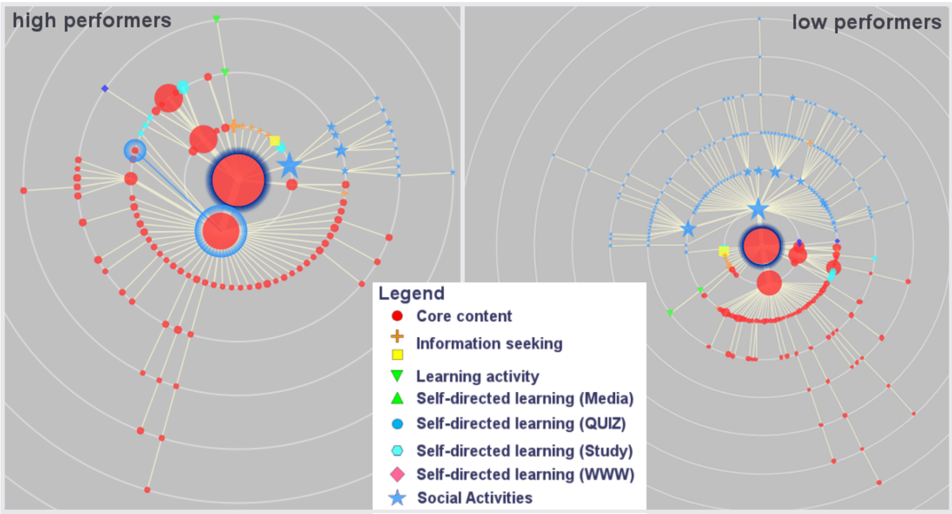


Figure 5.8: Differences between high and low performing students' most common paths in the 2nd year course.

The similarity of the visitation patterns in students classed as low performers in Y2 and the Y1 group was particularly interesting as it could be an early indicator of lower performance at the end of the year. The path to expertise in the higher performing students suggests increased efficiency in access, with shorter, frequent and targeted visits (i.e. the sessions are short for all types of content), whilst the lower performing group still shows broader and lengthier visits, more similar to the novice state in Y1. Y1 students also don't seem to differentiate between the core content and social interaction: patterns that remain similar in lower performing students moving from Y1 to Y2. This could be an index of distraction or an attempt to seek external validation.

Moreover, Figure 5.9 represents different images extracted from our session graph, representing different student groups that helped the analyst to demonstrate the differences in the modes of access of different types of users. Such visualisations were reported in the final project document, and were referred as the “site galaxy”.

Although all these observations need to be properly substantiated by statistical analysis of the data, the visual interaction with the navigation paths is essential to explore for meaningful patterns.

Logfile analysis

The analysis of WET logs revealed a more evenly distributed usage of both, visualisations and visual attributes of the system (see Figure 5.10). The difference with the behaviour seen in the AutoLearn project can be explained with the different analysis needs of the projects. While there existed only one dataset to be examined in the AutoLearn project, and in general, very clear goals defined after the early use that involved the comparison of the different usage metrics in the different courses, the goal of the e-STUB project involved more exploratory tasks across many datasets that represented different segments of the data.

To the question of why the analyst used the sessions graph visualisation less, he answered that the usage tree was more readable and provided very interesting usage patterns. However, the sessions graph, or “site galaxy” as referred to by the analyst, also helped to reveal unexpected patterns. We observed that logs contained no records regarding the usage of the filtering widget that facilitates the discovery of main navigation trends in such visualisations. This fact suggests a usability problem, as the analyst forgot the existence of this functionality.

In this case, colour and size were the most used visual attributes. However, it was interesting to discover, after the interview with the analyst, that the location attribute (which sorts the sibling nodes of a branch in the tree according to a specific metric) was used to bring together all the different pages, as can be seen in Figure 5.8.

5.5 Lessons Learned and Recommended Features

Although WET has provided ways to discover relevant patterns of the data, such patterns cannot be easily studied with it. Therefore, for visualisation

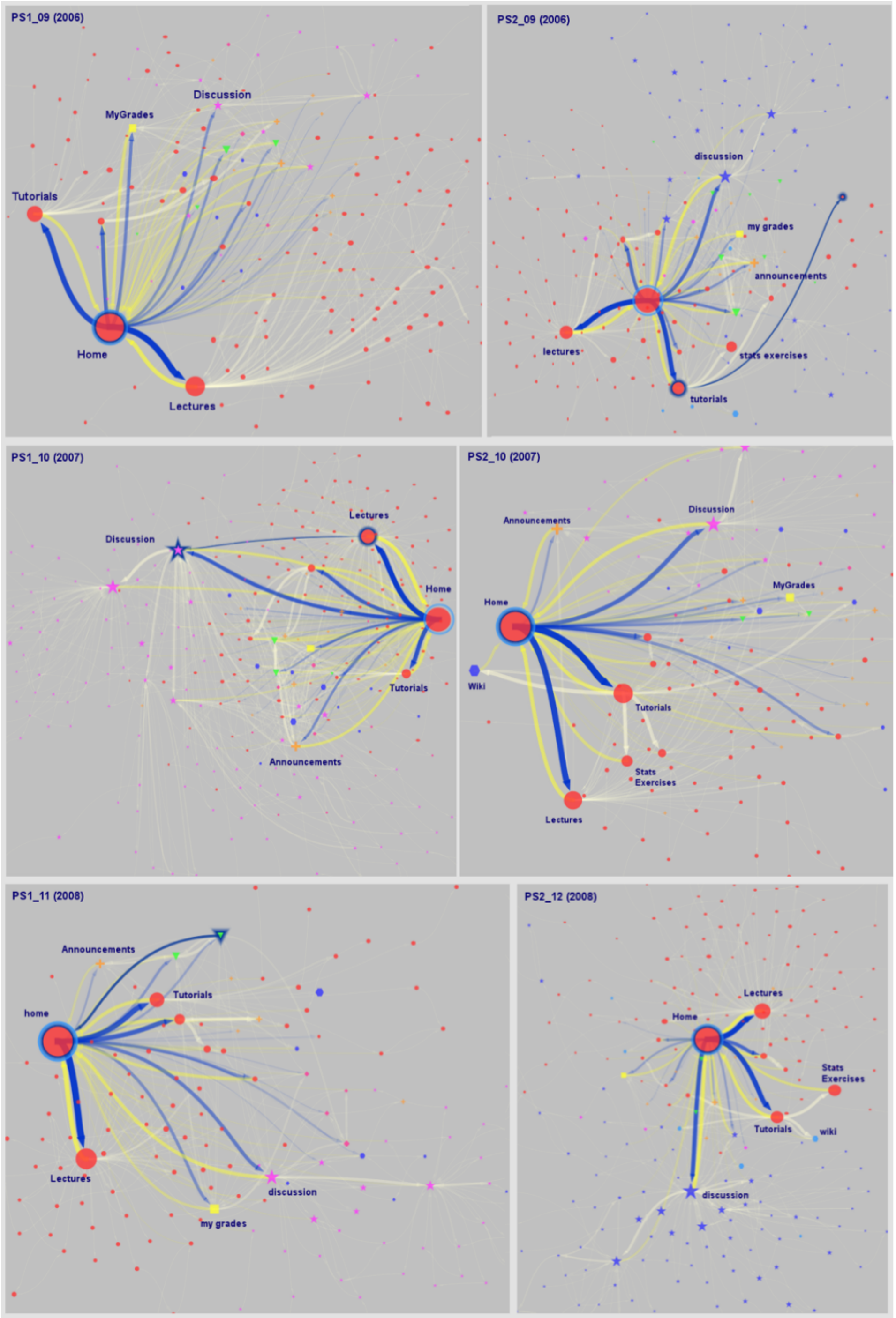


Figure 5.9: Different visualisations of the "site galaxy" with different segments of users provided by the analyst of the e-STUB project. Each image represents the session graph of the students sessions, classified in different groups. The visualisations enable the user to discover different behavioural patterns.

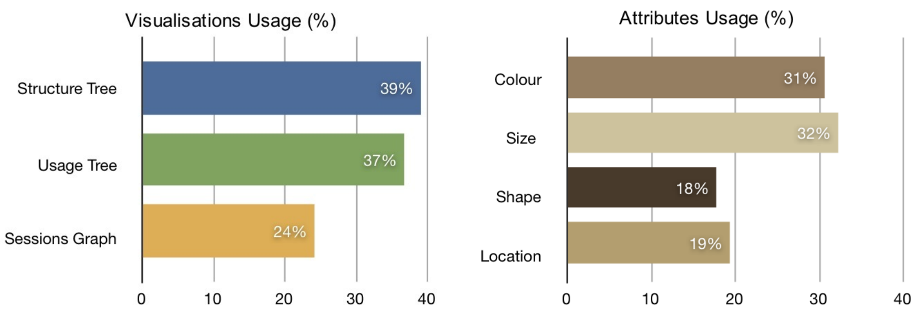


Figure 5.10: Statistics of the usage of the visualisations (left) and visual attributes in the e-STUB use case.

systems such as WET to be efficient, it is essential to provide them along with exporting tools able to generate readable files with the information that is being visualised. Moreover, qualitative observations revealed that visualisations should be accompanied with data tables.

Another relevant enhancement of the tool to be considered as future work should incorporate the ability to annotate the visualisations to facilitate collaboration, and the possibility to stratify the data on-demand, allowing a real-time calculation of metrics according to metadata intervals, such as time or type of users.

In addition, a very interesting and well rated capability of the WET system is its ability to incorporate any kind of metadata that can be taken into account during the data exploration, and which is difficult to be added in current web analytic packages.

Finally, we have seen the importance of developing MILC studies with tools provided with a logging system. The analysis of such logs may provide very useful information that can be used to guide the interviews with the analysts, helping to refresh users' memories as well as stressing the most used features. Moreover, we believe that the combination of a logging system with annotation tools may assist the automatic discovery of findings during the exploratory analysis.

5.6 Conclusions

The understanding of the usage patterns of teaching material offered via VLEs is a very important element that could provide valuable feedback to instructors and inform pedagogy in terms of both learning outcomes and teaching goals. However, web analytics and traditional log analysis tools are unable to provide a simple and accessible way to gain insight into the data with adequate tools for developing customised metrics.

Our work demonstrates how the use of WET provides a way to visually represent website data for the evaluation of the usage of two VLEs. Although we cannot claim that WET can be considered better than any other currently existing tool for analysing data from VLEs, we have seen how in both case studies our tool supported the discovery of uncovered and unexpected patterns and may potentially provide a valuable asset for instructors and policy makers to enhance the implementation and evaluation of e-learning platforms, proving the sensemaking ability of our system. We have also seen that the ability to embed meta-information to log data represents an important enhancement compared with existing web analytics tools.

In both cases the project leaders expressed that a visualisation tool could enhance their assessment and understanding of the data. Taking into account that the goal of applying web mining techniques to VLEs is to improve learning, which is not a tangible measure, contrary to what happens in e-commerce where profit is pursued, interactive tools such as WET become especially important as they provide an explorative environment to aid the generation of new hypotheses. A clear example of this fact is that, while in e-commerce there usually are one or many identifiable conversion pages, this is not feasible in learning platforms, as the real conversion is the knowledge acquired by the student, which can only be inferred from the sequence of actions performed by the student.

Results obtained in these use cases have been reported in the documentation of a European project (MEQ09) and in a PhD thesis (Vig10). In such thesis, the author who actively participated in the integration of WET to the data from the e-STUB project stated that "the representations offered with WET provide a simple and powerful way of representing visitation patterns in a meaningful way", which can be considered real evidence of the usefulness of our approaches.

Work from this chapter has been published in (PCVQ10).

Chapter 6

Exploring Asynchronous Online Discussions

In this chapter we present how we adapted WET to assist in the exploration of the often intricate structure of online discussions. Besides representing a new approach for readers and participants of online forums, the interactive capabilities of the system make it appealing for social researchers interested in understanding the phenomena and intrinsic structure of online conversations. We used discussions extracted from Slashdot.org, a widely known discussion panel, to evaluate the usefulness and adequacy of the capabilities of our system.

6.1 Research Problem

The number of large-scale discussions existing in the Web is growing every day, mainly because many web spaces incorporate discussion components in their interfaces. Such integrated interfaces have changed the way people contribute to public debates, creating new kind of conversation spaces identified as Flash Forums (DWM04). The main characteristics of Flash Forums are their large size, a tight focus overall with overlapping topics between threads, and a short timeframe for the conversation, as they usually arise from fresh news that requires rapid comments to maintain the freshness of the topic.

