ANNOTATION AND TRANSFORMATION OF WEB PAGES TO IMPROVE MOBILITY FOR VISUALLY IMPAIRED USERS

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

2005

By
Yeliz Yesilada
School of Computer Science

Contents

Abstract **10**

Declaration **11**

Acknowledgements **12**

1 Introduction **13**

1.1 Web Accessibility 14
1.2 The Problem Statement 14
1.3 Research Questions and Objectives 16
1.4 Thesis Outline 21

2 Literature Review **22**

2.1 Alternative Web Browsing 22
2.1.1 Screen Readers 23
2.1.2 Specialised Browsers 25
2.2 Audio Versus Visual Web Navigation 27
2.3 Web Accessibility Guidelines 29
2.4 Tools to Support Web Accessibility 32
2.4.1 Evaluation Tools 32
2.4.2 Repair Tools 35
2.4.3 Transformation Tools 36
2.4.4 Summary and Conclusions 44
2.5 Semantic Web and Accessibility 45
2.6 Summary and Conclusions 46

3 The Travel Analysis Framework **48**

3.1 Travel and Mobility on the Web 48

3.2 Travel Objects 50
3.2.1 Way Points 51
3.2.2 Orientation Points 54
3.2.3 Travel Assistants 55
3.3 The Travel Analysis Framework 56
3.3.1 Identification of Travel Objects 56
3.3.2 Classification of Travel objects 59
3.4 Evaluation of the Framework 60
3.4.1 Results and Discussion 62
3.5 Summary and Conclusions 68

4 The WAfA Ontology 70
4.1 Motivation and the Purpose 70
4.2 The Design Process 72
4.3 The Implementation 74
4.3.1 Authoring Semantics for Conceptual Structuring 78
4.3.2 Mobility Semantics for Navigational Structuring 83
4.3.3 Contextual Semantics 84
4.4 Summary and Conclusions 86

5 Dante: Annotating and Transforming Web Pages 88
5.1 Dante Approach and its Architecture 88
5.2 Annotating Web Pages 92
5.3 The Dante RDF Editor: Modelling Web Pages 97
5.4 Transforming Web Pages 99
5.4.1 Generic Heuristics 101
5.4.2 Object Specific Heuristics 109
5.4.3 Customisation of Heuristics 111
5.4.4 Combination of Heuristics 112
5.5 Summary and Conclusions 115

6 The Evaluation of Dante 116
6.1 The User Evaluation 117
6.1.1 Participants 117
6.1.2 Materials 118
6.1.3 Tasks 118

6.1.4 Methods 123
6.1.5 Procedure 124
6.1.6 Mental Workload: TLX Results 126
6.1.7 Results and Discussion 129
6.2 The Technical Evaluation 139
6.3 Summary and Conclusions 140

7 Summary and Conclusions 142
7.1 Thesis Overview 142
7.2 Contributions of the Thesis 143
7.2.1 The Travel Analysis Framework 144
7.2.2 The WAfA Ontology 144
7.2.3 Dante: Annotating and Transforming Web Pages 145
7.3 Outstanding Issues 147
7.4 Future Work 148

Bibliography 151

A The Travel Analysis Framework 163
A.1 The Evaluation Materials 163

B The WAfA Ontology 165
B.1 Authoring Semantics 165
B.1.1 The Concept ‘Atom’ 165
B.1.2 The Concept ‘Chunk’ 170
B.1.3 Concepts to describe Authoring Semantics 176
B.1.4 Properties of Authoring Semantics 179
B.1.5 Meta-Knowledge about Authoring Semantics 181
B.2 Contextual Semantics 182

C The User Evaluation 184
C.1 The Evaluation Materials 184
C.2 The Evaluation Methods 188
C.3 The Evaluation Results 190
C.4 TLX Scores 192

List of Tables

2.1 Screen readers. 24
2.2 Web accessibility guidelines. 30
2.3 Summary of the Web Content Accessibility Guidelines (WCAG 1.0). . 30
2.4 Evaluation tools. 33
2.5 Repair tools. 35
2.6 Transformation tools. 38

3.1 Travel objects and examples. 52
3.2 Travel objects on the home page of Google Directory. 64
3.3 Travel objects on the home page of RNIB. 65
3.4 Time taken for analysing the Google Directory and the RNIB page. . . 66

4.1 The overview of the WAfA ontology. 76
4.2 Example travel objects on the home page of Royal Mail. 84
4.3 Mobility semantics. 85

5.1 Example annotations done on the home page of Java (see Figure 5.6). 95
5.2 Example relationships on the home page of Java (Table 5.1). 99
5.3 Analyses of transformation techniques with respect to mobility. 102
5.4 Customisation of some transformation heuristics. 112
5.5 Combination of transformation heuristics if in-situ same page. 114

6.1 Tasks used in the main body of the evaluation. 121
6.2 Analyses of tasks (Table 6.1) with respect to mobility principles. . . . 122
6.3 Analyses of tasks (Table 6.1) with respect to mobility instruments. . . 123
6.4 Post-evaluation questions with respect to mobility. 125
6.5 Summary of TLX scores for the home page of Manchester University. 128
6.6 Summary of TLX scores for the home page of IMDB. 129

A.1 Questions for the overall evaluation of the framework. 163

B.1 The overview of the Authoring Semantics. 166
B.2 The overview of the concept “Atom”. 167
B.3 The overview of the concept “Chunk”. 171
B.4 The overview of the concept “NavigationalList” (see Table B.3). . . . 172
B.5 The overview of the concept “OrderedList” (see Table B.3). 172
B.6 The overview of the concepts used to describe “AuthoringConcepts” (see Table B.1). 177
B.7 The overview of the properties used to describe “AuthoringConcepts”. 179
B.8 The overview of the concepts used to describe the concept “Purpose”. 182

C.1 NASA Task Load Index (TLX) details. 189
C.2 Profiles of the participants. 191
C.3 Individual and mean rating for TLX factors for Figure C.3 and C.4. . . 192
C.4 Individual and mean rating for TLX factors for Figure C.5 and C.6. . . 192
C.5 Summary of the structured interview results for the original version of home page of Manchester University (see Figure C.3). 195
C.6 Summary of the structured interview results for the transformed version of the home page of Manchester University (see Figure C.4). . . 195
C.7 Summary of the structured interview results for the original version of the home page of IMDB (see Figure C.5). 196
C.8 Summary of the structured interview results for the transformed version of the home page of IMDB (see Figure C.6). 196

List of Figures

1.1 The home page of ACM (07/02/2005). 16
1.2 The Dante approach and the structure of the thesis. 19

2.1 The home page of Amazon (20/06/2004). 25
2.2 The home page of Amazon in audio (20/06/2004). 25

3.1 Travel Objects. 51
3.2 Example travel objects on the home page of HSBC (03/10/2004). . . . 57
3.3 The home page of Google Directory (12/03/2002). 60
3.4 The home page of RNIB (12/06/2002). 61

4.1 The home page of Royal Mail (03/10/2004). 72
4.2 The design process of the WAfA ontology. 73
4.3 Knowledge about a Web page. 77
4.4 *part-of* hierarchy vs. *is-a* hierarchy. 78
4.5 Relationships between top-level concepts of Authoring Semantics. . . 81
4.6 The specialisation of the concept “List”. 82

5.1 The basic architecture of Dante. 89
5.2 The home page of ACM transformed by the Dante pipeline. 91
5.3 Example RDF triples of an Annotea annotation. 93
5.4 The architecture of the COHSE Annotator. 93
5.5 Annotating the home page of Java with the WAfA ontology. 94
5.6 The home page of Java (15/01/2004). 95
5.7 The Dante annotation pipeline (see Figure 5.1 for the parts 3 and 4). . 96
5.8 The Dante RDF Editor creating relationship(s) between annotations. . 98
5.9 RDF Triples concerning the “Header” object on Figure 5.6. 100
5.10 Fragmentation of the home page of Java with a linear table of contents. 103

5.11 Fragmentation of the home page of Java with a linear table of contents (with guided tour support). 104
5.12 Fragmentation of the home page of Java with a nested table of contents. 104
5.13 Logical fragmentation of the home page of Java. 105
5.14 The home page of Java without header and footer. 106
5.15 The home page of Java without sidebars. 107
5.16 The home page of Java with a skip link. 108
5.17 Linearisation of the main content of the home page of Java. 108
5.18 The search engine in the home page of Java is moved to the bottom. . 110
5.19 An advertisement is moved to the bottom of the page and to a new file. 110

6.1 A result page of Google (17/08/2004). 119
6.2 The home page of Manchester University (17/08/2004). 119
6.3 The home page of Manchester University in audio. 119
6.4 The home page of IMDB (17/08/2004). 120

7.1 Possible applications of Dante. 149

A.1 The travel object classification questionnaire. 164

C.1 A result page of Google (17/08/2004). 185
C.2 A physical fragmentation of C.1. 185
C.3 The home page of Manchester University (17/08/2004). 186
C.4 A physical fragmentation of C.3. 186
C.5 The home page of IMDB (17/08/2004). 187
C.6 A physical fragmentation of C.5. 187
C.7 Comparing TLX *Mental Demand* for Figure C.3 and C.4. [O_{Mean}=13.88 T_{Mean}=5.63]. 193
C.8 Comparing TLX *Physical Demand* for Figure C.3 and C.4. [O_{Mean}=12.75 T_{Mean}=5.00] 193
C.9 Comparing TLX *Temporal Demand* for Figure C.3 and C.4. [O_{Mean}=10.38 T_{Mean}=4.50]. 193
C.10 Comparing TLX *Effort* for Figure C.3 and C.4. [O_{Mean}=13.63 T_{Mean}=4.38] 193
C.11 Comparing TLX *Performance* for Figure C.3 and C.4. [O_{Mean}=9.88 T_{Mean}=4.63] 193

C.12 Comparing TLX *Frustration Level* for Figure C.3 and C.4.
[O_{Mean}=15.00 T_{Mean}=3.38] 193
C.13 Comparing TLX *Mental Demand* for Figure C.5 and C.6.
[O_{Mean}=15.67 T_{Mean}=4.50]. 194
C.14 Comparing TLX *Physical Demand* for Figure C.5 and C.6.
[O_{Mean}=13.50 T_{Mean}=2.83] 194
C.15 Comparing TLX *Temporal Demand* for Figure C.5 and C.6.
[O_{Mean}=11.50 T_{Mean}=2.50]. 194
C.16 Comparing TLX *Effort* for Figure C.5 and C.6.
[O_{Mean}=14.33 T_{Mean}=3.17] 194
C.17 Comparing TLX *Performance* for Figure C.5 and C.6.
[O_{Mean}=16.17 T_{Mean}=3.50] 194
C.18 Comparing TLX *Frustration Level* for Figure C.5 and C.6.
[O_{Mean}=16.33 T_{Mean}=2.00] 194

Abstract

In the last decade, the growth and the popularity of the World Wide Web (Web) have been phenomenal. Originally, it was a purely text-based system that allowed assistive technologies to be designed to transform pages into alternative forms (e.g., audio) for disabled people. This meant that for the first time, a vast amount of information was available and easily accessible to disabled people. However, advances in technologies and changes in the main authoring language, transformed the Web into a true visual communication medium. These changes eventually made the Web inaccessible to visually impaired users. In particular, travelling around the Web became a complicated task, since the richness of visual navigational objects presented to their sighted counterparts are neither appropriate nor accessible to visually impaired users.

This thesis investigates principles and derived techniques to enhance the travel experience for visually impaired Web users. The hypothesis is that travel support for visually impaired users can be improved if Web pages are analysed to identify the objects that support travel and are then transformed in such a way that they can then fulfill their intended or implicit roles. This hypothesis is supported by the identification of structural and navigational properties of Web pages which have been encapsulated into an ontology (WAfA) to support machine processing; and the design of a flexible pipeline approach to annotate and transform Web pages by using this ontology. An experimental test-bed, Dante, has also been created based on this pipeline approach, that encodes these principles and techniques to transform Web pages. Finally, a user evaluation method is devised and applied to demonstrate that the travel experience of visually disabled users can be improved through the application of these techniques.

This research demonstrates that by providing machine processable data, regarding the structural and navigational properties of Web pages, applications can be created to present Web pages in alternative forms and so enhance the travel experience of visually impaired users. The work presented in this thesis is of practical value to the Web accessibility community and is an important case study for demonstrating Semantic Web technologies. By moving away from thinking that simple translation of text to audio is enough to provide access to Web pages, this thesis is part of the coming of age of assistive technology and is a significant contribution to redressing the inequality arising from visual dominance in the Web.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institution of learning.

1. Copyright in text of this thesis rests with the Author. Copies (by any process) either in full, or of extracts, may be made **only** in accordance with instructions given by the Author and lodged in the John Rylands University Library of Manchester. Details may be obtained from the Librarian. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the Author.

2. The ownership of any intellectual property rights which may be described in this thesis is vested in the University of Manchester, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the University, which will prescribe the terms and conditions of any such agreement.

Further information on the conditions under which disclosures and exploitation may take place is available from the head of School of Computer Science.

Acknowledgements

Completing this PhD. project could never have happened without the help and support of some people whom I would like to thank.

Firstly I would like to thank to my supervisors Carole Goble and Robert Stevens for all their support, advice and encouragement. I am also grateful to Simon Harper for his guidance and discussions we had which have been of great value in this project.

I would like to express my gratitude to the members of the Information Management Group (IMG) for creating such a friendly atmosphere, and for helping and inspiring me. Thanks should also go to everyone who participated in both evaluations.

This work has been supported by the Overseas Research Studentship (ORS) and the School of Computer Science of the University of Manchester.

My whole family, particularly my parents, have been both encouraging and supportive throughout the completion of this project, for which I am forever indebted. Thanks should also go to my grandmother who persuade me to do this PhD. I wish she could have seen that at the end I really made it!

This thesis is dedicated to my husband Nihat, without whose love and support this would not have been possible.

Chapter 1

Introduction

The World Wide Web (Web), the famous branch of the Internet, was developed in the early nineties [13]. Since then, the growth and popularity of the Web has been phenomenal. One of the many reasons for this is the simplicity of the language used to compose pages – the HyperText Markup Language (HTML) [80]. Originally, the Web started off as a purely text-based system. This allowed assistive technologies [3] to be designed to transform pages into alternative forms (e.g., audio) for disabled people. This meant that for the first time vast quantities of information were easily accessible to visually impaired people. However, advances in technologies and also changes in HTML itself, allowed Web pages to become true visual communication media. These transformations eventually made the Web inaccessible to disabled users [109]. In particular, for visually impaired users travelling around the Web is a complicated task, since the richness of visual navigational objects presented to sighted users are neither appropriate nor accessible.

This thesis proposes techniques and principles to enhance the travel experience of visually impaired users on the Web. The hypothesis is that the travel experience of visually impaired users can be improved if Web pages are analysed to identify objects that support travel and are transformed in a way that these objects can fulfill their intended roles. In supporting this hypothesis it is claimed that by identifying these objects and adding semantics to make them computationally available, a computer application can be designed and developed to transform Web pages by using these semantics and different techniques to enhance the travel support provided.

1.1 Web Accessibility

Web Accessibility refers to the practice of making pages on the Web accessible to all users, especially to those with disabilities [87, 109]. Disabled people usually use assistive technologies[1] to access Web pages in alternative forms (e.g, audio, Braille). Although an accessible Web means unprecedented access to information for people with disabilities, recent research suggests that the best practice on accessibility has not yet been achieved. For example, [61, 68] found the accessibility of the high street stores, banks and universities in the UK extremely disappointing. [2] surveyed 20 'Flagship' governmental Web sites in the UK and concluded that 75% needed immediate attention in one area or another. Recently, the Disability Rights Commission (DRC) also conducted an extensive user evaluation whose report [3] concludes that most Web sites (81%) fail to satisfy even basic accessibility requirements.

The Web plays an increasingly important role in many areas (e.g., education, government) so an accessible Web that allows people with disabilities to actively participate in society is essential for equal opportunities in those areas[2]. Furthermore, Web accessibility is not only a social issue but it is also becoming a legal requirement [87, 109]. Nations and continents including the UK[3], Australia, Canada and the United States are approving specific legislation to enforce Web accessibility.

The research presented in this thesis contributes to the knowledge in the Web accessibility field. The main goal of this research is to understand how to provide better accessibility support for visually impaired users[4]. Therefore, before we present the objectives of this thesis, in the following section we present how visually impaired users access Web pages and what kind of problems they have, and discuss the work that has been done in this field.

1.2 The Problem Statement

Visually impaired users usually access Web pages by using screen readers. Screen readers are assistive technologies that render pages in audio and allow either simple line based interaction or full reading of a page, from top to bottom, one word at a

[1] **Assistive technology** refers to hardware and software designed to facilitate the use of computers by people with disabilities [3].

[2] See `http://www.w3.org/WAI/intro/accessibility`.

[3] See `http://www.drc-gb.org/open4all/law/`.

[4] The term **visually impaired** is used as a generic phrase for all those people who have either little or no vision not corrected by human artifice.

time. While screen readers enable visually impaired users to access the Web, it is still a difficult task. For example, [32] surveyed Web sites and concluded that "the Web is about three times more difficult to use for users who are visually impaired than it is for sighted users. The DRC report also states that among disabled groups studied (e.g., blind, dyslexic), "blind users constituted the most disenfranchised group" [3].

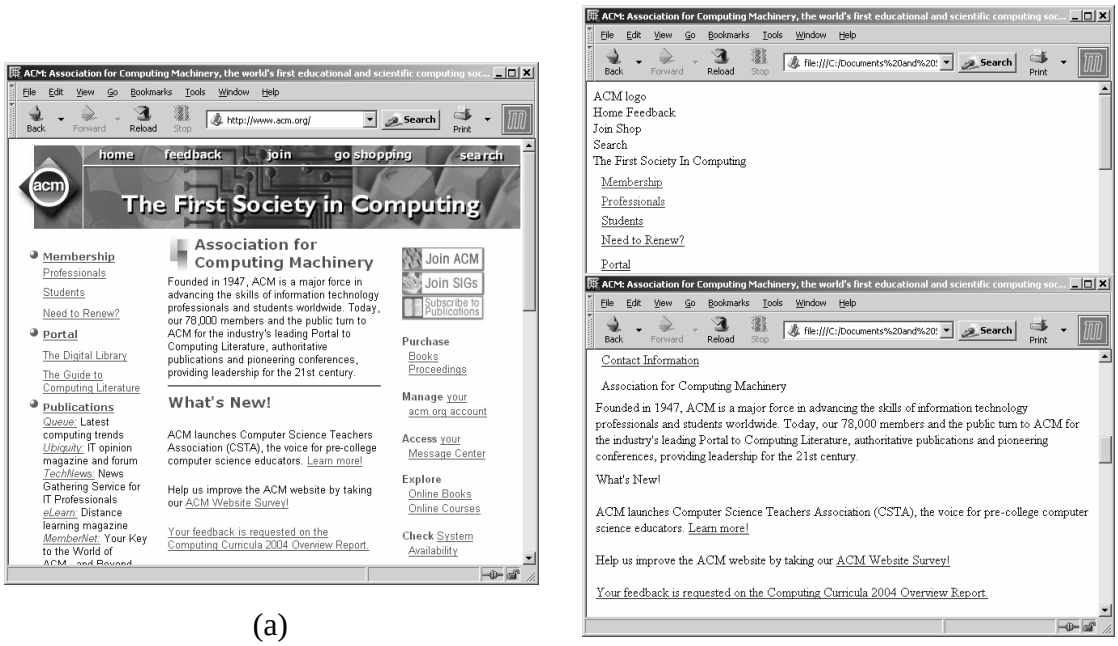
Inappropriately designed Web pages and the insufficiency of currently available assistive technologies are the two main reasons for the problems that visually impaired users encounter [55]. Although, guidelines are introduced to promote accessibility on the Web [32, 49], few pages conform to these guidelines [3]. Nowadays, Web pages are designed in a complex and nonlinear fashion, and the focus is on visual interaction [82, 40]. The home page of ACM[5] is a good example (see Figure 1.1). The page is visually quite well organised, for example the navigation links are separated from the main content by using sidebars, page title is highlighted in the header, etc. While these visual elements are mainly used to create aesthetically pleasing environments, they actually provide visual guidance to ease the access to that environment [119]. However, as these spatial cues are not explicitly specified in the source code, they are not accessible to screen readers therefore to visually impaired users. For example, Figure 1.1 (b) shows how the ACM home page is rendered by a screen reader in audio; the page is actually perceived as a long list of items. Screen readers focus on directly translating content of a page to audio rather than supporting easy movement within or between pages.