Discussion components in Flash Forums have to handle very large conversations that usually tend to grow proportionally with traffic, reaching hundreds or even thousands of comments per thread (NC10). This fact is compromising the usability and effectiveness of most of the current web interfaces (vBulletin[1], phpBB[2], SlashCode[3]), as they use a linear, usually nested and multipage approach. In addition, users from Flash Forums have to scan the most important comments in a short time period due to the time constraints imposed by the forums.

The main motivation of our research is to demonstrate the inefficiency of current conversation interfaces and to apply and test our highly interactive system for exploring Flash Forums. We expect that WET will increase the awareness of the conversation structure and hence, improve the detection of relevant comments, enhancing the readers experience.

6.2 Related Work

The visual representation of conversations is part of the discipline of "Social Visualisation". This term was introduced in (DKV99) as the visualisation of social information for social purposes. According to that definition, Social Visualisation may be understood as the representation of both, the structure of the participants of a social network and their interactions within the community.

Such interactions may be found in asynchronous (blogs, wikis, newsgroups, mailing lists) or synchronous form (chats). While collecting information from chats is not an easy task because those systems are usually proprietary and restricted, asynchronous online discussions leave a public trail that contains the contributions of the participants.

The analysis of such conversations may enable the discovery of hot topics and appealing subjects to boost up the community. Furthermore, researchers may benefit from that information to study the behavioural patterns behind such conversations and to interpret the social aspects of online debates.

Asynchronous online discussions can be categorised according to their structure into single-threaded and multi-threaded conversations. Single-threaded

[1] http://www.phpbb.com/
[2] http://www.vbulletin.com/
[3] http://www.slashcode.com/

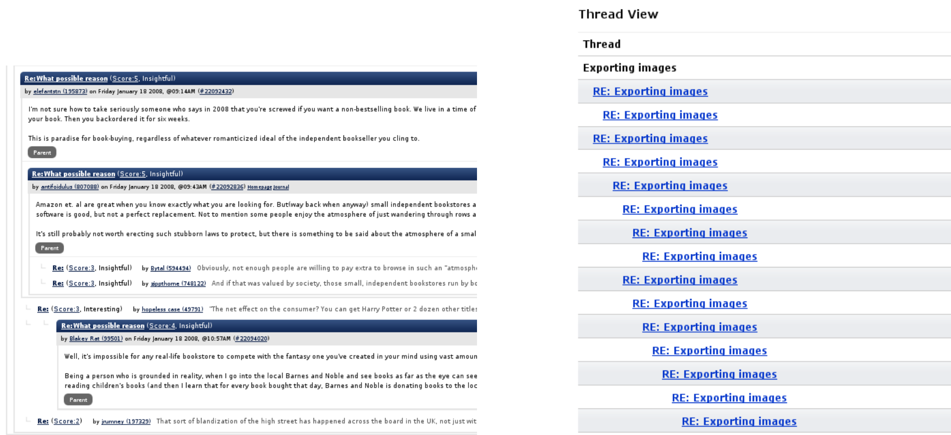
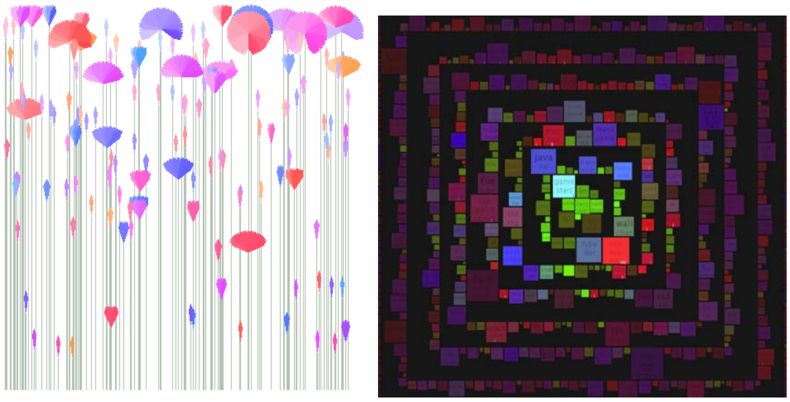


Figure 6.1: Web interfaces of conversations in Slashdot (left) and SourceForge (right).

discussions are usually embedded in digital newspapers or in blogs, and provide just one thread of comments. This characteristic makes the conversation difficult to be followed by a human reader because there is no physical structure that facilitates the discovery of comment replies.

On the other hand, multi-threaded conversations usually provide indentation or adequate titles that denote when a comment arises as a reply of another text. Two examples can be found in Figure 6.1, where the left subfigure shows a screenshot of Slashdot.org and the right one an example of a SourceForge.net discussion panel. In these examples the reader may observe the existence of a clear structure, which makes it easier to follow the different threads that were originated from the primary post. However, large debates originate countless pages with comments cumbersome to be read and navigated, complicating the process of understanding the whole conversation.

Due to the large amount of information existing in online discussions, Information Visualisation techniques have been widely used to develop visual interfaces that ease the exploration and understanding of this kind of data. An example of Social Visualisation was proposed in (DKV99), where authors introduced Chat Circles, an interface for depicting synchronous conversations that enables the discovery of the role of chat participants. The authors also presented Loom which is a system for creating visualisations of the participants and interactions in a threaded Usenet group, where posts



(a) People Garden (XD99) presents every author in the community as a flower whose petals are ordered and coloured according to post attributes.

(b) Authorlines (VS04) shows a time based plot of the overall number of contributions to the community.

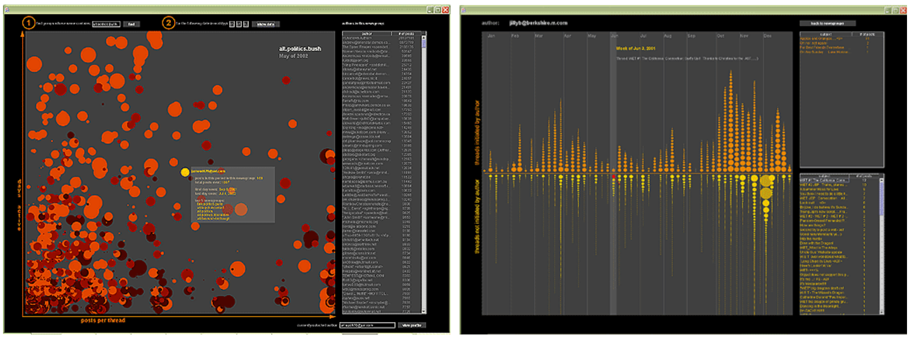
Figure 6.2: Social visualisations presented by Viegas and Smith (VS04).

and comments are represented as connected dots in the space. Moreover, they also classify posts according to their "mood", generating coloured visual patterns.

Conversation Map (Sac00a; Sac00b) is another system designed to enable scientific and non-scientific users to visually analyse online conversations. It provides three visualisations based upon the underlying social network of a thread, its semantic relations based on discussion themes, and a display area where individual threads are shown.

While previously presented tools give an overview of the discussion activity within a community, another trend has been the representation of author activity. In that sense, PeopleGarden (XD99) represents an original approach for depicting the behaviour of online communities. This system was based on the representation of every author in the community as a flower whose petals are ordered and coloured according to post attributes such as number of replies or posting time (Figure 6.2(a)). Although such an organic metaphor provides an interesting overview of a community, it may fail when tackling very large communities with thousands of users.

Focusing on author activity rather than on discussion structure, (VS04)



(a) Newsgroup Crowds (VS04) shows a scatter plot that enables participants activity in a newsgroup.

(b) Authorlines (VS04) shows a time based plot of the overall number of contributions to the community.

Figure 6.3: Social visualisations presented by Viegas and Smith (VS04).

introduces two visualisations to support social awareness in online spaces called Newsgroup Crowds and Authorlines. These visualisations emphasise the social aspects behind the contributions of users to the community rather than the patterns that can be extracted from the discussion structures themselves. Specifically, Newsgroup Crowds (Figure 6.3(a)) is a scatter plot that enables the understanding of the activity of participants in a given newsgroup during a period of time. The plot visualises an author's activity by depicting his/her number of active days and his/her average of posts per thread.

Likewise, Authorlines (Figure 6.3(b)) represent a visualisation of the activity of a single author, showing the number of contributions to the community during one year through vertical bars made up of sized circles that represent the controversy (number of replies) generated per every author's comment.

Contrary to previous approaches that focus on users activity, Difvis from Kim and Johnson (KJ06) focuses on displaying threads in a forum to provide readers with an overview of the available debates. Each thread in the forum is mapped to a square whose dimension is set according to its number of comments. The colour of the Squares is used to show popularity, activity, and the temporal aspect of the threads, in which the intensity delivers the significance of each characteristic.

The above tools provide social visualisations by means of depicting authors' behaviour as well as the whole cloud of contributions in online discussions.

However, one might miss important aspects of human behaviour when only analysing the general picture of online discussions. Surprisingly, little work has been presented regarding the representation of large conversations to support its social analysis at the same time as supporting users' actions, facilitating the navigation and easy scanning to enhance reading and contributing. Nevertheless, Netscan, presented by Smith and Fiore (SF01), constitutes a first approach of social visualisation that takes into account the visualisation of the conversation structure. In that sense, Netscan provides, among other things, a visualisation with a classic tree used to show the structure of a debate (Figure 6.4).

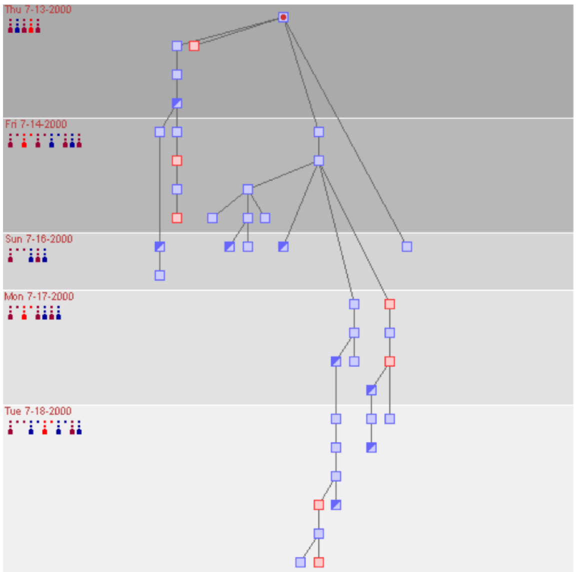


Figure 6.4: Conversation thread visualisation from Netscan. Nodes depth corresponds to posting time, while background grey bands reflect calendar days. Red nodes identify the author of the main post, while half-shaded-box glyphs identify top posters in the thread. Humaniform glyphs at the left end of each day and grey band are also used to indicate how many people posted to the thread on that day.

Another example is Forum Reader, introduced by Dave et al. (DWM04), which enables the navigation of Flash Forums, supporting the scanning of interesting comments. It is made up of five components, two of them being visualisations of a conversation and the rest being navigation and reading panels. Regarding the visualisations, the system provides two threaded visualisations. The first one, called Tree View, is a tree widget that uses an indented tree metaphor (like the classic file browser in most operative systems), which enables the navigation between and within threads. This visualisation allows the user to browse among all the different conversations of the target forum. The second visualisation, called Thumbnail View is intended to counter the disorientation users often experience by providing indented rectangles representing comments whose height corresponds to the message length. Although authors approximation is interesting, and focuses on providing a user centred design, Thumbnail View may still be cumbersome to navigate with very large debates.

Narayan and Chesire presented TLDR in (NC10), a new interface for exploring and navigating large-scale discussions focused on what they claim are three main user goals in large scale discussions: identification, navigation, and filtering of interesting content. The main novelty of their system is that it integrates with the Reddit forum, and provides a visualisation of a whole conversation using an icicle plot, while providing single visualisations of the ongoing active threads in the focus debate (Figure 6.5).

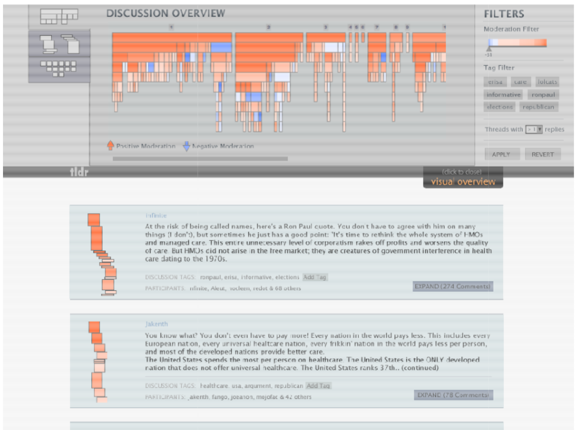


Figure 6.5: Screenshot of the Reddit forum with the TLDR system. The 'discussion overview' is shown at the top of the browser while single threads are presented with a widget that depicts activity.