There are tools to evaluate Web pages against accessibility guidelines and repair any contravention [8, 9, 87, 109]. Although these are important to promote Web accessibility, they mainly lack an understanding of how visually impaired people interact with Web pages in audio; what kinds of objects they need or use, and what kind of spatial environment is easier for them to navigate, etc.

The accessibility issues discussed here can be summarised as follows:

- Visually impaired users have difficulties accessing the Web [2, 3, 33, 61, 68];
- Most Web pages do not satisfy even basic accessibility requirements [3];
- Screen readers cannot see the structure of the document (e.g., proximity or grouping of the objects) and the implicit visual cues (e.g., typography or font size) that play a crucial role in understanding the content of the document [119];
- Existing work does not provide an understanding of how visually impaired users access the Web in audio [8, 9, 87, 109].

[5] See `http://www.acm.org/`.



(a) (b)

Figure 1.1: The home page of ACM (07/02/2005).

1.3 Research Questions and Objectives

In this section, we discuss a number of questions which arise from the issues discussed in the previous section and from our hypothesis, present the objectives of this thesis based on these questions and introduce the contributions to meet these objectives.

(Q1) – How do visually impaired users travel around Web pages?

Existing research suggests that there is more work to do on understanding how visually impaired users really move around and between Web pages [46]. This lack of understanding led to poor travel support for visually impaired Web users [53, 54, 55].

Harper [53] introduced the notion of travel and mobility on the Web to improve the accessibility of Web pages for visually impaired users by drawing an analogy between virtual travel and travel in the physical world. **Travel** is defined as the confident **navigation** and **orientation** with **purpose**, ease and accuracy within an **environment**. **Navigation** suggests an opportunity of movement within the local environment. **Orientation** is the knowledge of the basic spatial relationships between objects within the environment, and the objects and the traveller. **Mobility** is the ease and confidence at which travel can be accomplished. **Environment** is the context in which the traveller journeys through and includes the way the landscape is rendered and perceived.

In order to be able to adapt and transfer the findings and observations of real world travel studies to the virtual Web world, a model of travel has been proposed [46]. This model consists of three components: objects (e.g., landmarks), actions (e.g., landmark detection) and instruments (e.g., preview). Travellers navigate and orientate themselves by consulting memory *objects* and detecting and identifying landmarks. Consultation, detection and identification are accomplished through the *mobility instruments* of in-journey guidance, previews, probes and feedbacks. Even though this model provides an understanding of how visually impaired people travel on the Web, it still does not provide comprehensive knowledge about the objects that traveller use to accomplish the purpose of their journey [125] (see Chapter 2 and 3).

(Q2) – What kind of objects do visually impaired people use or need to use in an environment to complete a successful journey?

In the real world, [73, 88] suggest that travellers use or may need to use environmental features or elements in order to complete a successful journey. We call these features and elements **travel objects**. For instance, they use landmarks and memory objects to reassure themselves that they are safe to proceed and are going the right way. If we look at the virtual Web landscape from the *travel* perspective these objects also exist on Web pages. For example, in Figure 1.1 (a), the logo of the ACM can be considered as a unique and memorable feature of the environment [66]. These kinds of objects are considered as landmarks or reference points. They provide users with information on where they are in a page, just as unique features aid navigation in physical environments [37]. However, in the audio rendering of Web pages, these objects cannot play their roles (see Figure 1.1 (b)). Assistive technologies (i.e., screen readers) do not treat them in a special and appropriate form, so their role in supporting the travel is diminished. They need to be presented in a way that they can fulfill their intended roles and support travel on the Web. But first we need to identify these objects and their roles in detail.

This thesis extends the model of travel [46] by creating a classification of travel objects and identifying their characteristics in detail [125] (see Chapter 3).

(Q3) – Can Web pages be systematically analysed to extract travel objects?

Harper [53] has proposed a mobility analysis framework, however the focus is on providing a mobility rating which shows the mobility support of the page being analysed. In practice, this framework proved to be inappropriate and insufficiently detailed to be

used to identify travel objects and their roles. Therefore, we present a description and evaluation of a framework, called the travel analysis framework, that systematically analyses Web pages to extract travel objects. This framework provides guidelines to identify these objects and proposes a set of techniques to classify them based on their roles in supporting a traveller's journey. A procedure is also devised to evaluate this framework. The application of this evaluation procedure demonstrates that the analysis process encoded in this framework is systematic and consistent [125] (see Chapter 3).

(Q4) – How can knowledge about these objects be captured in a computationally available form?

In order to demonstrate our hypothesis which claims that transforming Web pages by using identified travel objects can improve travel support, we need to be able to computationally process them. This requires associating identified knowledge with objects on a Web page.

One possible way of associating this information externally is using semantic annotation techniques. Semantic annotation is the process of semantically enriching Web pages by adding metadata[6] so that they are not only available for human consumption but also available for machine processing [51]. However, in order to accomplish this semantic annotation we need to capture knowledge about these objects in a way that enables us to perform the annotations. A widely known way of knowledge capturing is creating "ontologies". An ontology allows us to create machine-interpretable definitions of concepts to create a controlled vocabulary [86]. Therefore, the Web Authoring for Accessibility (WAfA) ontology has been developed to encapsulate knowledge about travel objects in a computationally available form (see Chapter 4).

(Q5) – Do the travel objects provide enough understanding of both structural and navigational properties of Web pages?

Although the classification of travel objects and the travel analysis framework provides an understanding of how travel objects are used or needed to be used in a typical journey, we still lack the understanding of the underlying structure of the Web pages. Travel objects do not provide information about the overall structure and organisation of Web pages, and how these objects are presented. Therefore, this thesis investigates the relationship(s) between how these objects are *presented* and *used*.

[6]**Metadata** is data describing content/functionality.

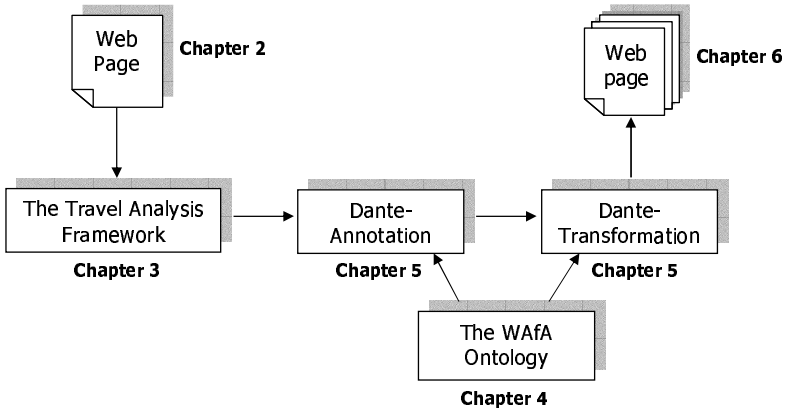


Figure 1.2: The Dante approach and the structure of the thesis.

It is observed that HTML which is the main Web authoring language does not provide enough information about travel objects. It mainly provides a vocabulary that can be used to markup the physical properties of Web pages (e.g., image IMG tag) but not the navigational (e.g., reference point) and structural properties (e.g., logo). Therefore, travel objects are extended by studying existing work on Web design, transcoding, Web accessibility, etc. to identify the terminology that people use to describe the structure of Web pages. Therefore the WAfA ontology encodes knowledge about both structural and navigational properties of Web pages (see Chapter 4).

(Q6) – How can Web pages be annotated with the WAfA ontology?

The annotations with the WAfA ontology are mainly required to perform the transformations of Web pages. This then raises the question "what would be the best way to link the annotation and transformation processes?". The main requirements for this can be specified as (i) the output from the first component will be the input to the other; (ii) loosely coupled components would be beneficial as they can easily be replaced with alternative implementations. Based on these requirements, this thesis proposes a method which is based on a flexible pipeline approach. This approach provides the opportunity that the annotations can arrive in different formats from different sources. We have developed Dante which is an experimental test bed that shows how this pipeline can be developed. Even though in Dante some particular tools are used to annotate and transform Web pages, the infrastructure proposed by this pipeline can still be used to accommodate alternative implementations. Figure 1.2 shows the process encoded in Dante [122, 123, 124].

We surveyed annotation tools and techniques to find out if we can use an existing tool in Dante. Our survey revealed that the existing tools and techniques focus on the annotation of the content rather than the structure of the pages. We also found out that even though these annotation tools can be used to annotate Web pages, none of them addresses the issue of creating relationship(s) between the annotations. Therefore, we used an existing annotation tool and developed another tool to create relationship(s) between the annotations (see Chapter 5).

(Q7) – How can annotations be used to transform Web pages?