Moreover, Gómez et al (GKL08) studied the social network emerging from the user comment activity in Slashdot.org and proposed a radial tree visualisation as a visual metaphor for browsing and examining the contents of highly discussed posts. As we will describe in following sections, we took advantage of the customisable capabilities of WET to provide the visualisation that the authors mentioned that "may be used to describe statistically how information is structured in a thread".

Hence, our goals are both, to enable researchers to analyse a single conversation, and to offer a visual interface to support discussion participants and readers to explore its contents.

In the following sections we will detail the features of our prototype, which allows the user to investigate the smallest details of conversation threads while at the same time improving the standard representation of forum threads as a whole.

6.3 Visualising Online Conversations

In (WGFS07), Welser et al. showed how visualisations assisted researchers in the analysis of social aspects of online discussions. However, as we have already seen, most of the existing visual systems address the problem of visualising all the posts and comments within a community, discouraging the analysis and navigation through single multi-threaded conversations. Such discussions may have a large amount of user contributions and represent a good source for analysing and understanding social behaviour by itself.

Typical web interfaces of newsgroups and forums use pagination and nesting as a way to enable the navigation through multi-threaded conversations as can be seen in the screenshots from Slashdot.org and SourceForge.net discussion components of Figure 6.1. Nevertheless, more complex approaches are also used where non-high rated comments are collapsed by default, leaving more screen space to valuable contributions. However, pagination and indentation are still inefficient when representing highly discussed posts with several hundred, or even thousands of comments.

We take advantage of the Data Processing and Mining system of WET to crawl conversation target forums, and generate a graphmls file with specific data attributes that can be explored with the visualisation system.

In this case, and following the visual approach proposed in (GKL08), the radial tree metaphor shows the conversations by representing the posts as nodes, and edges to the replying relation between them. Hence, we use the main post as the root of a tree and its direct replies are considered part of the first level. Likewise, further replies are located in deeper levels. The provided "discussion map" follows the Information Seeking Mantra (Shn96),

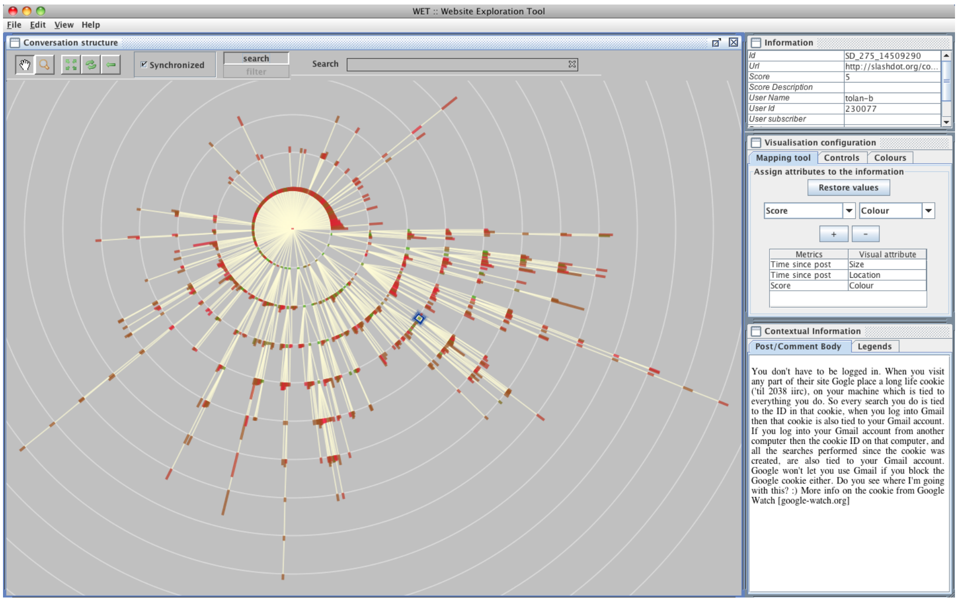


Figure 6.6: Screenshot of the user interface of WET adapted for visualising large conversations. The main frame shows the Conversion Map, while information and control panels are in the right side. Bottom right panel shows the text of a selected comment (highlighted in blue in the visualisation).

providing an overview of the discussion while enabling the extraction of details on demand: by clicking on any node of the tree, its corresponding message is shown in a message board adapted in a new tab located in the legends panel (see Section 3.3.2 for a complete description of the different areas of WET user interface). Thus, the main discussion post is placed in the centre of the tree, while its comments surround it in a concentric and nested manner, allowing the discovery of hot topics, or user-to-user debates, which can be easily identified as outliers in the hierarchy.

Figure 6.6 represents a screenshot of the user interface of WET adapted to the visualisation of conversations. A radial tree represents the discussion map, coloured according to the score of each comment and sized according to their time difference with the main post to show the evolution of the conversation.

This version of WET represents a slight modification of the user interface presented in 3.3.2, with the only addition of a panel that shows the text corresponding to the selected comment. Therefore, this prototype benefits from all the features available in the system, such as modifying the root of the hierarchy, obtaining tooltips with detailed information of the nodes metadata expressed in the configuration file, and the ability to map calculated metrics into visual attributes. For instance, as explained in Section 3.3.2, the contextual menu enables the user to open the web page of the selected comment in the visualisation.

Although the radial tree helps in building a mental map on how a discussion is structured, it may also be of interest for sociologists or web researchers to focus their attention on a specific debate (which corresponds to a sub-tree in our hierarchy). This can be done by taking advantage of the built-in capabilities of WET, which allow the user to drag a node to the centre of the tree. This is similar to what we presented in the previous chapter, but in this case, this interaction enables the user to filter the conversation by only representing one subthread, hiding the rest of the tree, and helping the user to focus on a sub-thread of the discussion. Furthermore, the system has a built-in search engine that enables the highlighting of posts relevant to a query defined by the user, as can be seen in Figure 6.7. Contrary to current built-in search engines in forums that show an ordered list of relevant comments, our system locates and highlights them in the discussion map, allowing for the discovery of how many comments they have provoked.

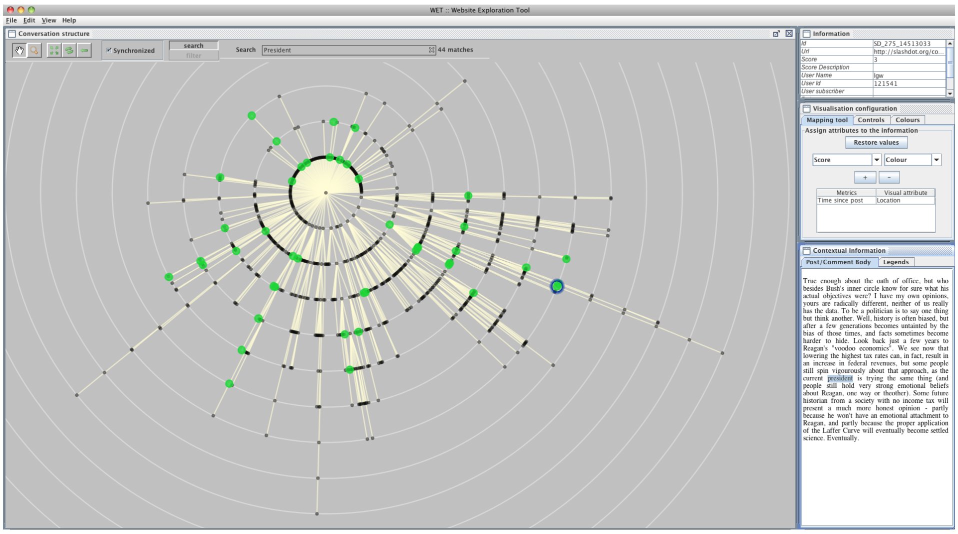


Figure 6.7: The search engine helps to locate comments with a specific text. In the image, green nodes represent comments that contain the word "President".

The highlighting system also offers several modes aimed at focusing users' interests and assisting in the discovery of information. For instance, it provides the highlighting of the path from a desired node to the root to help identify a specific thread of the discussion. Another feature of this system is to illuminate the comments written by the same author, allowing the user to easily discover whether an author has been active in a conversation or not, or to identify all the contributions of a specific participant.

Our tool supports in-depth analysis by allowing the usage of the Mapping Tool which enables users to select any metric, and encode it using any of the visual attributes available. Such metrics may be defined in a configuration file, and refer to attributes existing in the GraphML file.

6.4 Use Case: Visualising Slashdot Discussions

We integrated conversations from Slashdot.org into our tool. Slashdhot.org is a representative Flash Forum that was created at the end of 1997 and has since then metamorphosed into a website that hosts a large interactive community capable of influencing public perceptions and awareness on the topics addressed. The site's interaction consists of short-story posts that often carry fresh news and links to sources of information with more details. These posts incite many readers to comment on them and provoke discussions that may go on for hours or even days. The comments can be nested when users directly reply to a comment. This way the discussions can typically reach a depth of 7 but occasionally even depths of 17 have been observed. The structure of this discussion trees has been analysed in detail in (GKL08).

Although Slashdot allows users to express their opinion freely, moderation and meta-moderation mechanisms are employed to judge comments and enable readers to filter them by quality. The moderation system was analysed in (LR04a), and gives an integer between -1 and 5 to every comment, being 5 the highest score.

This use case is based on news posts and the corresponding comments collected from Slashdot by a web-crawler in the form of raw HTML-pages in September 2006. The collected posts were published between August, 2005 and August, 2006 on Slashdot. More details on the data and its retrieval process can be found in (KGM$^+$08). This raw data has been transformed into GraphML files.

We use these files as input for our visualisation tool, which then is able to represent general features of online discussions as well as some Slashdot specific metrics like comment score.

From the data available, our web mining system calculates a set of metrics chosen according to their existence in common forums, as well as their importance regarding the information they provide of a single comment:

Score: an integer value between -1 and 5, indicating the comment's score obtained from Slashdot's moderation system. Lampe and Resnick (LR04b) analysed the importance of the score in the process of locating relevant comments. The mapping of this metric into a suitable representation might improve the understanding of the whole hierarchy, allowing an easy identification of top rated comments in the discussion. The existence of concentrated values in the same branch of the radial tree may provide an interesting visual cue that the user can use to explore the thread.

Time since post: represents the amount of minutes since the initial post of the discussion was published. It enables the creation of representations that provide a general feeling on how the discussion has evolved in time and allows the user to focus his/her attention on a specific group of early or late posts. Moreover, this metric is crucial, and should be taken into account when providing a real time visualisation of a Flash Forum, as stated previously.

Time since parent: similar to the Time since post, but the elapsed time is relative to the parent of the comment. This metric also allows the exploration of the evolution of the discussion in time but, in this case, allows the user to concentrate on the children of a specific node to visually discover the most recent ones.

Maximum Children's Depth and Total number of children: These metrics reinforce the notion of the controversy level of each comment by calculating the maximum depth and the total number of originated comments. As described in (GKL08), controversy can be measured according to number of comments existing at a certain depth.

Comment length: long comments may intrinsically incorporate more ideas, and hence, may be another interesting and easy way to collect metric of a post. However, conciseness is an important aspect of relevant comments in Fast Forums due to the fast pace of these kinds of conversations.

Figure 6.8 shows a subthread dynamically filtered from a larger conversation. Moreover, the picture shows how the different visual attributes may affect the conversation map. Moreover, the image reveals two different debates created by two users that constantly reply to any comment to their posts.

Due to the visual representations based on the user highlighting and the visual representation of the h-index, researchers have been inspired to focus on the role of dialog chains in the discussions, which are far more common than expected before visualising the discussions with our tool. These chains are produced when two users start to reply to each other's messages several times, often reaching a depth far greater than the h-index of the discussion.

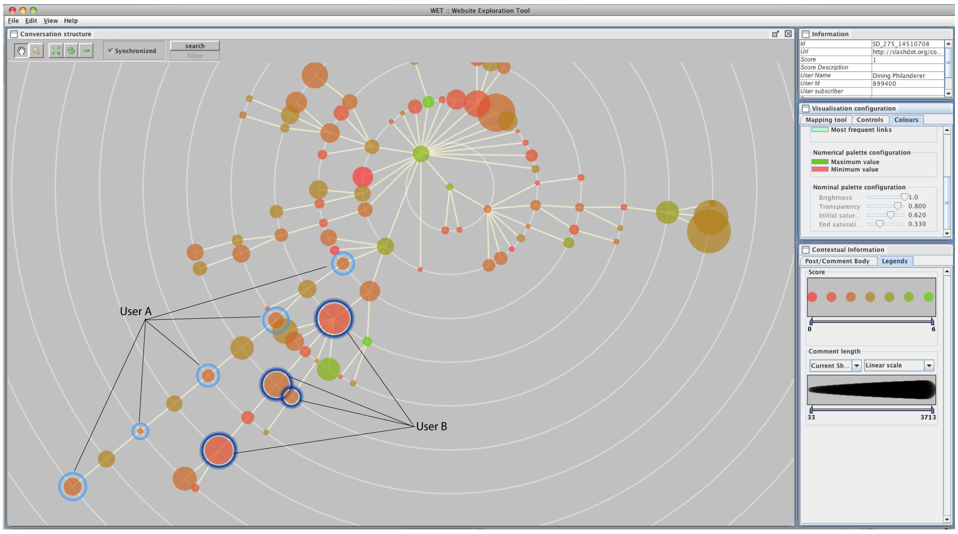


Figure 6.8: A sub-thread from a conversation. Score is represented with colour while items size denotes comment length. The built-in highlighted system of WET enables the user to discover the behaviour of two authors, who continuously respond to new replies to their posts, generating debates. The colour coding suggest that their contributions are not well rated by the community.

6.5 Controlled Experiment

Narayan and Cheshire (NC10) analysed in detail the behaviour of users of the Reddit forum[4], and discovered that the most important factors when determining which content to read are users' own perception of content and the relative position of the comments. In their analysis, the authors identified three main characteristics of a comment that made it appealing or interesting to the readers: comment length, number of replies and position in the discussion. Surprisingly, the author of the comment was not ranked as important.

We conducted a formative usability study aimed at assessing the abstraction level introduced in Section2.6 of our tool, comparing users' performance with a set of tasks related to the factors identified above with our interface and the Slashdot.org web interface. Slashdot's interface can be considered a good baseline as it integrates the most advanced features for browsing and reading conversations. It incorporates collaborative and dynamic filtering for rating and hiding non-relevant comments, a threaded interface with links that facilitate the navigation through the parent comments to ease the identification of subthreads in the conversation, and the ability to collapse comments belonging to the same subthread.

We present below the main hypothesis of our experiment:

- Current web-based conversation interfaces do not provide enough context to the users, so they cannot understand their depth and breadth, preventing the discovery of the controversy of the different comments understood as the number of replies and subthreads that arise from one comment (H1).
- Users will find relevant content more easily with our interactive visualisation tool than with the Slashdot web interface (H2).
- Current threaded approaches fail to show large conversations, which means that the size of the thread should clearly affect the performance of the users while detecting relevant or desired content (H3).

From this evaluation we also expected to collect qualitative evidence that would help us to better understand how users interact with our tool as well as to discover usability problems.

[4]http://www.reddit.com/

6.5.1 Study Design

Task	Description
T1	Locate and count the number of comments with the best score that are a direct reply of the main post
T2	Find again the comments with the best score that are a direct reply of the main post. Which of these comments originated a discussion (subthread) with the highest number of best rated comments?
T3	Find the longest comment in the subthread that the evaluator will show you
T4	Find out the reply of the main post that originated the biggest subthread (biggest number of replies including the replies to the replies, etc.)
T5	Find out the sequence of comments that ended-up with the comment that the evaluator will show you, and read the very first word of each one
T6	Find a debate on the last subthread. That is to say, find a sequence of comments where only two users that reply to each other interact (A – B – A – B)

Table 6.1: Tasks used in the experiment with their type.

We recruited 13 users for the evaluation (12 males and 1 female), all of them being experienced computer users familiar with tasks related to reading conversation from forums. Almost half of them also had experience with opinion mining methods, which made them especially skilful with forum interfaces. Also, half of the users already knew the Slashdot.org website, although only one was a regular reader. However, all of them expressed they had vast experience dealing with forum-based interfaces.

We used a repeated measures within subject factorial design with two datasets, the two interfaces and the tasks presented in Table 6.1. All of them were intended to provide evidence regarding H2 and were designed according to users preferred comment features highlighted in (NC10). As H1 can be understood as a generalisation of H2, the overall conclusions from H2 combined with observations and users feedback helped to draw conclusions about H1. Finally, in terms of the experiment's datasets, we selected two real conversations extracted from Slashdot according to their sizes and structure. The first one was an average-sized thread with more than two hundred comments, while the other had more than a thousand comments. This type of conversation occurs about once a month in Slashdot. Although we did only use two different-sized conversations in our study, their big difference in size helped in extracting results to confirm or reject H3.

Each user performed every task four times, one per dataset and interface so we could measure time differences. Tasks and environment settings (dataset and interface) were counterbalanced using Latin Square to avoid the learning effect. Moreover we modified the default settings of the Slashdot web interface for a fairer comparison. By default, Slashdot shows the very first 50 comments of a conversation, automatically folding the ones less interesting according to a score threshold set to 1 (from a scale of -1 to 5), while the rest are displayed on demand. As WET shows all the comments at the same time, the conversation threads of Slashdot were fully unfolded at the beginning of every task.

Users also had 15 minutes training before using each interface for the first time. We gave separate training per interface as we observed in a pilot test that users rapidly forgot and confused the functionalities of each interface. The whole experiment lasted and average of 70 minutes with a maximum of 5 minutes to complete each task.

We recorded users' screens using screencast software that also registered users' voice. Although users were not allowed to ask questions during the evaluation, voice recording was useful for reviewing comments during the post-test questionnaire.

6.5.2 Results and Analysis

We used a repeated measures analysis of variance to determine differences between time and errors per task. Errors were quantified binary with a score of 0 if the task was accomplished successfully within 5 minutes, and 1 otherwise.

Tables 6.2, 6.3, 6.3 and 6.5 show times and errors (red coloured cells) of all users and tasks. Purple cells represent abandoned tasks mainly due to their complexity, or because the maximum time of 5 minutes was exceeded. At a glance, it can be observed that the number of errors with WET is smaller than with the Slashdot web interface, as can also be appreciated in Figure 6.9.

	T1	T2	T3	T4	T5	T6
U1	0:02:53	0:01:18	0:01:01	0:03:07	0:00:54	0:03:11
U2	0:00:21	0:01:10	0:00:34	0:01:03	0:00:30	0:00:47
U3	0:00:21	0:00:09	0:00:20	0:02:53	0:00:32	0:01:16
U4	0:00:46	0:00:18	0:00:21	0:02:50	0:00:31	0:04:52
U5	0:00:36	0:00:28	0:00:15	0:01:50	0:00:54	0:00:39
U6	0:00:10	0:00:16	0:00:14	0:03:07	0:00:38	0:01:08
U7	0:00:13	0:00:28	0:00:18	0:01:23	0:00:31	0:02:06
U8	0:00:35	0:00:32	0:00:27	0:01:06	0:00:25	0:00:32
U9	0:00:22	0:00:50	0:00:12	0:02:45	0:00:54	0:00:36
U10	0:00:10	0:00:16	0:00:13	0:01:31	0:00:55	0:00:32
U11	0:00:14	0:00:15	0:00:19	0:01:13	0:00:56	0:00:26
U12	0:00:25	0:00:21	0:00:14	0:02:25	0:00:38	0:00:49
U13	0:00:21	0:00:42	0:00:24	0:02:13	0:00:27	0:00:54

Table 6.2: User's time per task with the small dataset and the Slashdot web interface. Red cells indicate that there was an error during the task

	T1	T2	T3	T4	T5	T6
U1	0:01:26	0:01:21	0:01:00		0:01:14	0:00:55
U2	0:01:29	0:01:16	0:02:17	0:04:29	0:02:06	0:04:09
U3	0:00:37	0:01:06	0:01:42	0:05:00	0:02:21	0:01:39
U4	0:01:08	0:01:38	0:01:24	0:04:32	0:01:06	0:01:56
U5	0:00:46	0:00:41	0:00:40	0:02:00	0:00:37	0:01:02
U6	0:00:45	0:00:49	0:02:19	0:01:32	0:01:53	0:02:14
U7	0:01:07	0:03:05	0:01:32		0:00:45	
U8	0:00:49	0:00:33	0:00:36	0:01:40	0:01:13	0:03:12
U9	0:00:33	0:01:25	0:01:34	0:04:30	0:00:47	0:01:47
U10	0:00:58	0:00:48	0:00:57	0:04:06	0:01:22	0:02:37
U11	0:00:51	0:00:41	0:00:50	0:05:00	0:00:42	0:01:41
U12	0:01:16	0:01:18	0:02:25	0:04:30	0:01:17	0:03:48
U13	0:00:42	0:01:06	0:01:27	0:05:00	0:00:34	0:01:10

Table 6.3: User's time per task with the big dataset and the Slashdot web interface. Red cells indicate that there was an error during the task. Users had major problems to accomplish T4, as the Slashdot interface is not well suited for comparing the size in the subthreads of a conversation.

First results clearly showed that T4 was the hardest task to accomplish with the Slashdot web interface, especially with the large dataset, as only one user who guessed the best solution at the end of the 5 minutes was able to complete it. Regarding this task, users stated that it was "virtually impossible with such big conversations" and that they would never investigate the largest subthread in a conversation. However, they agreed that it might be an interesting feature to be aware of as it provides a notion of comment controversy. On the contrary, significant differences were found between the two interfaces with the small dataset ($p = .004$), WET being the one that provided the fastest results. It is also important to note that

	T1	T2	T3	T4	T5	T6
U1	0:00:12	0:00:30	0:00:20	0:01:03	0:01:00	0:00:21
U2	0:00:13	0:00:30	0:00:10	0:01:47	0:00:41	0:01:02
U3	0:00:10	0:00:19	0:00:13	0:00:29	0:00:45	0:00:31
U4	0:00:14	0:00:21	0:00:14	0:00:10	0:00:26	0:00:12
U5	0:00:20	0:00:13	0:00:13	0:00:16	0:00:19	0:00:38
U6	0:00:13	0:00:18	0:00:19	0:00:13	0:00:52	0:00:41
U7	0:00:44	0:00:46	0:00:18	0:00:42	0:00:37	0:02:46
U8	0:00:17	0:00:18	0:00:14	0:01:06	0:00:33	0:02:10
U9	0:00:17	0:00:22	0:00:11	0:00:21	0:01:05	0:00:39
U10	0:00:12	0:00:24	0:00:12	0:00:11	0:00:30	0:00:19
U11	0:00:23	0:00:33	0:00:47	0:00:22	0:00:57	0:01:11
U12	0:00:21	0:00:20	0:00:19	0:00:26	0:00:34	0:00:29
U13	0:00:12	0:00:14	0:00:15	0:00:08	0:00:24	0:00:16

Table 6.4: User's time per task with the small dataset and WET. Red cells indicate that there was an error during the task.

	T1	T2	T3	T4	T5	T6
U1	0:00:27	0:00:37	0:00:22	0:00:50	0:01:20	0:00:50
U2	0:00:40	0:00:42	0:01:09	0:02:36	0:00:58	0:02:39
U3	0:00:26	0:00:56	0:00:14	0:00:40	0:01:02	0:00:40
U4	0:00:31	0:00:49	0:00:28	0:00:41	0:00:59	0:00:23
U5	0:00:27	0:00:37	0:00:22	0:00:40	0:00:44	0:01:57
U6	0:02:04	0:02:08	0:00:18	0:00:18	0:01:39	0:00:50
U7	0:00:27	0:04:42	0:00:32	0:00:09	0:01:25	0:01:05
U8	0:00:26	0:01:00	0:00:19	0:00:43	0:00:55	0:01:43
U9	0:01:05	0:01:10	0:00:31	0:00:21	0:01:05	0:01:17
U10	0:00:18	0:00:37	0:00:16	0:00:13	0:01:10	0:00:23
U11	0:00:44	0:01:45	0:00:18	0:01:04	0:01:04	0:02:02
U12	0:00:21	0:00:55	0:00:29	0:01:03	0:00:12	0:01:24
U13	0:01:17	0:01:11	0:00:16	0:00:12	0:00:36	0:00:33

Table 6.5: User's time per task with the big dataset and WET. Red cells indicate that there was an error during the task.

there were no significant differences between the two datasets with WET, which may indicate that the time needed to accomplish such tasks does not depend on the size of the thread. In this case, we observed that 84% of the users used the location attribute to place the desired comment on the right-hand side of the visualisation, as can be seen in Figure 6.10. This result partially supports H1 although it must taken into account that our system incorporates a metric that indicates the total number of comments existing under every node (Section 6.4). Nevertheless, highly unbalanced threads might be noticed very easily with our Conversation Map, as was also pointed out by a user.