This thesis proposes different techniques to transform Web pages to provide alternative views of a page to support a variety of journeys. The proposed transformation techniques are based on the model of travel. These techniques can be considered as intermediate operations that transform pages into a form which can easily be accessed with a screen reader. Our experiments with the proposed transformation techniques also demonstrate that the quality and the quantity of the annotations depend on the transformation techniques to be performed. With these transformation techniques, the WAfA ontology is also evaluated by assessing its competency to satisfy the requirements of these techniques (see Chapter 5).

(Q8) – How can Web pages be evaluated for their travel support?

The transformation techniques encoded in Dante show that performed annotations can be used to transform Web pages into alternative forms. Now we need to ask the question "do the transformed pages provide better travel support?". In order to answer this question we need to first ask "how can we evaluate the travel support provided?".

This thesis introduces an evaluation procedure that is based on the model of travel and aims to compare the travel support provided by the original and transformed pages. This procedure is mainly a comparative task based evaluation which uses three evaluation methods Cooperative evaluation [120], Task Load Index (TLX) [57] and structured interview [38] (see Chapter 6).

(Q9) – Do transformed pages provide better travel support?

The procedure developed is applied in order to demonstrate that the transformed pages provide better travel support. Participants were asked to perform a number of questions on both original and transformed versions of a number of pages. The results

demonstrate that the transformations enhance the experience of visually impaired Web travellers by improving mobility and consequently enhancing the travel support provided. For example, participants found the transformed version to be more organised, less cluttered and more manageable. Along with such positive results, this evaluation also highlighted certain issues that need further investigation. For example, some transformation techniques can abolish some of the reference points with which participants are familiar (see Chapter 6).

1.4 Thesis Outline

The remainder of this thesis is organised as follows, as depicted in Figure 1.2:

Chapter 2 Presents the state of the art in Web accessibility, discusses assistive technologies (e.g., screen readers), Web accessibility guidelines and tools, the Semantic Web and accessibility, and compares audio and visual Web navigation.

Chapter 3 Discusses the notion of travel on the Web in detail, presents the travel analysis framework that encodes techniques to identify and classify travel objects, and explains the evaluation of the framework that demonstrates that this framework is systematic enough to be used as the basis of a tool.

Chapter 4 Introduces the WAfA ontology, describes its design and implementation processes, and presents its three sub-ontologies which are (i) *authoring concepts* to define structural properties of Web pages, (ii) *mobility concepts* to define navigational properties of Web pages and (iii) *contextual concepts* to represent context of a journey.

Chapter 5 Describes the experimental prototype of Dante: Explaining its knowledge-driven annotation pipeline – particularly by discussing the role of the WAfA ontology, and then presents different transformation techniques that are used to transform Web pages.

Chapter 6 Explains the procedure used for the user evaluation to evaluate Dante's approach, describes how the data is used, and presents the results of the evaluation.

Chapter 7 Reviews the work presented and the extent to which the stated objectives have been met. The significance of the major results is summarised, outstanding issues are discussed and directions for future work are suggested.

Chapter 2

Literature Review

This chapter reviews the published literature related to Web accessibility. The aim is to demonstrate why accessing and using Web content is a difficult task for visually impaired users, and what has been done in the Web accessibility field to improve this. Visually impaired people use assistive technologies to access the Web. These technologies work satisfactorily as long as the page is designed well. However, this is not the case for many pages. Accessibility guidelines are proposed to promote Web accessibility but as recent surveys demonstrate [3], few designers follow them. There are also tools to automatically validate or repair Web pages against these guidelines, but since certain aspects of the evaluation process are subjective, the supported automated evaluation is limited. Similarly, there are also tools to transform Web pages into a more accessible form for visually impaired users; however most of these tools lack understanding of visually impaired users' interaction with Web pages and their requirements. In order to address such user requirement issues, audio interaction is compared to visual interaction. Finally, in the Semantic Web vision, assistive technologies can also be considered as agents that need to automatically interpret Web pages. Therefore, the current Web accessibility work in this field is discussed. This chapter concludes by summarising the findings of this review.

2.1 Alternative Web Browsing

Visually impaired users access Web pages by using assistive technologies that produce speech output and support keyboard interaction. These include screen readers and browsers specialised in audio interaction. Different assistive technologies will render different environments for the same page. They all support different functionalities

that are important parts of the overall environment that users interact with.

2.1.1 Screen Readers

These are software applications that vocalise the onscreen data from the top left corner to the bottom right, line by line, one word at a time [87]. They are typically used with conventional Internet browsers such as Internet Explorer (IE)[1]. Formerly, conventional browsers generated the visual rendering and then the screen reader used this to present the page in audio. Although the potential functionalities of these conventional browsers could be accessed, full interaction with the underlying source code (e.g., HTML) was lost. For example, they could not recognise and differentiate HTML elements such as tables, headings or frames. Nowadays, most screen readers provide audio rendering directly from the HTML source code, for example, JAWS5.0[2] uses the Microsoft Active Accessibility (MSAA)[3] technology with IE. While this can be seen as an improvement on traditional screen readers, it is still not a complete solution.

The current generation of screen readers work well as long as the Web page is designed linearly and marked-up properly to represent its logical structure rather than its visual layout. However, nowadays Web pages are complex and nonlinear, a journalistic style of writing has been adopted [82] with the focus on visual interaction [40]. The HTML source code, and hence the Document Object Model (DOM)[4], usually represents the visual layout and the presentation of the page rather than its logical structure [97]. For instance, different typography is usually used to emphasise headings instead of using appropriate HTML elements (H1. . .H6).

Current screen readers also support different functionalities to interact with the logical structure of a Web page. Table 2.1 shows example screen readers and the functionalities provided which are important for supporting travel and mobility within a Web page and therefore between Web pages. These functionalities are grouped into two: movement and list. Movement functionalities enable user to move in a Web page by using specific kinds of objects, for example “table movement” helps users to move from one table (explicitly marked up with the HTML Table tag) to another. List functionalities provide a list of specific kinds of objects in a separate window, for example “list frames” provides a list of headings that are explicitly marked up

[1]See `http://www.microsoft.com/windows/ie`.

[2]See `http://www.hj.com`.

[3]**MSAA** is a technology that provides a standard mechanism for exchanging information between applications and assistive technologies (see `http://www.microsoft.com/enable/`).

[4]See `http://www.w3.org/DOM/`.

Functionalities	Jaws5.0	Windows-Eyes4.5[a]	lookOUT4.0[b]	Hal[c]
Movement functionalities: *enable users to move from one object to another.*				
Character movement	√	√	√	√
Word movement	√	√	√	√
Sentence movement	√	√		
Line movement	√	√		√
Paragraph movement	√	√		
Link movement	√	√	√	√
Heading movement	√	√		
List movement	√	√		
Element movement	√			
Frame movement	√	√		
Form movement	√	√		
Table movement	√	√		
Row movement	√	√		
Column movement	√	√		
Cell movement	√	√		
List functionalities: *list only specific objects in a separate window.*				
List links	√			√
List headings	√			√
List frames	√			√

[a]See `http://www.gwmicro.com/`.
[b]See `http://www.screenreader.co.uk/`.
[c]See `http://www.dolphinuk.co.uk/`.

Table 2.1: Screen readers.

with the HTML heading tags (H1-H6). However, not all screen readers support these functionalities. Moreover, their efficiency and effectiveness are highly dependent on the quality of the underlying markup.

The home page of Amazon[5] is a good example (see Figure 2.1). The page is visually quite well organised: the navigation links are separated from the main content by using sidebars, distinctive colours are used in the header, etc. However with the audio rendering (see Figure 2.2) provided by a screen reader, it is difficult to understand the content, to orientate yourself in the page and to get an overview of the page. Moreover, the page is quite long and has a number of visually distinctive sections, but when a screen reader user requests the "list of headings" in the page (see Table 2.1), the response given by a screen reader is "there are no headings in this page". This is a misleading information but since the page is not marked-up properly, screen readers cannot access the headings. Therefore, even though screen readers support travel and mobility functionalities, pages themselves do not.

[5]See `http://www.amazon.com/`.

Figure 2.1: The home page of Amazon (20/06/2004).

This page has 196 links image map link ref=top_nav_sb_gateway/102-1145037-0464903 image map link ref=cm_wl_topnav_gateway/102-1145037-0464903?type=wishlist image map link ref=topwl...

Figure 2.2: The home page of Amazon in audio (20/06/2004).

Several studies have also highlighted the ineffectiveness of screen readers for navigating and browsing the Web [39]. The main issues can be highlighted as follows:

- Screen readers focus on sensory translation (direct translation to audio) rather than travel and mobility.
- The effectiveness of functionalities supported by screen readers (see Table 2.1) are highly dependent on the design of the page. If the page is not well-structured and marked up, they can provide misleading information.
- They produce superficial information about a page; layout which is often a key for understanding the content of the page, style, font, or nature of information is lost.
- They do not adequately convey the logical structure and semantics of the content, nor do they provide users easy ways to select which parts of a page to listen to.

2.1.2 Specialised Browsers

These are browsers that are specialised in providing audio output. As opposed to screen readers which need to be used with a conventional browser to access the Web,

these browsers can be used as stand alone audio browsers. IBM Home Page Reader (HPR) [8], pwWebSpeak[6], BrookesTalk [128] and Emacspeak [98] are some of the widely known specialised browsers. Compared to others, Emacspeak is not stand-alone, but is a speech output interface to the Emacs text editor[7]. Therefore, Emacspeak will always have the shortcoming of working only under Emacs.

These browsers all use HTML source code to render the page in audio, for example Emacspeak first generates the DOM representation of the page and then renders the page based on that model. These browsers also provide different functionalities specifically to interact with the structure of the page. For example, HPR provides ten reading modes including table navigation mode, frame navigation mode, items reading mode, etc. Although these reading modes can reduce the number of key combinations and simplify the command language, they can also be confusing because the user always needs to maintain a mode.