In terms of the rest of the tasks, significant differences were found in the overall completion time (obtained by summing up all the times per user,

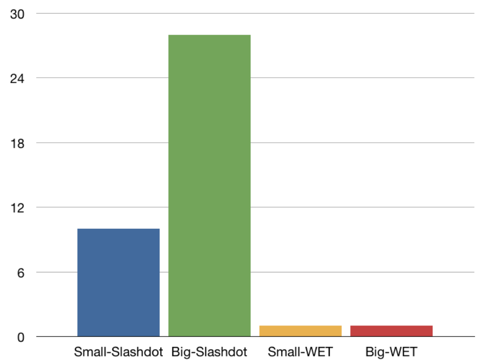


Figure 6.9: Number of errors with the different experiment settings.

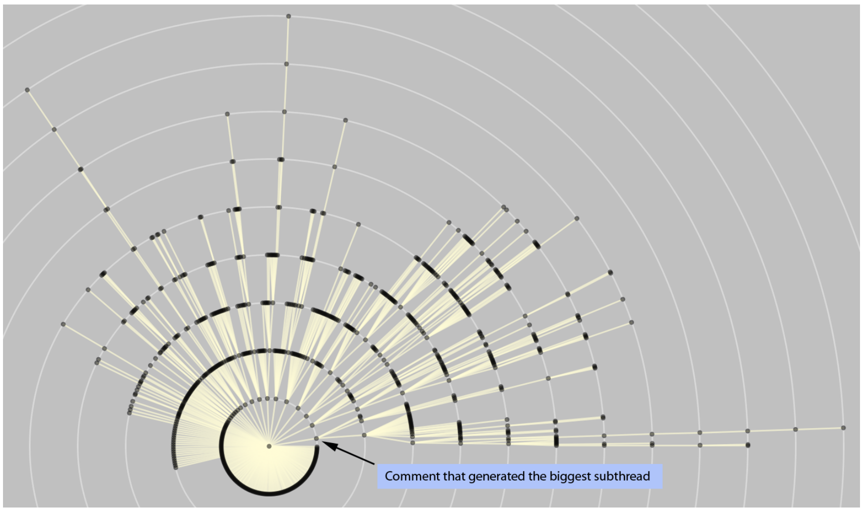


Figure 6.10: Location helps to easily identify the comment with the biggest subthread.

dataset and interface except T4) and errors with both datasets, (with $p < 0.05$). No matter the dataset, the average time and number of errors was significantly smaller with WET than with the Slashdot web interface. However, no major significant differences were found in terms of single tasks time completion except T3, where the existence of representative metrics not available in Slashdot do not allow the user to extract conclusions regarding the usefulness of the visualisation techniques applied to our tool. Nevertheless, it is interesting to note in Table 6.3 that users made a considerable amount of errors in tasks T1 and T5. This result may support H1, as was observed in video recordings; the overwhelming amount of comments reduced the efficiency of using indentation for distinguishing between comments from different levels. Furthermore, Figure 6.9 shows the significant difference in error rates between the two systems.

The non existence of significant differences can be explained by the lack of training that users had with our tool, as can be seen in the recordings where some of them had doubts when trying to use specific functionalities of the system, such as the different highlighting methods, introduced in the training session. On the contrary, the Slashdot web interface follows classic approaches more familiar to the users that means they have fewer doubts when performing the tasks with such a system. Such a problem was identified in (Car08) and is a recurrent problem in the evaluation of InfoVis systems. However, the review of the recordings as well as the post evaluation questionnaire showed that users understood the rationale behind the Conversation Map without any problems with there being 12 out of 13 users who rated the tool as "Easy" or "Very Easy" to use. Nevertheless, the same number of users stated that they were "Completely Sure" that the tool might assist them in the process of discovering interesting comments or subthreads in conversations. The results of the post-questionnaire can be observed in Figure 6.11.

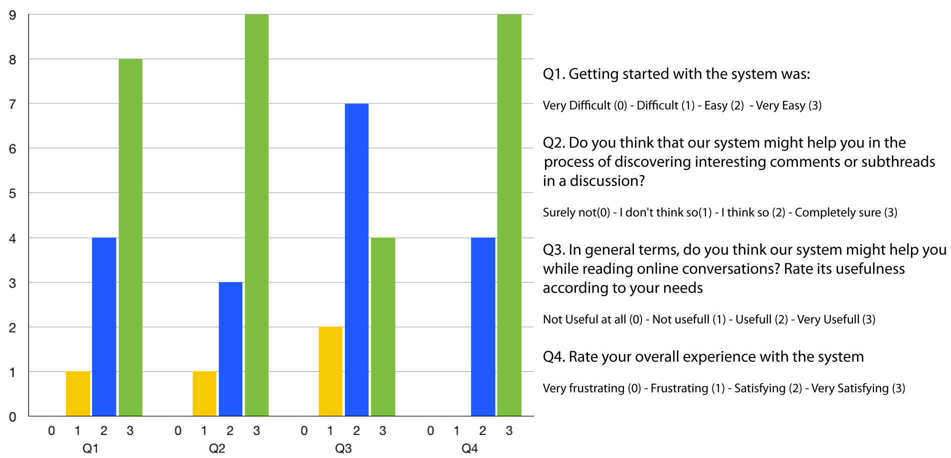


Figure 6.11: Post-test questions were ranked from 0 (low ranking) to 4 (best ranking).

Another interesting finding was that although web based forums such as

Slashdot allow comment filtering through its score, users acknowledged their interest in discovering relevant comments according to other features such as the controversy they generate. In that sense, users stated that having a visualisation of the whole conversation was very positive and might influence their reading behaviour. However, users also revealed that our tool might not be the best choice for reading the thread, as it only provides space for one comment at a time, contrary to current forums. It is then clear that users still prefer the classic threaded and linear approach for reading the conversation, while our approach may support the navigation through the conversation for finding relevant comments and the awareness of the discussion context in terms of structure.

6.6 Conclusions

The existence of large conversations is becoming more prevalent as many pages incorporate conversation components. While their hierarchical structure may provide a clear organisation that might enable participants and researchers to understand how a discussion is structured, the lack of an overview and interactive methods for browsing and navigating them are compromising the usability and effectiveness of classical web-based discussion interfaces.

Following the call of Gómez et al. in (GKL08), we took advantage of the customisable nature of WET to visualise online discussions with the radial tree metaphor, allowing the easy discovery of their threaded structure. Our interface is aimed at enabling readers, conversation participants and researchers to navigate through the often intricate discussion structure, allowing the customisation of the visual items according to a set of comment features. The main interactive map of the conversation provided by our system may help in the understanding the structure as well as the level of controversy existing in a thread.

Our system contributes to the discipline of Social Visualisation by complementing existing tools and offering visual approaches to analyse a discussion in detail. We have applied it to conversations from Slashdot.org and evaluated it through a controlled experiment with a set of users that accomplished a set of tasks related to the discovery of interesting comments according to several criteria such as the number of generated subthreads, the users rating or comment length. Results show that our system outperformed the Slashdot.org web interface in the overall time spent per task, while a few tasks were completely unmanageable with the traditional linear interface. Users' qualitative feedback also showed the inefficiency of current discussion interfaces, while suggesting that our conversion map may assist the navigation, awareness and finding of relevant comments in large conversations. However, the traditional approach seems to be best for reading content, which suggested the need to integrate both approaches: the visual conversation map for exploring and discovering interesting comments, and the classic approach for reading them. Nevertheless, feedback from users suggest that WET is too complex to merely assist in the exploration of conversation threads. From these comments, we infer that a more specific customisation of the tool might improve its usability for non-analytic purposes. That is to say, rather than providing the mapping tool and the rest of the interaction panels, users suggested that the existence of the conversation map colour coded according to the score of the comments would be enough, and would simplify the usage of the tool.

Our prototype may be easily adapted to permit the representation of a conversation in real time, being an interface for enhancing the engagement of users to the discussion. Evaluation results also suggest that the tool is especially useful for researchers working in the analysis of social interactions in online forums, as visual inspection of the forum conversations is very important when identifying new phenomena or finding interesting relations between structure and variables. This task is often cumbersome using standard visualisation tools or just numerical analysis. Once a certain interesting characteristic with our tool in specific discussions has been identified, an exhaustive analysis can be performed afterwards to demonstrate the generality of the findings.

Future research directions will focus on the integration of our conversation map in discussion components. Our main hypothesis is that the context provided by the map of the conversation as well as its exploration capabilities will dramatically change the way users read and participates in flash forums.

Furthermore, refining the approach presented in (SF01) where the entire author relations network is visualised, we will take advantage of the graph visualisations available in WET to include the representation of ego-networks and other author interactions in our platform.

Work from this chapter has been published in (PCK09).

Chapter **7**

Final Discussion

In this dissertation we have addressed the problem of analysing and understanding Web spaces, in an effort to bridge the disciplines of Web Mining and InfoVis/VA. To tackle this problem, we developed a visual tool called WET, which provides an environment to visualise and explore graph-based data. The customisable nature of our system has allowed us to adapt it in order to support the exploration of different Web spaces, such as websites, virtual learning environments, and large conversations existing in online forums.

In each of these Web spaces we have stressed the main problems that users must face, and evaluated our tool using a variety of evaluation techniques such as controlled experiments and evaluations with real users in real scenarios to report evidences of the usefulness of some of the techniques available in our tool.

In this chapter we will present a summary of our contributions, discuss the generalisability of our results and suggest future research directions.

7.1 Contributions

InfoVis and Visual Analytics are still young fields. While early research was mainly devoted to the generation of new visual metaphors and interaction techniques to represent abstract and large datasets, these disciplines still lack evidence to help identify the best approaches to solve specific problems, as well as to find out best practices to support the investigation of data. To do so, many researchers recommended concentrating efforts on moving from research into practice (Pla04; TC05; Mun09), aimed at learning from designing, deploying and studying InfoVis/VA tools in real world scenarios.

In an effort to contribute in this regard, we have introduced an integrated architecture that enables the visualisation of Web spaces in the form of graph-based structures, and used it to tackle problems related with the understanding of different types of Web spaces. We have also provided further evaluations that have stressed the benefits and the potential of our visualisations in three different Web spaces. According to the goals presented in Section 1.2, our main contributions have been:

The development of a customisable visualisation tool. In this dissertation we have introduced the development of our customisable system (see Chapter 3), which provides a coordinated environment to explore data. The system incorporates interactions to modify the visual attributes of represented items as well as dynamic filtering mechanisms to reduce the amount of information that is being displayed.

The conversion of large graphs into meaningful subgraphs. Our Graph's Logic System describes a mechanism to reduce the complexity of large graphs. While classic techniques tackle this problem by forming data clusters, our approach is able to extract smaller subgraphs that maximise the existence of relevant data according to the needs of the analyst. Hence, the system is able to filter non-relevant data items while preserving the structure of the graph (Section 3.2).

The application of Infovis/VA techniques for assessing websites. One of the main goals in this dissertation has been the visualisation of website data (see Chapter 4). This goal was motivated by the difficulty expressed by the analysts of websites to comprehend the vast amount of usage data that they have access to. Our challenge has been to provide effective visualisations of website structure and usage in order to aid the evaluation of websites. Our system creates visual abstractions of website structure and usage that provide a context to understand web metrics that can be mapped on top of the visualisations using different visual attributes. In terms of structure, *we have proposed an extension of the classic breadth first search algorithm* that improves the number of highly visited links visible, while preserving the depth of the pages in the hierarchy according to its shortest distance in terms of clicks from a specific root page. The combination of the hierarchy obtained with this algorithm represented with a radial tree, and the ability to map usage data on top of it as well as to filter most relevant links represents novel approaches that can be used for the assessment of the usability of a website. This approximation allows the user to discover design inconsistencies, and provide a context that can help in the generation of new hypothesis about the data. This fact became especially apparent in a user study with domain experts, who stated that this approach may definitely help them to elaborate more complex hypothesis, such as in the case of analysing the broken links of a site.