HPR and pwWebSpeak also address issues concerning orientation and provide functionalities to support it. They both provide functions to get information about the position in the page, but present different information. HPR provides information about the current object such as its physical location (e.g., at 29% of the page). However, for such information to be useful, the user needs to have a good understanding of the overall physical layout of the page. Therefore, there is a danger that the provided information can confuse the readers rather than improve their spatial orientation. pwWebSpeak only provides the element number of the active item, for instance if the user is currently reading the logo and requests the cursor position, then pwWebSpeak gives the position as 'at element 20 over 126'. The given information is at an abstract level and far from enabling the user to understand his(her) position on the page. Moreover, they do not present an abstract view of the logical structure of the page.

Compared to other specialised browsers, BrookesTalk mainly focuses on summarising pages and facilitating Web searching and browsing. BrookesTalk uses Natural Language Processing (NLP) text summarisation and abstraction techniques to facilitate audio browsing [128]. BrookesTalk also provides a summary of the page which includes title, number of words, links, headings, images, extracted keywords and meta keywords. While the summary is crucial for getting an overview of the page, it does not include information about the layout, spatial organisation and visual cues that are also important for travel and mobility.

[6]See `http://www.soundlinks.com/pwgen.htm`.
[7]See `http://www.emacs.org/`.

Besides these well-known audio browsers, there are also other audio browsers where the focused users are not visually impaired. For example, HearSay has a different approach compared to other audio browsers discussed here because it performs structural and semantic analysis of HTML documents and automatically creates Voice Extensible Markup Language (voiceXML)[8] dialogs [96]. HearSay also does not support keyboard interaction, and accepts commands in audio. Although, HearSay combines a number of novel approaches, it is still a prototype and user evaluations demonstrate that it requires a lot of improvement.

As a summary, these browsers render Web pages by directly examining the underlying HTML source code. This used to be an advantage over traditional screen readers, but nowadays most screen readers also render pages with this. Therefore, some of the issues stated for screen readers are also valid for these specialised browsers as well and some specific issues can be summarised as follows:

- They do not provide as many features as conventional Web browsers and tend to be behind the current state of the art.
- They can have problems in handling complex HTML structures such as tables and frames.
- They often do not provide information about the layout, structure and spatial organisation of the page which are crucial for travel and mobility.
- The issue of being "specialised" requires extra effort from the user to learn and use a new tool in order to access the Web.
- Although, some (e.g., HPR) implement most of the 'user agents' guidelines from W3C [49], the focus is still on sensory translation.

2.2 Audio Versus Visual Web Navigation

Compared to visual interaction, audio interaction with Web pages is more difficult. One of the main reasons is that Web pages are primarily designed for visual interaction and the assitive technologies discussed above try to render them in audio. But there are fundamental differences between audio and visual interaction which are discussed here in detail. Knowing the differences is important for designing better Web pages, assistive technologies or intermediate tools for visually impaired users.

[8] See `http://www.w3.org/TR/voicexml20/`.

Persistent versus Transient The visual rendering of Web pages is *persistent*, the screen acts as an external memory, as opposed to audio rendering which is *transient*. In visual interaction, the user can shift the focus of attention rapidly. The monitor is a two-dimensional medium that combines persistence (look at it after as long as you please) with selective updating (interact with a screen without changing the rest of the screen) [85]. Thus the screen acts as a form of external memory and the amount of information which has to be stored in the user's own memory is kept to minimum. However, in auditory rendering, the user is forced to hold the necessary information in memory [93]. Therefore, the number of items presented in audio has to be kept minimum.

Serial versus Parallel Compared to visual rendering, audio is less focused, and more *serial* and linear in nature; the user cannot easily and quickly shift the focus or skip the irrelevant material; accessing could be random as well as serial. Visual rendering is not parallel but sighted users have good control of their sight and can switch focus between areas quickly and easily. The combination of catching attention and shifting focus means that we can process visual information almost in *parallel*. The serial nature of audio scanning provides a drastic contrast to visual scanning. Visual presentation relies on layout for presentation, whereas audio relies on the sequence of presentation [84, 94]. Therefore, in audio linearised material can be read more easily than multi-dimensional material such as tables.

Structure Audio presentation is not as rich as visual presentation. Users cannot pick out important information or create *connections* between separate data [85]. The *structure* of the document (e.g., proximity, grouping of the objects) and the included visual cues (e.g., typography; font size,) play a crucial role in understanding the content of the document [119]. They provide implicit supplementary information about the content so when the same document is rendered in audio, the implicit information encoded in the structure and visual presentation of the page is lost. Therefore, it is crucial to encode the same information in the underlying source code of the document so that it can be presented in alternative ways in audio.

Active versus Passive In audio presentation, the users do not *actively* search for a target, but rather they must wait and try to ambush it. The lack of flexible control can impose a greater load on working memory as unwanted interpolated material intervenes between candidate selections and the users' final choice. The navigation functionalities also tend to be limited in the auditory presentations so the limited functionalities have negative implications; users become *passive* listeners rather than *active*

readers [103]. For example, it is demonstrated that alphabetical lists are not better than random lists, as users cannot easily jump to a random point or skip some items in the list whereas sighted users can. Therefore, it is important that the user has control over the information flow [94].

Random versus Serial In visual presentation, the content of the page can be accessed *randomly* as well as *serially*, whereas in audio rendering content tends to be accessed serially, typically from top to bottom [84, 94]. Therefore, it is important that significant information is located at the top so that it is accessed first in the audio presentation. Whereas in visual presentation, the visual distinctiveness of the significant information is more important than its location.

Amount of Information Finally, the other two important issues are the *amount* of information and the *time* taken to process the information. Spoken options will take longer to process than the written options. The amount of information that can be accessed at once in audio rendering is much less compared to visual rendering due to the bandwidth difference between the visual and auditory channel [94]. Therefore, the amount of information presented in audio must be carefully chosen.

These differences are important for understanding how visually impaired users access Web pages and what needs to be considered in order to support audio navigation and movement in Web pages. Particularly, these differences are addressed when Dante transforms Web pages to provide better audio interaction for visually impaired users.

2.3 Web Accessibility Guidelines

The difficulties that visually impaired people have accessing the Web are not only because of technological advances, but also because of the inappropriately designed Web pages. The World Wide Web Consortium (W3C) recognises these difficulties and provides guidelines to promote accessibility. There are also other organisations that provide accessibility guidelines, including The Royal National Institute for the Blind (RNIB) and The American Foundation for the Blind (AFB) (see Table 2.2).

While these organisations highlight some of the important accessibility issues, the W3C accessibility guidelines are more complete and cover key points of all the others. Moreover, the W3C not only provides guidelines for creating accessible content [27], but also provides guidelines for user agents[9] and authoring tools[10].

[9] An application that retrieves and renders Web content [49].
[10] An application that is used for creating Web content [110].

Organisation & Guidelines	**Website**
WAI Guidelines	`http://www.w3.org/WAI/`
Section 508 Guidelines	`http://www.section508.gov/`
RNIB Guidelines	`http://www.rnib.org.uk/`
AFB Guidelines	`http://www.afb.org/`
Macromedia and Accessibility	`http://www.macromedia.com/`
DRC Report	`http://www.drc-gb.org/`
Nielsen Norman Group Report	`http://www.nngroup.com/`
Dive into Accessibility	`http://www.diveintoaccessibility.org/`
IBM guidelines	`http://www-306.ibm.com/able/guidelines/`
Nova Report	`http://www.cerlim.ac.uk/projects/`

Table 2.2: Web accessibility guidelines.

No.	**Guideline**
Theme 1: Ensuring graceful transformation	
1.	Provide equivalent alternatives to auditory and visual content.
2.	Do not rely on colour alone.
3.	Use markup and style sheets and do so properly.
4.	Clarify natural language usage.
5.	Create tables that transform gracefully.
6.	Ensure that pages featuring new technologies transform gracefully.
7.	Ensure user control of time-sensitive content changes.
8.	Ensure direct accessibility of embedded user interfaces.
9.	Design for device-independence.
10.	Use interim solutions.
11.	Use W3C technologies and guidelines.
Theme 2: Making content understandable and navigable	
12.	Provide context and orientation information.
13.	Provide clear navigation mechanisms.
14.	Ensure that documents are clear and simple.

Table 2.3: Summary of the Web Content Accessibility Guidelines (WCAG 1.0).

The Web Content Accessibility Guidelines 1.0 (WCAG 1.0) describe how to make accessible Web content and Web sites [27]. They are presented in two themes: graceful transformation (of content, structure and presentation), and making content understandable and navigable (see Table 2.3). The specifications provide 14 guidelines, but unfortunately only three of them are in the second theme; the rest such as creating tables that transform gracefully, are oriented to support sensory translation of text content to audio [46]. The Nielson Norman Group has also published guidelines to assist designers to create accessible and usable Web pages [32]. Although these guidelines are based on a series of usability tests of several different Web sites, the guidelines themselves are not different from others. Lately, the Web Accessibility Initiative (WAI) which is a W3C group is working on a new version of WCAG. However, this version has yet to be completed and published[11].

As the DRC report concludes, although the compliance with WCAG 1.0 is necessary, it is not a sufficient condition for ensuring that sites are practically accessible and usable by disabled people [3]. The DRC report also provides a number of recommendations to improve the navigation and orientation issues addressed in WCAG1.0.

The User Agent Accessibility Guidelines 1.0 (UAAG 1.0) describe how to make browsers and media players accessible [49]. The specification emphasises the importance of accessibility of the user interface, enabling the user to have access to the content and helping the user to orientate. Similarly, the Authoring Tool Accessibility Guidelines 1.0 (ATAG 1.0) provide key issues to assist in designing authoring tools that produce accessible Web content and assist in creating an accessible authoring interface [110]. Most of the guidelines in this specification focus on the creation of standard and accessible markup, but they pay little attention to how authoring tools can assist Web designers to create understandable and navigable Web pages (including how to include different orientation mechanisms).