In an effort to satisfy the needs of the analysts gathered in a formative evaluation, we also developed two different visualisations:

- *An interactive graph generated through the superimposition of all the paths performed by users of a site.* The generated graph revealed many new possibilities to the analysts due to its capabilities for showing a fine grained description of users' movements, enabling them to find main behavioural trends that cannot be discovered with current tools, which mainly provide static funnel charts that need to be customised a priori.
- *A new hierarchical approach that takes advantage of a maximum branching algorithm.* This approach, called usage tree, is especially interesting as it is a novel approach for representing the most used routes that started at a specific landing page. Using a maximum branching algorithm, our technique is able to generate hierarchies that represent a summary of the most navigated paths.

According to domain experts, the linking and brushing capabilities of our system applied to the comparison between the structure of a website and the usage tree is a powerful tool to compare the shortest path from a root page to a desired node with the most common route performed by the users. In our study, *we also characterised the size of the graphs generated by the superimposition of users routes starting at a specific page* performed in five well navigated websites. As a result, we saw that the topology of the webgraphs obtained turned the complexity of the maximum branching algorithm into quasi linear (Section 4.4.3). A thinking aloud protocol provided feedback from domain experts, who stated that the visualisations of WET might complement current tools in order to better assist in the investigation of usage data (Section 4.5).

With the study of our visualisations with domain experts we have learned that simple visualisations, such as the structure tree, that do not require a deep exploration of the data are preferred, mainly due to time restrictions and pressure in reacting fast to what happens in the target websites. It is interesting to note that recent research conducted in other domains has provided similar insights in this regard (Sed10). Moreover, for Infovis/VA solutions to succeed it is very important to integrate them within real data currently available in analysis tools, which currently provide APIs containing already pre-processed usage data.

Visualisation of Virtual Learning Environments. Virtual Learning Environments (VLEs) can benefit from Web analytics tools and methodologies as they are web based applications. The analysis of usage data from these kinds of platforms may dramatically improve the understanding of the behaviour of the students, as well as understanding the usefulness of the learning material. We explored the impact of our visualisations in this regard, and conducted two MILC studies with usability experts. In this process, we *characterised the main problems that instructors must face when dealing with usage data from learning environments* (Section 5.1). One of the main characteristics is that, contrary to what happens in e-commerce where profit (that can be easily measured) is pursued, instructors need more qualitative approaches to understand usage data and to formulate hypothesis about the quality of their materials and the usability of the courses.

We have seen how in both case studies our tool supported the discovery of uncovered and unexpected patterns (Section 5.4), suggesting that it may potentially provide a valuable asset for instructors and policy makers to enhance the implementation and evaluation of e-learning platforms. In order to better understand how the analysts used our tool, we developed a logging system, rarely used in most MILC studies reported to date, that helped us to keep track of the analysts' behaviour. Due to the different analysis goals of the two studies, we cannot infer that any of the provided visualisations are better than the others. However, we saw that all of them were used and revealed interesting patterns to the analysts. With these studies, we have learnt the importance of these kinds of logging systems, as they allowed us to get an understanding of how our system was used before the final interviews with the analysts. This fact simplifies the task of interviewing users, as clear questions may arise after the analysis of such logs.

Visualisations of large conversation in online forums. In this dissertation we have also introduced the usage of our tool for assisting the exploration and navigation through asynchronous online conversations. These types of debates are transforming the way we communicate, and have been spreading through websites within discussion widgets, raising new types of conversations whose main characteristics are large size, a tight focus overall with overlapping topics between threads, and a short timeframe for the conversation. The main problem in such forums is that readers and contributors have very little time to decide where to read or contribute next.

Taking advantage of the main characteristics of the WET architecture, we used the radial tree metaphor to represent these conversations in an interactive interface that enables the user to explore and navigate the different comments of a conversation. *We applied our approach to a real use case based on a popular news site, and evaluated our solution* with a controlled experiment.

Although no major statistical differences were found in terms of tasks performance, the interactive conversation map provided by WET significantly reduced the number of errors performed by the users while accomplishing a set of tasks that mainly involved the discovery of interesting comments and subthreads.

User feedback was also encouraging, suggesting the integration of our visualisations in current forums. However, we learnt that the linear and nested organisation of comments is preferred when reading conversations suggesting the development of a mixed approach containing our conversation map for locating especially interesting areas of the discussion as well as the classical linear approach to read them.

We claim that the interactive capabilities of WET could also benefit to social researchers who might want to analyse in detail the structure of large asynchronous conversations.

7.2 Evaluation Process and Generalisability

As we have already presented, we attempted to validate this PhD work and demonstrate the benefit of its use in three different types of Web spaces. We first conducted a formative evaluation with domain experts in the field of web analytics to validate the adequacy of the visualisations in a real scenario. The exhaustive analysis sessions conducted with domain experts were useful to get positive feedback of the novelties presented by our tool, as well as relevant features to be added in future versions such as the inclusion of report and annotations tools in favour of best analysis practices. Such evaluation also helped us to report major usability problems of the system that were addressed before the rest of the evaluations. Rather than validating the benefits of our tool itself, we have contrasted the advantages of our approaches, suggesting their inclusion in current web analytics solutions.

With a refinement of the system, we conducted in parallel two main evaluations: a controlled experiment to compare our system with the interface of a very well known online forum, and a long-term study with analysts of virtual learning environments. The former revealed that our system assisted in the discovery of relevant comments in the site, presenting statistically significant results in number of errors, although presenting reading problems that suggest a mixed approach. The latter has been particularly interesting as two real analysts with real needs used our tool during a long period. Although with such evaluations we cannot claim that our tool is better than any other system, our study draws interesting conclusions based upon main findings provided by the users, who rated the tool and its visualisations very positively. The comparison of the results in the long term study along with qualitative feedback obtained with the domain experts from the formative evaluations provides first clues about the impact that the visual approaches provided by WET may have when incorporated with current analysis tools.

From these results, it has to be considered that the effort of moving research into practice in order to benefit from the realism of the experiments has compromised precision and generalisability, mainly due to the difficulty in accessing a wide number of domain experts (McG95; Sed10). Nevertheless, as we have seen throughout this dissertation, encouraging feedback has been collected in favour of the usefulness of the Infovis/VA techniques applied that may serve as guidelines to further research in the same direction.

7.3 Future Work

The short term future work of this thesis must encourage the enhancing of the WET tool, by incorporating more Visual Analytics techniques to provide support for hypothesis validation and justifications as suggested in the analysis sessions. Features such as annotation and exporting tools to facilitate the revision of the obtained visualisations may also improve the usefulness of the tool.

Beyond the scope of the WET project, a lot of work remains to be done in the visualisation of the different Web spaces presented so far, as well as in the Infovis/VA field. In this section we will present what we believe are the most promising directions, and propose some new ideas to tackle them.

7.3.1 Within the Web Analytics Community

The short term future work should be related with the integration of WET with APIs of tools such as Google Analytics, which were not available at the time that this research was carried on, and may make our system more widely available. Making the tool more accessible to analysts by integrating it with their tools may help in the discovery of the impact of the different visualisations. Moreover, such integration may also facilitate the development of more long term evaluations.

Furthermore, in this dissertation we have introduced our Graph's Logic System, which is capable of extracting contextual subgraphs of large websites. Another possibility to reduce the large amount of pages to be visualised might be through clustering techniques. Although traditional clustering techniques such as the ones based on textual properties of the pages, or even based on their location in the structure of the site might be useful, an interesting new approach would be to extend them by also considering *clusters of pages generated according to their HTML structure*. That is, knowing that there are a lot of dynamic pages in the Web that are generated through scripts that use HTML templates to load information stored in a database, we can assume that related pages may share most of their HTML structure. Techniques for clustering textual documents may be applied to non-frequent HTML tags (selected according to their tf-idf weight) appearing in a website. This method might be very useful in a combination of traditional clustering approaches to discover very similar pages such as product pages in an e-commerce website.

Moreover, it can be of much interest to analyse the best coupling of WET visualisations with data reports existing in current web analytics packages. As we have observed in our evaluations, analysts want to have numbers face to face with visualisations to validate hypothesis obtained through the visual exploration of the data.

Furthermore, there are many types of data involved in the analysis of websites that we have not worked with, and should be considered in further research. For instance, the visualisation of the keywords used by users might be of extreme importance to SEOs, helping to understand organic traffic and Pay Per Click strategies as big websites may have hundreds of thousands of keywords used by the users, which represents a challenge to understand.

7.3.2 Within the Virtual Learning Environments Community

The novelty of VLEs opens up a wide range of research opportunities. From our perspective, the most important one should be the integration of a visualisation tool such as WET in the same VLE, allowing the instructor to keep track of students' behaviour during the development of the course. Such interaction may provide a priori data that may help to adequate teaching strategies during the course.

Furthermore, we have assisted analysts in the evaluation of VLEs where the students are passive actors. From a psychological point of view, it would be of much interest to use more 2.0 concepts, understanding how converting students into active actors of the VLE may influence the teaching strategies and their learning process.

7.3.3 Within the Context of Asynchronous Conversations

In this dissertation we have seen how our conversation map can support the context awareness of a conversation and the location of interesting comments. However, for this approach to succeed, further research must implement these kinds of visualisations into the same forums, and use usage data to understand readers and contributors behavioural patterns. Such patterns may help to understand how the visualisation of the conversation structure may affect the way users read and contribute to forums. In addition, it can also be interesting to see how the conversation map can affect the reader and contributors from conversation threads with single-threaded interfaces, such as the ones existing in many online newspapers, helping to understand if the existence of a clear structure in the conversation may improve its readability and attractiveness to readers.

Moreover, although some studies show that the author is not relevant when selecting new messages to read, we believe that the reutilisation of the graph visualisation in WET (applied in the sessions graph) may be convenient to depict the social network among the different authors existing in a conversation. With this visualisation integrated in a smooth way, it would be interesting to see if the relevance of the comments changes according to their author.

7.3.4 Within the Fields of Information Visualisation and Visual Analytics

The evaluation of WET with domain experts has revealed that most of them suggested using the provided visualisations as ways to communicate analysis results to the non-expert board of directors. This finding suggests a need to better understand and conceptualise the differences between using visualisations for communicating data and for exploring it.

With our work, we would also like to call for more long term evaluations performed in real scenarios, as the success stories from research conducted in real scenarios may encourage the interest of potential adopters. Along this line, it is important to further investigate methods to support such evaluations, such as the establishment of a well defined methodology to log users' actions, and the study of techniques to understand the potentially huge amounts of data generated in this regard.

7.4 Epilogue

Information Visualisation and Visual Analytics have a great potential that still needs to be exploited by regular large information sets, consumers and analysts. While research is still needed to better understand how visualisations may take better advantage of preattentive processing to facilitate the understanding of data, it is also very important to come up with taxonomies to identify the most useful visual and interaction techniques to solve specific tasks or problems.

Nevertheless, much InfoVis/VA research is being carried out through user centred design approaches that take into account user feedback in all stages of the design process. However, the novelty of many InfoVis/VA techniques is usually difficult to be accepted by early adopters. The complexity and pressure of their work usually prevents them from having enough time to understand and learn new visual systems. We believe that there is a need to train professionals to read and understand advanced graphics with successful practices. In fact, new generations that have grown up with the Internet and video games are much more used to seeing and interpreting interactive visualisations than we have ever been used to. Therefore, new generations will definitely overcome the visual illiteracy problem that we are facing nowadays.

Bibliography

Each reference indicates the pages where it appears.