Although the overall vision of these guidelines is good, many designers rarely follow them. According to the DRC report [3], most of the Web sites in their evaluation (81%) fail to satisfy the most basic WCAG 1.0 categories. Designers usually view these guidelines as irrelevant, too restrictive or too time-consuming to implement. Moreover, as the DRC report [3] highlights designers have an inadequate understanding of the needs of disabled users and how to create accessible websites. For example, not many people consider the differences between audio and visual navigation of Web

[11] See `http://www.w3.org/TR/wcag2-req/`.

pages. Some studies also show that the application of the guidelines is subject to interpretation; two designers applying the same set of guidelines to same set of pages generate different results [63]. As a conclusion, although the guidelines are useful, they are only part of the overall process of supporting Web accessibility.

2.4 Tools to Support Web Accessibility

Numerous tools are available to help either Web developers to build better accessible pages or end-users to modify pages to meet their needs [12]. They differ in several ways, including functionalities (e.g. testing, fixing) and method of use (online service, desktop application integrated in authoring tools). They are grouped into three, depending on the provided functionalities and supported user groups (e.g., end users or designers): Evaluation, Repair, and Transformation tools.

2.4.1 Evaluation Tools

Evaluation tools [13] analyse pages or sites against accessibility guidelines (see Table 2.2) and return a report or a rating. They are important for Web accessibility as they provide a medium for designers or authors to validate their pages against published guidelines without actually reading and manually applying them [87]. While these tools encourage markup that conforms to the specifications and guidelines, no one except the Web page designer can really enforce it.

We will review some of the tools under two categories: accessibility evaluation and HTML validation tools (see Table 2.4 [14]) [64]. The first validates pages against guidelines and the second validates pages against the HTML specifications. Some recent studies have concentrated on formally representing guidelines and checking pages against these formal representations [113]. Although this approach can overcome the subjectivity aspect of the guidelines, designers cannot be expected to represent the guidelines in a formal language that is crucial for this approach.

[12] See `http://www.w3.org/WAI/ER/existingtools.html` for a list of available tools.
[13] They are also referred as validation tools.
[14] See `http://www.Webaim.org/techniques/articles/freetools/` for comparison of some of these tools.

Title	Developer	Website
Accessibility Evaluation Tools		
Bobby	Watchfire	http://bobby.watchfire.com/
Lift	Nielsen Norman Group	http://www.usablenet.com/
AccessEnable	RetroAccess	http://www.retroaccess.com/
AccVerify	HiSoftware	http://www.hisoftware.com/access/
Cynthia Says	ICDRI and HiSoftware	http://www.cynthiasays.com/
Wave	WebAIM	http://www.wave.webaim.org/
PageScreamer	Chruncy Technologies	http://www.crunchy.com/tools/
Page (site) Valet	Webthing	http://valet.webthing.com/
Web Access Toolbar	NILS	http://www.nils.org.au/
AccMonitor online	Hi-Software	http://www.hisoftware.com/
Ask Alice	SSB Technologies	http://askalice.ssbtechnologies.com/
Coast	Coast Software	http://www.coast.com/
WeXACT	Watchfire	http://www.webxact.com/
TAW	Sidar in Spanish	http://www.tawdis.net/
Torquemada	WebxTutti	http://www.Webxtutti.it/testa_en.htm
Dr Watson	Addy	http://watson.addy.com/
HTML Validation Tools		
Dr HTML	Imagiware	http://www2.imagiware.com/RxHTML/
W3C CSS-Validator	W3C	http://validator.w3.org/
W3C CSS-Validator	W3C	http://jigsaw.w3.org/css-validator/
WDG HTML Validator	WDG	http://www.htmlhelp.com/tools
Weblint	Weblint	http://www.weblint.org/

Table 2.4: Evaluation tools.

Accessibility Evaluation Tools

Bobby and Lift are the most widely used accessibility evaluation tools (Table 2.4). Bobby evaluates entire Web sites for accessibility and provides a report for each page. Pages are validated against WCAG 1.0 [27] as well as Section 508 guidelines from the Access Board of the U.S. Federal Government[15]. The returned reports tend to be large and mostly consist of recommendations for manual checks. In order to fully understand the reports, one needs to have a deep understanding of the relevant guidelines (see Table 2.3). Three types of information are returned: (i) The list of automatically detected accessibility errors (e.g., missing ALT text for images); (ii) The errors that cannot be fully automatically checked and need further manual examination (e.g., making sure that information is not only encoded in colours); (iii) The accessibility checks that require manual examination (e.g., checking that all link phrases make sense when read out of context). Thus the effectiveness of Bobby is highly dependent on the developer's ability to understand and interpret the report. Moreover, similar to most of the other

[15]See http://www.section508.gov/.

automated evaluation tools, it cannot properly validate dynamic pages [105].

LIFT Online is a tool for testing Web pages against Section 508 and WCAG 1.0. It reviews the site based on some rules derived from these guidelines. It provides several features, for instance, rule prioritisation which allows users to prioritise or disable rules used in a particular test. Checking that keywords are provided, images include ALT text and image map links have textual version, are some examples for the rules applied during a test. Although LIFT provides a report which is easier to understand compared to the Bobby's report, it does not provide the depth of the accessibility review that Bobby provides.

There are also tools that simulate how a page is accessed by a screen reader user, for example aDesigner [108]. aDesigner focuses on evaluating usability of pages as opposed to validating them against guidelines. The usability issues addressed include speed to reach target content, the effectiveness of skip-navigation links[16], etc. The analysis of these issues are then visualised to guide designers when reformulating their pages. Even though aDesigner supports a more sophisticated approach compared to others, designers still need to learn and use a separate tool to repair their pages.

HTML Validation Tools

These are tools to validate source code of Web pages against specifications [17]. The two most widely used tools are the W3C HTML Validation and Doctor HTML.

The W3C HTML Validation service was introduced to validate Web pages against one of the HTML specifications such as XHTML[18]. It also provides several reporting characteristics that can be specified including showing the parsed DOM tree and the outline of the current page.

Doctor HTML is a commercially available tool which can be used to analyse either a single Web page or an entire Web site. It provides a variety of tests including a document structure test which looks for unclosed HTML tags, and image syntax which checks image commands for HEIGHT, WIDTH, ALT tags, etc. Some checks are quite important for accessibility, for instance, checking whether ALT tags exists.

[16] A link that is used within a page for skipping over groups of mostly repetitive navigation links or constructs to get to the main content of the page.

[17] See `http://www.w3.org/MarkUp/`.

[18] The Extensible HyperText Markup Language, see `http://www.w3.org/TR/xhtml1/`.

Title	Developer	Website
Accessibility Repair Tools		
AccRepair	Hi-Software	`http://www.hisoftware.com/`
A-Prompt	SNOW	`http://aprompt.snow.utoronto.ca/`
InFocus	SSB Technologies	`http://www.ssbtechnologies.com/`
LIFT for Dreamweaver	Nielsen Norman Group	`http://www.usablenet.com/`
LIFT for FrontPage	Nielsen Norman Group	`http://www.usablenet.com/`
PageScreamer	Crunchy	`http://www.crunchy.com/tools/`
HTML Repair Tools		
Tidy	Dave Raggett	`tidy.sourceforge.net/`

Table 2.5: Repair tools.

Summary and Conclusions

The DRC report [3] states "the use of automated tools is only the first steps towards accessibility". WCAG also emphasises that such tools "cannot identify all accessibility issues". While the automated evaluation tools can be a quick reference to designers to evaluate their pages, they have many limitations:

1. They cannot evaluate some of the important elements including the layout of the information, the navigation structure, etc.

2. They cannot check pages against all the guidelines or checkpoints as some of them require human judgement, in this case they can only give 'warnings'. For example, they can check the existence of ALT text, but not the meaning or significance of the provided ALT text. Therefore, these tools alone cannot verify effective compliance to guidelines [3].

3. They cannot evaluate the appearance or content of graphics.

4. They can confirm that a page is compliant to guidelines but cannot guarantee that it is easy to use [108]. For example, a text-only page is likely to get a high ranking without actually considering the quality of the content [70].

2.4.2 Repair Tools

Repair tools also evaluate pages against accessibility guidelines or HTML specifications, but compared to evaluation tools, they try to fix the identified problems. Table 2.5 summarises a number of repair tools and here we discuss two examples: A-prompt and Tidy.

Accessibility Repair Tools A-Prompt includes a validator that evaluates accessibility of pages against WCAG 1.0. If A-prompt detects a problem it displays necessary dialogs, and guides the user to fix the problem. For example, it displays a dialog box if there is no title for a frame. It also allows a user to set the level of accessibility check against the priorities provided by WCAG 1.0 and Section 508 guidelines.

HTML Repair Tools Tidy identifies and repairs HTML errors in pages. If it cannot fix the problems automatically, it prints warnings for those problems. It also offers advice on accessibility problems, such as suggesting the addition of a summary attribute to table elements.

Similar to evaluation tools, repair tools also require human judgement and cannot automate the entire process. They also have the same limitations that are specified for evaluation tools.

2.4.3 Transformation Tools

Fundamentally, the evaluation and repair tools focus on assisting the authors to modify or correct their pages. On the other hand, transformation tools focus on assisting Web users. By using different techniques, these tools mainly transform pages into another form to meet users' needs [19]. There has been a number of studies on page transformations. Depending on the approach, we have grouped the available studies into six which are discussed in the following sections.

Generating Text-Only Version

It is a common approach to maintain a text-only version of the original site to support accessibility. The three most widely known tools are: Betsie [81], Textualise [28] and the LIFT Text Transcoder [111] (see Table 2.6). Betsie is designed to create an automatic text only version of British Broadcasting Corporation's (BBC) Website. Textualise and the LIFT transcoders are commercially available. All three are server-side applications and designed for supporting mainly Web site designers so that they can easily create text-only versions of their Web sites. Textualise and LIFT transcoder also require authors to provide some structural hints in order to generate text-only versions. Betsie has the shortcoming that it only works with the BBC's Website.

As Lawton [109, Chapter 1] states "the issue of text only versions crosses into the

[19] This process is also referred as **Transcoding** (see `http://www.w3.org/TR/annot/`).

idea of separate versus inclusive design". When Web pages started to become a visual communication media, text-only versions were considered to be a good solution for accessibility. Assistive technologies could not handle complex structures and with text-only versions, designers were not constrained anymore. However, assistive technologies developed remarkably. Additionally, with recent design technologies (e.g., Cascading Style Sheets (CSS) [20], Scalable Vector Graphics (SVG) [21], etc.), pages can be designed that are not only visually appealing, complex and dynamic but are also accessible. Therefore, there should not be a need for alternative versions to support accessibility any more.

The shortcomings of this approach can be summarised as follows:

- Text-only versions are not updated as frequently as the primary versions, so they tend to be old and lack information or facilities compared to the original page.
- Keeping two versions synchronised is not an easy task, even with the tools discussed above. Duplicate effort could seriously tax human and machine resources.
- Primary versions often lack even the most basic accessibility requirements.

[20] See `http://www.w3.org/Style/CSS/`.
[21] See `http://www.w3.org/TR/SVG/`.

Tool	Location	Preferences	Annotations	Users	Heuristics	Automatic	Layout	Role	Importance	Chunking
Generating Text-Only Version										
BETSIE [81]	Server	No	No	Designers	No	?	No	No	No	No
Textualize [28]	Server	Yes	Yes	Designers	No	Yes	No	No	No	No
LIFT [111]	Server	Yes	Yes	Designers	No	Yes	No	No	No	No
Transforming based on user's preferences										
Gateway [20]	Server	Yes	No	VI	No	Yes	No	No	No	No
Transforming based on rules and heuristics										
WAB [69]	Proxy	No	No	VI	No	Yes	No	No	No	No
[99]	Client	Yes	No	VI	No	Yes	No	No	No	No
Transforming based on external annotations										
[9]	Proxy	No	Yes	VI	No	?	Yes	Yes	Yes	No
[59]	Proxy	No	Yes	SSD	?	?	Yes	Yes	Yes	Yes
[60]	Proxy	No	Yes	SSD	?	?	No	No	No	No
SWAP [101]	Server	No	Yes	CDU	No	?	Yes	Yes	No	No
Transforming for Small Screen Devices										
Digestor [15]	Proxy	Yes	No	SSD	No	Yes	No	No	No	No
PowerBrowser [22]	Proxy	No	No	SSD	Yes	Yes	Yes	No	No	No
[24]	Server	Yes	No	SSD	No	Yes	Yes	Yes	No	Yes
[25]	Server	No	No	SSD	Yes	Yes	Yes	Yes	No	Yes
[127]	Server	No	No	SSD	No	Yes	Yes	No	Yes	Yes
Filtering Tools										
Muffin [18]	Proxy	Yes	No	All	No	Yes	No	No	No	No
WebCleaner [4]	Proxy	Yes	No	All	No	Yes	No	No	No	No

VI– Visually impaired users.
SSD- Small screen device users.
CDU- Cognitively disabled users.
?– Is not fully supported.

Table 2.6: Transformation tools.

Transforming Based on User's Preferences

There are some transformation tools that adapt pages or parts of pages based on the user's preferences. This is an advantage over other tools as users can specify how they want to *see* or read the page, and can easily direct the transformation process. The Web Access Gateway [20] is an example of such a tool whereas Tablin is an example of tools that transform complex structures, in this case tables [1] (see Table 2.6). Gateway provides an HTML form into which user enters the Uniform Resource Locator (URL) of the page to be processed along with transformation preferences. These preferences include options regarding frames, tables, style and images. Fundamentally, Gateway tries to serialise most of the HTML elements (e.g., frames) by inserting one after the other, tables by linearising row by row, images by replacing the graphic with the provided ALT text, and so on. Although Gateway performs serialisation on the server side, similar serialisation techniques have already been implemented by currently available assistive technologies on the client side.

Tablin transforms Tables. Tables are complex and important structures that are commonly used for laying out Web pages [126]. In that sense Tablin can also be considered as a page transformation tool. Tablin has an online form into which users can specify their preferences then it linearises tables according to these specified preferences. This obviously requires prior knowledge regarding tables, but if the users already know the structure, then the table can be read without linearising it. Moreover, this linearisation eliminates the multi-dimensional nature of tables which supports navigation and browsing [121].

Transforming Based on Heuristics

As opposed to tools that transform pages according to users' preferences, there are tools that perform transformations based on a set of predefined heuristics or settings. WAB [69], and [99] are examples. WAB is a proxy server that modifies pages according to four heuristics which are: (i) adding the word 'link' before each link in the page; (ii) adding a title list and a link list at the end of the page; (iii) adding two links referring to added lists at the top of the page; (iv) adding keywords such as 'Checkbox' or 'Button' before each of those form elements. Although these modifications were useful when WAB was proposed, they are actually not needed and useful anymore. The new generation assistive technologies already support these functionalities. However, WAB is one of the early works on transformation and demonstrates the applicability

of the transformations.

Richards *et al.* [99] propose an extension to Internet Explorer targeting for an older population with low vision. The supported transformations include line spacing, changing colours, enlarging banner text, etc. All these transformations can be useful for screen magnification users but not for screen reader users. Moreover, most of these adaptations are already supported by conventional browsers.

Transforming Based on External Annotations

In a transformation context, annotation means computer understandable data associated to human understandable data [12]. There are some tools that perform transformations by using such kinds of annotations, for example, [9], [60] and [59]. Asakawa *et al.* [9] present an approach to transform pages for visually impaired people, whereas, [59, 60] transform pages for small screen device users. Even though the requirements of these two user groups are related, there are still major differences; one uses visual interaction, the other is audio. Some other approaches also exist that focus on cognitively disabled people [22] [101]. This user group also have different requirements, for instance a page is more accessible if a simple language or illustrations (or symbolic language) are used.

Asakawa *et al.* [9] classify annotations into two: structural and commentary. Structural annotations are used to identify visually fragmented groupings, their roles and importance. Commentary annotations are used to provide description regarding these identified groupings. The notion of visually fragmented groupings is not defined properly, the only information given is that, the page is fragmented into groupings by using various types of visual effects such as layout tables. They propose nine structural roles: proper content, updated index, heading, footer, delete, advertisement, norole and layout table. Seeman [101] also proposes a similar vocabulary. These roles form a small, limited set of structural vocabulary which most of the time is not rich enough to completely annotate the structural and navigational properties of a page. The WAfA ontology covers a richer vocabulary that includes these as well. Asakawa *et al.* [9] transform pages by presenting groupings in descending order of their importance. If a role is not assigned to a group, the importance will be a default value which means that groups will be placed in the middle of the page. Therefore, the transformation may result in a more inaccessible page than before. Seeman [101] uses annotations to provide a simplified text for the cognitively disabled users with additional pictures and

[22] People who have problems in reading and comprehending text [27].

schematic diagrams.

Hori *et al.* [59] also have a similar approach to [9], but uses an extended vocabulary for annotations. The extended vocabulary includes information about the client capability, fidelity, alternatives, etc. These are important requirements for small screen devices but are not needed in the context of visually impaired users.

Hori *et al.* [60] propose a very simple vocabulary for the annotations which include only 'keep' (content should be preserved) and 'remove' statements. Although this can be a cost-effective approach, keeping and removing content is contextual; it mainly depends on what the user wants to do.

All these approaches [9, 59, 60] use XML Path Language (XPath [23]) to refer to a part of a page. Even though XPaths can allow them to talk about parts of a page, the XPaths can easily become obsolete by site redesign, although, Abe *et al.* [5] suggest that some form of XPaths are more robust than others. Moreover, since creating external annotations can be an expensive process, Takagi *et al.* [107] present an algorithm to overcome this. It tries to use existing annotations for a page on other pages on that site. Although it is a promising approach, pages in that site need to be structured similarly and different annotations are still required for different sites.

Transforming for Small Screen Devices

Numerous approaches to adapting Web pages for small screen devices have been proposed. These approaches mainly involve developing systems that can take an existing Web page designed for the desktop, along with the characteristics of the target display device, and re-author the page through a series of transformations so that it can be appropriately displayed on the device. They can also be useful and important for visually impaired users because small screen device users also have a limited view of pages. However, in general the requirements of these two user groups are different. Adaptation systems for small screen devices need to consider the cost and availability of the bandwidth, screen sizes, memory capability of devices, quality of images, etc., none of these requirements are important for visually impaired users.

Different techniques are proposed to adapt pages for small screen devices which are discussed as follows:

Image Customisation The most simplistic approach is to reduce image and font sizes so that the page can easily be displayed on such devices [15, 102, 116]. These customisations are irrelevant to visually impaired users.

[23] See `http://www.w3.org/TR/xpath`.

Structural Customisation Some approaches use structural HTML elements such as headings (H1-H6) to customise pages for these devices. [15, 116] partition a page into several pages and use headings (H1-H6) to create a table of contents with links to these partitions. Indeed some screen readers already support a function to access only headings in the page (see Table 2.1). [15, 21, 116] summarise pages by using the first sentence of each paragraph (e.g., P tag) in a page. However, it is well known that pages are not appropriately marked up and designers usually use visual elements to represent headings and paragraphs.

Link Customisation Some summarisation techniques are introduced that focus on the links in the page. Buyukkokten *et al.* [22] summarise pages by presenting only links. The technique removes repeated links and presents them in different orders such as in alphabetical order or according to ranks generated by their algorithm. It also tries to present them hierarchically depending on the structure of the page. The simplified version of this technique is commonly supported by most screen readers and link summarisation features of screen readers can be extended with this technique.