[AH98] K. Andrews and H. Heidegger. Information slices: Visualising and exploring large hierarchies using cascading, semi-circular discs. In *Proc of IEEE Infovis 98 late breaking Hot Topics*, pages 9–11, 1998. 35

[AK07] K. Andrews and J. Kasanicka. A comparative study of four hierarchy browsers using the hierarchical visualisation testing environment (hvte). In *IV '07: Proceedings of the 11th International Conference Information Visualization*, pages 81–86, 2007. 52, 53, 57

[AMA07] D. Archambault, T. Munzner, and D. Auber. Topolayout: Multilevel graph layout by topological features. *Visualization and Computer Graphics, IEEE Transactions on*, 13(2):305 – 317, march-april 2007. 31

[And95] K. Andrews. Visualising cyberspace: information visualisation in the harmony internet browser. In *Information Visualization, 1995. Proceedings.*, pages 97–104, Oct. 1995. 2, 3, 68

[And06] K. Andrews. Evaluating information visualisations. In *BELIV '06: Proceedings of the 2006 AVI workshop on BEyond time and errors*, pages 1–5. ACM, 2006. 43, 99

[Ant09] D. Antón. Estudi i implementació de tècniques de millora del solapament a la metàfora visual del radial tree, 2009. Undergraduate Thesis Project under the supervision of Víctor Pascual Cid and Juan Carlos Dürsteler. 83

[AWS92] C. Ahlberg, C. Williamson, and B. Shneiderman. Dynamic queries for information exploration: An implementation and evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 619–626. ACM New York, NY, USA, 1992. 41

[Ber02] B. Berendt. Using site semantics to analyze, visualize, and support navigation. *Data Min. Knowl. Discov.*, 6(1):37–59, 2002. 27

[BHVW00] M. Bruls, K. Huizing, and J.J. Van Wijk. Squarified treemaps. In *Proceedings of the joint Eurographics and IEEE TCVG Symposium on Visualization*, pages 33–42, 2000. 35

[BN01] T. Barlow and P. Neville. A comparison of 2-d visualizations of hierarchies. In *INFOVIS '01: Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)*, page 131, 2001. 57

[BPMW98] S. Brin, L. Page, R. Motwami, and T. Winograd. The PageRank citation ranking: bringing order to the web. In *Proceedings of ASIS98*, pages 161–172, 1998. 28

[BRS92] R. A. Botafogo, E. Rivlin, and B. Shneiderman. Structural analysis of hypertexts: identifying hierarchies and useful metrics. *ACM Transactions on Information Systems (TOIS)*, 10(2):142–180, 1992. 21, 22, 68, 76

[BYP06] R. Baeza-Yates and B. Poblete. A website mining model centered on user queries. *Semantics, Web and Mining. M. Ackermann et al. (Eds.): EWMF/KDO 2005*, Springer LNAI 4289:1–17, 2006. 68

[BYRN+99] R. Baeza-Yates, B. Ribeiro-Neto, et al. *Modern information retrieval.* Addison-Wesley Reading, MA, 1999. 14

[Car08] S. Carpendale. Evaluating information visualizations. *Information Visualization. Human-Centered Issues and Perspectives (Chapter 2)*, 4950/2008:19–45, 2008. 112, 150

[CB01] C. K. Crook and D. Barrowcliff. Ubiquitous computing on campus: Patterns of engagement by university students. *International Journal on Human Computer Interaction*, 13(2):245–256, 2001. 111

[CC02] C. Chang and M. Chen. A new cache replacement algorithm for the integration of web caching and prefectching. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 632–634, New York, NY, USA, 2002. ACM. 26

[CCP09] C. Collins, S. Carpendale, and G. Penn. Docuburst: visualizing document content using language structure. In *Computer Graphics Forum*, volume 28, pages 1039–1046. John Wiley & Sons, 2009. 36

[Che05] C. Chen. Top 10 unsolved information visualization problems. *IEEE Computer Graphics and Applications*, 25(4):12–16, 2005. 42, 43

[Chi02] E. H. Chi. Improving web usability through visualization. *IEEE Internet Computing*, 6(2):64–71, 2002. 2

[CMS99a] S. K. Card, J. D. Mackinlay, and B. Shneiderman, editors. *Readings in information visualization: using vision to think.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999. 2, 9, 13

[CMS99b] R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, 1:5–32, 1999. xii, 24, 25, 88, 89, 115

[CPM+98] E. H. Chi, J. Pitkow, J. Mackinlay, P. Pirolli, R. Gossweiler, and S. K. Card. Visualizing the evolution of web ecologies. In *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 400–407, 1998. 69, 111

[CS99] J. Cugini and J. Scholtz. Visvip: 3d visualization of paths through web sites. In *DEXA '99: Proceedings of the 10th International Workshop on Database & Expert Systems Applications*, page 259. IEEE Computer Society, 1999. 69, 111

[CVN+07] F. Castro, A. Vellido, A. Nebot, F. Mugica, C. Nord, C.J. Girona, and M. Hidalgo. Applying data mining techniques to e-learning problems. *Intelligence (SCI)*, 62:183–221, 2007. 110

[CZT+07a] J. Chen, T. Zheng, W. Thorne, D. Huntley, O. R. Zayane, and R. Goebel. Visualizing web navigation data with polygon graphs. In *IV '07: Proceedings of the 11th International Conference Information Visualization*, pages 232–237. IEEE Computer Society, 2007. 69

[CZT+07b] J. Chen, T. Zheng, W. Thorne, O. R. Zaiane, and R. Goebel. Visual data mining of web navigational data. In *IV '07: Proceedings of the 11th International Conference Information Visualisation*, pages 649–656, 2007. 69

[DBETT98] G. Di Battista, P. Eades, R. Tamassia, and I.G. Tollis. *Graph* *drawing: algorithms for the visualization of graphs*. Prentice Hall PTR Upper Saddle River, NJ, USA, 1998. 30

[DK01] B. Diebold and M. Kaufmann. Usage-based visualization of web localities. In *APVis '01: Proceedings of the 2001 Asia-Pacific symposium on Information visualisation*, pages 159–164, Darlinghurst, Australia, Australia, 2001. Australian Computer Society, Inc. 91

[DKV99] J. Donath, K. Karahalios, and F. B. Viégas. Visualizing conversation. In *HICSS '99: Proceedings of the Thirty-Second Annual Hawaii International Conference on System Sciences-Volume 2*, page 2023. IEEE Computer Society, 1999. 132, 133

[DMM97] U. Dogrusöz, B. Madden, and P. Madden. Circular layout in the graph layout toolkit. In *GD '96: Proceedings of the Symposium on Graph Drawing*, pages 92–100, London, UK, 1997. Springer-Verlag. 31

[DWM04] K. Dave, M. Wattenberg, and M. Muller. Flash forums and forumreader: navigating a new kind of large-scale online discussion. *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*, Nov 2004. 131, 136

[D07a] J. C. Drsteler. Infovis diagram. `http://www.infovis.net/printMag.php?lang=2&num=187`, 2007. [Online; accessed 05-April-2010]. 12, 13, 15, 29, 39

[D07b] J. C. Drsteler. Infovis diagram. `http://www.infovis.net/printMag.php?num=186&lang=2`, 2007. [Online; accessed 05-April-2010]. 13

[Ead92] P. Eades. Drawing free trees. *Bulletin of the Institute for Combinatorics and its Applications*, 5(2):10–36, 1992. 37

[Edm67] J. Edmonds. *Optimum Branchings*. J. Res. Nat. Bur. Standards, 1967. 18, 92

[Eic01] S. G. Eick. Visualizing online activity. *Communications. ACM*, 44(8):45–50, 2001. 34

[EKS03] M. Eiglsperger, M. Kaufmann, and M. Siebenhaller. A topology-shape-metrics approach for the automatic layout of uml class diagrams. In *SoftVis '03: Proceedings of the 2003 ACM symposium on Software visualization*, pages 189–ff. ACM, 2003. 30

[Eng04] Y. Engelhardt. *The Language of Graphics - A framework* *for the analysis of syntax and meaning in maps, charts and diagrams.* PhD thesis, University of Amsterdam, Nederlands, 2004. 58

[ES93] K.A. Ericsson and H.A. Simon. *Protocol analysis: Verbal reports as data (Rev. ed.).* Cambridge, Ma: MIt Press, 1993. 43

[ESSF09] E. Enge, S. Spencer, J.C. Stricchiola, and R. Fishkin. *The Art of SEO.* Oreilly & Associates Inc, 2009. 28

[FdOL03] M.C. Ferreira de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: a survey. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):378–394, 2003. 2

[Few09] S. Few. *Now You See It. Simple Visualization Techniques for Quantitative Analysis.* Analytics Press, 2009. 9

[FGW02] U.M. Fayyad, G.G. Grinstein, and A. Wierse. *Information visualization in data mining and knowledge discovery.* Morgan Kaufmann Pub, 2002. 15

[FL05] F. M. Facca and P. L. Lanzi. Mining interesting knowledge from weblogs: a survey. *Data & Knowledge Engineering*, 53(3):225–241, 2005. 26

[FR91] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Softw. Pract. Exper.*, 21(11):1129–1164, 1991. 31

[Fry04] B. Fry. *Computational Information Design.* PhD thesis, Massachusetts Institute of Technology, 2004. 13

[Fry08] B. Fry. *Visualizing data.* O'Reilly, Jan 2008. 69

[Fur86] G.W. Furnas. Generalized fisheye views. *ACM SIGCHI Bulletin*, 17(4):23, 1986. 50

[GKL08] V. Gómez, A. Kaltenbrunner, and Vicente López. Statistical analysis of the social network and discussion threads in slashdot. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 645–654. ACM, 2008. 137, 138, 141, 142, 151

[HBM+06] J. Hardy, S. Bates, D. McKain, K. Murray, J. Paterson, B. McGonigle, L. Vigentini, and A. Jackson. The modus operandi of the next generation e-learner; an analysis of tracking usage across the disciplines. *Research Proceedings of the 13th Association of Learning Technology Conference*,

2006. 111, 124

[HC04] J. Heer and S.K. Card. Doitrees revisited: scalable, space-constrained visualization of hierarchical data. *Proceedings of the working conference on Advanced visual interfaces*, pages 421–424, 2004. 50

[HCL05] J. Heer, S.K. Card, and J.A. Landay. Prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430. ACM, 2005. 53, 63

[HF06] N. Henry and J.D. Fekete. Matrixexplorer: a dual-representation system to explore social networks. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):677–684, 2006. 50

[HL01a] J. I. Hong and J. A. Landay. Webquilt: a framework for capturing and visualizing the web experience. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pages 717–724. ACM, 2001. 69

[HL01b] J. I. Hong and J. A. Landay. Webquilt: a framework for capturing and visualizing the web experience. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pages 717–724. ACM, 2001. 111

[HM04] S.H. Hong and T. Murtagh. Visualisation of Large and Complex Networks Using PolyPlane. In *Graph Drawing*, pages 471–481. Springer, 2004. 38

[HS09] R. Hoekman and J. Spool. *Web Anatomy: Interaction Design Frameworks That Work*. New Riders Pub, 2009. 118

[HSW07] V. Hollink, M. Someren, and B. J. Wielinga. Navigation behavior models for link structure optimization. *User Modeling and User-Adapted Interaction*, 17(4):339–377, 2007. 27

[Ing99] A. Ingram. Using web server logs in evaluating instructional web sites. *Journal of educational technology systems*, 28(2)(137-57), 1999. 111

[JS91] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *VIS '91: Proceedings of the 2nd conference on Visualization '91*, pages 284–291, Los Alamitos, CA, USA, 1991. IEEE Computer Society Press. 34, 35

[Kau07] A. Kaushik. *Web analytics: an hour a day*. Sybex, Jan 2007.

27, 70, 88

[KB00] R. Kosala and H. Blockeel. Web mining research: A survey. *ACM SIGKDD Explorations Newsletter*, 2(1):1–15, 2000. 19, 20

[KE02] T. A. Keahey and S. G. Eick. Visual path analysis. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02)*, page 165. IEEE Computer Society, 2002. xiii, 70, 71, 111

[Kei02] D. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, Jan 2002. 41

[KGM+08] A. Kaltenbrunner, V. Gómez, A. Moghnieh, R. Meza, J. Blat, and V. López. Homogeneous temporal activity patterns in a large online communication space. *IADIS International Journal on WWW/INTERNET*, 6(1):61–76, 2008. 141

[KJ56] J.B. Kruskal Jr. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 7(1):48–50, 1956. 18

[KJ06] B. Kim and P. Johnson. Graphical interface for visual exploration of online discussion forums. *Computer Science Faculty Publications*, Jan 2006. 135

[KK89] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Inf. Process. Lett.*, 31(1):7–15, 1989. 31

[KMSZ06] D.A. Keim, F. Mansmann, J. Schneidewind, and H. Ziegler. Challenges in visual data analysis. In *Information Visualization, 2006. IV 2006. Tenth International Conference on*, pages 9 –16, jul. 2006. 11

[Kob04] A. Kobsa. User experiments with tree visualization systems. In *INFOVIS '04: Proceedings of the IEEE Symposium on Information Visualization*, pages 9–16, 2004. 52, 58

[LBOT00] B. Lan, S Bressan, B. Ooi, and K. Tan. Rule-assisted prefetching in web-server caching. In *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*, pages 504–511, New York, NY, USA, 2000. ACM. 26

[LR04a] C. Lampe and P. Resnick. Slash(dot) and burn: Distributed moderation in a large online conversation space. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in* *computing systems*, pages 543–550. ACM Press, 2004. 141

[LR04b] C. Lampe and P. Resnick. Slash(dot) and burn: distributed moderation in a large online conversation space. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 543–550, 2004. 142

[LRP95] J. Lamping, R. Rao, and P. Pirolli. A focus+ context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, page 408. ACM Press/Addison-Wesley Publishing Co., 1995. 38

[LT07] J. L. Ledford and M. E. Tyler. *Google Analytics 2.0*. Wiley, 2007. 26, 27

[LTT09] J. L. Ledford, J. Teixeira, and M. E. Tyler. *Google Analytics, 3rd Edition*. Wiley; Pap/Dig edition, 2009. 110

[MB95] T. Munzner and P. Burchard. Visualizing the structure of the world wide web in 3D hyperbolic space. In *Proceedings of the first symposium on Virtual reality modeling language*, page 38. ACM, 1995. 2, 3, 38, 42, 68

[MBC+05] J. Mostow, J. Beck, H. Cen, A. Cuneo, E. Gouvea, and C. Heiner. An educational data mining tool to browse tutor-student interactions: Time will tell. *Proceedings of the Workshop on Educational Data Mining, National Conference on Artificial Intelligence*, pages 15–22, 2005. 111

[McG95] J.E. McGrath. Methodology matters: Doing research in the behavioral and social sciences. In *Human-computer interaction*, page 169. Morgan Kaufmann Publishers Inc., 1995. 158

[MEQ09] R. Navarro M. Estrada and M. Quixal. Report of the evaluation in the second testing phase. technical report. autolearn project. Technical report, Barcelona Media - Innovation Centre, 2009. 120, 129

[MG05] D. Mladenic and M. Grobelnik. Visualizing very large graphs using clustering neighborhoods. *Lecture Notes in Computer Science: Local Pattern Detection*, 3539/2005, 2005. 50

[Mun98] T. Munzner. Drawing large graphs with h3viewer and site manager. In *GD '98: Proceedings of the 6th International Symposium on Graph Drawing*, pages 384–393. Springer-Verlag, 1998. 68

[Mun09] T. Munzner. A nested process model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):921–928, 2009. 2, 44, 99, 107, 154

[NC10] S. Narayan and C. Cheshire. Not too long to read: The tldr interface for exploring and navigating large-scale discussion spaces. In *The 43rd Annual Hawaii International Conference on System Sciences. Persistent Conversations Track*. ACM, 2010. 132, 137, 144, 145

[PB94] J. E. Pitkow and K. A. Bharat. Webviz: A tool for world-wide web access log analysis. In *Proceedings of the 1st International WWW Conference*, pages 271–277, 1994. 3, 69

[PBY06] B. Poblete and R. Baeza-Yates. A content and structure website mining model. In *Proceedings of the 15th international conference on World Wide Web*, page 958. ACM, 2006. 23

[PC08] V. Pascual-Cid. An information visualisation system for the understanding of web data. Poster at IEEE Symposium on Visual Analytics Science and Technology, 2008. VAST '08., Oct 2008. 107

[PCBYD+09] V. Pascual-Cid, R. Baeza-Yates, J.C. Dürsteler, S. Mínguez, and C. Middleton. New techniques for visualising web navigational data. *Proceedings of the 13th Conference in Information Visualisation (IV09)*, pages 621 –626, Jul 2009. 107

[PCBYD10] V. Pascual-Cid, R. Baeza-Yates, and J.C. Dürsteler. Visual web mining for website evaluation. *Journal of Web Engineering, Rinton Press*, 2010. To appear. 107

[PCK09] V. Pascual-Cid and A. Kaltenbrunner. Exploring asynchronous online discussions through hierarchical visualisation. In *Proceedings of the 2009 13th International Conference Information Visualisation*, pages 191–196. IEEE Computer Society, 2009. 152

[PCVQ10] V. Pascual-Cid, L. Vigentini, and M. Quixal. Visualising virtual learning environments: Case studies of the website exploration tool. *Information Visualisation, International Conference on*, 0:149–155, 2010. 129

[PD07] V. Pascual and J.C. Drsteler. Wet: a prototype of an exploratory search system for web mining to assess usability. *Proceedings of the 11th International Conference in Information Visualisation (IV07)*, Jan 2007. 107

[PHMAZ00] J. Pei, J. Han, B. Mortazavi-Asl, and H. Zhu. Mining access patterns efficiently from web logs. In *PADKK '00: Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications*, pages 396–407, London, UK, 2000. Springer-Verlag. 91

[PL08] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135, 2008. 22

[Pla04] C. Plaisant. The challenge of information visualization evaluation. *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, May 2004. 2, 42, 44, 112, 154

[PPPS03] D. Pierrakos, G. Paliouras, C. Papatheodorou, and C. D. Spyropoulos. Web usage mining as a tool for personalization: A survey. *User Modeling and User-Adapted Interaction*, 13(4):311–372, 2003. 26

[Pri57] R.C. Prim. Shortest connection networks and some generalizations. *Bell system technical journal*, 36(6):1389–1401, 1957. 18

[PS08] A. Perer and B. Shneiderman. Integrating statistics and visualization: case studies of gaining clarity during exploratory data analysis. *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 265–274, 2008. 112

[PS09] A. Perer and B. Shneiderman. Integrating statistics and visualization for exploratory power: From long-term case studies to design guidelines. *Computer Graphics and Applications, IEEE*, 29(3):39–51, May 2009. 44, 112

[Ram03] P. Ramsden. Learning to teach in higher education. *RoutledgeFalmer, London*, 2003. 109

[RMC91] G.G. Robertson, J.D. Mackinlay, and S.K. Card. Cone trees: animated 3d visualizations of hierarchical information. *Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through technology*, pages 189–194, 1991. 37, 68

[RT81] E. M. Reingold and J. S. Tilford. Tidier drawings of trees. *IEEE Trans. Softw. Eng.*, 7(2):223–228, 1981. 33

[RV07] C. Romero and S. Ventura. Educational data mining: A survey from 1995 to 2005. *Expert Syst. Appl.*, 33(1):135–146, 2007. 110

[Sac00a] W. Sack. Conversation map: a content-based usenet newsgroup browser. In *IUI '00: Proceedings of the 5th international conference on Intelligent user interfaces*, pages 233–240. ACM, 2000. 134

[Sac00b] W. Sack. Discourse diagrams: Interface design for very large-scale conversations. In *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 3*, page 3034. IEEE Computer Society, 2000. 134

[SB92] M. Sarkar and M. H. Brown. Graphical fisheye views of graphs. In *CHI '92: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 83–91, New York, NY, USA, 1992. ACM. 83

[Sed10] M. Sedlmair. *Visual Analysis of In-Car Communication Networks*. PhD thesis, University of Munich (LMU), 2010. 156, 158

[SF01] M. A. Smith and A. T. Fiore. Visualization components for persistent conversations. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 136–143. ACM, 2001. 136, 152

[Shn81] B. Shneiderman. Direct manipulation: A step beyond programming languages (abstract only). In *Proceedings of the joint conference on Easier and more productive use of computer systems.(Part-II): Human interface and the user interface-Volume 1981*, page 143. ACM, 1981. 41

[Shn96] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, page 336, 1996. 14, 40, 118, 138

[SHS10] H. Schulz, S. Hadlak, and H. Schumann. The design space of implicit hierarchy visualization: A survey. *Visualization and Computer Graphics, IEEE Transactions on*, PP(99):1 –1, 2010. 36

[SNS06] Hans-Jorg Schulz, Thomas Nocke, and Heidrun Schumann. A framework for visual data mining of structures. In *ACSC '06: Proceedings of the 29th Australasian Computer Science Conference*, pages 157–166, Darlinghurst, Australia, Australia, 2006. Australian Computer Society, Inc. 16, 17

[SP01] M. Spiliopoulou and C. Pohle. Data mining for measuring and improving the success of web sites. *Data Min. Knowl. Discov.*, 5(1-2):85–114, 2001. 27

[SP06] B. Shneiderman and C. Plaisant. Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies. *BELIV '06: Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, May 2006. 44, 112

[Spi00] M. Spiliopoulou. Web usage mining for web site evaluation. *Communications of the ACM*, 43(8):127–134, 2000. 68

[SSW05] S. Sobol, C. Stones, and A. Whitworth. Qualitative evaluation of e-learning using a visualisation tool. *Proceedings, IADIS Conference on Applied Computing*, 2005. 111

[STT81] K. Sugiyama, S. Tagawa, and M. Toda. Methods for visual understanding of hierarchical system structures. *Systems, Man and Cybernetics, IEEE Transactions on*, 11(2):109 – 125, feb. 1981. 31, 33

[SZ00] J. Stasko and E. Zhang. Focus+ context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *Proceedings of the IEEE Symposium on Information Vizualization*, page 57. Citeseer, 2000. 36

[TC05] J.J. Thomas and K.A. Cook. Illuminating the path: The research and development agenda for visual analytics. *IEEE Computer Society*, 2005. xii, 2, 10, 11, 44, 49, 62, 154

[TK02] P. Tan and V. Kumar. Discovery of web robot sessions based on their navigational patterns. *Data Min. Knowl. Discov.*, 6(1):9–35, 2002. 25

[TM04] M. Tory and T. Moller. Human factors in visualization research. *IEEE transactions on visualization and computer graphics*, 10(1):72–84, 2004. 43

[TM05] M. Tory and T. Moller. Evaluating visualizations: do expert reviews work? *IEEE Computer Graphics and Applications*, 25(5):8–11, 2005. 43

[TS07] O. Turetken and R. Sharda. Visualization of web spaces: state of the art and future directions. *SIGMIS Database*, 38(3):51–81, 2007. 18, 68

[vHP09] F. van Ham and A. Perer. "Search, Show Context, Expand on Demand": Supporting Large Graph Exploration with Degree-of-Interest. *IEEE Transactions on Visualization* *and Computer Graphics*, 15(6):953–960, 2009. 50

[Vig09] L. Vigentini. Using learning technology in university courses: do styles matter? *Multicultural Education and Technology Journal*, 3, 2009. 121

[Vig10] L. Vigentini. *From Learning to e-Learning: A psychological framework to evaluate the individual differences in students interaction with learning technology*. PhD thesis, University of Edinburgh, 2010. To appear. 129

[VS04] F. Viegas and S. Smith. Newsgroup crowds and authorlines: visualizing the activity of individuals in conversational. *System Sciences*, Jan 2004. xiv, 134, 135

[War04] C. Ware. *Information visualization: perception for design*. Morgan Kaufmann, 2004. 10, 13, 29, 40, 42

[WGFS07] H. T. Welser, E. Gleave, D. Fisher, and M. Smith. Visualizing the signatures of social roles in online discussion groups. *The Journal of Social Structure*, 8(2), 2007. 138

[WHA07] W. Willett, J. Heer, and M. Agrawala. Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1129–1136, 2007. 63

[WI06] J.Q. Walker II. A node-positioning algorithm for general trees. *Software: Practice and Experience*, 20(7):685–705, 2006. 33

[WLSW01] R.S. Wurman, L. Leifer, D. Sume, and K. Whitehouse. *Information anxiety 2*. Que, 2001. 13

[WPCM02] C. Ware, H. Purchase, L. Colpoys, and M. McGill. Cognitive measurements of graph aesthetics. *Information Visualization*, 1(2):103–110, 2002. 30

[WR09] T. Wang and Y. Ren. Research on personalized recommendation based on web usage mining using collaborative filtering technique. *WSEAS Trans. Info. Sci. and App.*, 6(1):62–72, 2009. 26

[XD99] R. Xiong and J. Donath. Peoplegarden: creating data portraits for users. In *UIST '99: Proceedings of the 12th annual ACM symposium on User interface software and technology*, pages 37–44. ACM, 1999. 134

[YFDH01] K.P. Yee, D. Fisher, R. Dhamija, and M. Hearst. Animated exploration of dynamic graphs with radial layout. In *IEEE*

Symposium on Information Visualization, 2001. INFOVIS 2001, pages 43–50, 2001. 59

[ZSN+06] T. Zuk, L. Schlesier, P. Neumann, M.S. Hancock, and S. Carpendale. Heuristics for information visualization evaluation. In *Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, page 6. ACM, 2006. 43