Chunking There are some studies on automatically analysing HTML pages to identify coherent chunks of information (visual groupings) in the page where they are only visually accessible [21, 24, 25, 127]. Although these techniques are important for automated processing, they mainly focus on certain types of pages with certain sets of rules. They are not generic enough to apply to different types of pages; they work well with small well-structured Web pages but generate almost unusable results for complex Web pages. The identified chunks are then used in different ways: (i) [21, 25, 127] partition pages based on the identified chunks and the target screen size; (ii) Yin *et al.* [127] propose an algorithm to assign ranks to these chunks and then reorder and display them depending on these ranks; (iii) Buyukkokten *et al.* [21] propose different techniques to summarise identified chunks in order to provide a summary of the page which includes displaying only first few sentences, extracting important keywords from these chunks and displaying most important sentence in the chunks; (iv) Chen *et al.* [25] display a thumbnail representation of the identified chunks and then the user can zoom in to see the complete chunk. Apart from this, all other techniques can also be useful for visually impaired users, but need to be reconsidered as they all focus on the limitations of these devices, for example, the amount of scrolling, identifying chunks that can only fit into a certain size of page, etc. Furthermore, even though they simplify pages,

the focus is still on visual interaction. For visually impaired users the requirements of audio interaction need to be considered. Finally, none tries to understand the overall structure of the page and different components that have different roles in supporting user journeys. Although, Chen *et al.* [24] propose a function based object model, it still does not provide a deep understanding of the page structure and how users use these objects in a typical journey.

Although these approaches present different techniques which can be quite useful for visually impaired users, the requirements of their focused user group is different than our target user group [65]. Therefore, Dante adapts some of these techniques by modifying them in order to provide better support for visually impaired users.

Filtering Web pages

As opposed to transformation tools, these do not focus on restructuring pages. Their aim is to remove unnecessary information from the page. Muffin [18] and WebCleaner [4] are widely known filtering tools. These tools are important as they can easily remove elements from the page that cannot be handled properly by screen readers such as Javascript, cookies, animations, advertisements, Java applets, etc. However, as it is the case with text only pages, removing these elements does not guarantee an accessible page nor generate a page that is easy to navigate and move around as this can also depend on the page structure and layout.

Modifying Alternative Text (ALT)

Images are important components of visual presentation of Web pages. However, they are inaccessible to visually impaired users unless an alternative text (ALT) is provided for them. Since designers usually design pages with visual communication in mind, they tend to ignore adding alternative text to images. To overcome this problem, a number of techniques are proposed including ALTifier [114] and ALT-server [34]. Altifier encodes a set of heuristics to guess missing ALT text for images. Even though encoding heuristics can be useful, it is still very difficult to guess in which context the image is used or why it is used (e.g., for decoration or information), etc. For example, the same image can be used in different contexts which require different ALT text. Even if all the images have ALT texts, this does not mean that the page is accessible. Ensuring that images have ALT texts is only a partial solution.

Summary

The extensive work reviewed in this section demonstrates the importance of transformation tools. It would be too idealistic to believe that designers will design a page that will meet the requirements of different user groups. However, if designers create a page that is properly marked up to represent its logical structure, these transformation tools can modify pages to meet the user requirements. In the Web accessibility field, these tools can be considered as intermediary tools that adapt pages in a way that they can be easily accessed by assistive technologies. The main findings of the review can be summarised as follows:

- Different studies demonstrate the effectiveness of transformation tools.
- Generating a text-only version without considering structural and navigational properties of pages may not be the right solution for supporting Web accessibility.
- Filtering tools can be useful but removing unnecessary items from the page does not guarantee an accessible page.
- Focusing on certain accessibility issues such as missing ALT text is important but it is just a partial solution.
- The transformation work for small screen devices can be useful for visually impaired users as well but because of the different requirements they cannot be applied as they are. They need to be changed and adapted for visually impaired users.
- None of the transformation approaches tries to have a deep understanding of the overall structure of Web pages, particularly the components that are used by users to move around pages.
- Even though external annotations are expensive to create and maintain, they proved to be effective for transformations.

2.4.4 Summary and Conclusions

We have reviewed evaluation, repair and transformation tools. Evaluation and repair tools can help designers to ensure the accessibility of their pages and transformation

tools enable end-users to modify pages in different ways. The main conclusions can be summarised as follows:

- Automated evaluation and repair tools cannot identify and fix all accessibility issues but they can be a quick reference to designers to validate their pages against accessibility guidelines.

- Transformation tools are promising for adapting pages to meet user needs but none of the available transformation tools try to understand how visually impaired users move around pages and what kind of objects they use or need to use in order to easily move around. Similarly, available transformation tools do not focus on understanding the logical structure of the page which is crucial for supporting travel and mobility around the pages. They also ignore the fundamental differences between audio and visual navigation.

2.5 Semantic Web and Accessibility

The Semantic Web, as articulated by Tim Berners-Lee *et al.* [14], is an extension of the current Web in which the resources are well defined and explicitly interpretable not only by humans, but also by automated agents. The automation of tasks such as precise search, resource discovery, information filtering, etc. depends on migration of data from *machine-readable* to *machine-understandable* form. One possible way of achieving this is to associate metadata (i.e., data describing content/functionality) to Web resources. The W3C proposes a formal data model called the Resource Description Framework (RDF)[24] for describing metadata. One mechanism for associating such metadata is annotation [11]. However, automated tools cannot benefit from these annotations unless they agree on the semantics (meaning) of the used language. In order to do so, ontologies have been proposed which include machine-interpretable definitions of basic concepts in a specific domain and relations among them [86]. They are also known as controlled vocabularies and defined as "a shared specification of a conceptualisation" [48]. Recently, an ontology language, OWL, has been standardised [76].

If we look at the Semantic Web from a Web accessibility point of view, assistive technologies such as screen readers can also be considered as agents that need to understand and interpret Web pages. When visually impaired users read a Web page,

[24]**RDF** is a graph-based model for describing Internet resources such as Web pages in triples (see `http://www.w3.org/RDF/`).

they rely on the screen reader's interpretation of the page. Therefore, screen readers need to have a good understanding of the page, particularly its structure, in order to appropriately present and support movement around the pages.

In the context of the Semantic Web, few studies have been undertaken on Web accessibility. As part of the W3C Semantic Web activity[25], the Evaluation and Repair Language (EARL) is proposed [26]. EARL is an RDF based framework for recording automatic or manual evaluation results. This language provides a means for expressing accessibility evaluation results in a machine understandable form. Traditionally, icons are used to demonstrate that the page conforms to a certain set of guidelines. Although EARL is quite important to overcome the problem of *icons for claims*, it is still the authors' responsibility to fix all the identified problems.

The potential of Semantic Web technologies such as ontologies and annotation techniques have not been exploited enough yet in the Web accessibility field. These technologies have already proved their usefulness in a range of application domains (e.g., [14]), which suggests that they can also be useful for Web accessibility.

2.6 Summary and Conclusions

This chapter has reviewed the "Web accessibility" field. The main conclusions of this review are summarised here. Visually impaired people have difficulties accessing the Web, either because of the inappropriately designed Web pages or the insufficiency of currently available technologies. This lack of accessibility leads to poor mobility and travel support for visually impaired users.

Screen readers or Specialised browsers are widely used by visually impaired users to access the Web. For these access technologies to work properly, Web pages must be appropriately designed and must be encoded in valid HTML that conforms to its DTD (Document Type Definition) and various accessibility guidelines. The W3C Web Accessibility Initiative and others recognise these difficulties and provide guidelines to promote accessibility on the Web. Unfortunately, not many designers follow these guidelines. Additionally, access technologies such as screen readers have focused on supporting the sensory translation of visual content to audio rather than deeply affecting travel and mobility on the Web.

While the approach of evaluation and repair tools is sensible, they cannot automatically evaluate and repair some of the important issues for supporting Web accessibility

[25] See `http://www.w3.org/2001/sw/`.

(e.g., the layout of the information, the navigation structure, etc.). Therefore, the provided solutions are limited and partial.

Different techniques are proposed for transforming pages to improve accessibility support. Although these are promising, none of them provides an understanding of how visually impaired users navigate and move around pages. For example, none of them considers the difference between audio and visual navigation which is important for supporting nonvisual interaction. Moreover, they do not have a deep understanding of the structures of Web pages and how users use or need to use such logical structure to easily navigate around the pages. Without such understandings, the modifications cannot be as effective as they should be. The following chapter presents our approach to provide an understanding of how visually impaired users navigate around pages and what kinds of objects they use or need to use to complete their journeys.

The Semantic Web is a promising field which is considered as an extension of the current Web. Web accessibility has not been explored in this field which needs to be considered from the beginning of this extension rather than being an afterthought. Therefore, in the following chapters, a possible approach will be presented that uses Semantic Web technologies to support Web accessibility.

Chapter 3

The Travel Analysis Framework

Although the existing work to promote Web accessibility is extensive, it lacks a systematic understanding of how visually impaired people move around pages; what kinds of objects they need or use; and what kind of spatial environment is easier for them to journey through. In order to provide such understanding and improve the accessibility of the Web for visually impaired users, the notion of travel was introduced [53].

Users make journeys through the Web. Web travel encompasses the tasks of orientation and navigation, the environment and the purpose of the journey. The ease of travel, its mobility, varies from page to page and site to site. Knowledge about travel and mobility in the physical world gives a context for travel and mobility on the Web. A model of travel was proposed to transfer and adapt real world findings to the Web [46]. For visually impaired users, in particular, mobility on the Web is reduced; the objects that support travel are inaccessible or missing altogether. These objects need to be presented in a way that they can fulfill their intended or implicit roles. The travel analysis framework provides techniques to identify and classify these objects [125]. However in order to capture such techniques in a framework, these objects and their characteristics need to be identified in detail; which has meant extending the proposed model of travel. The evaluation carried out has shown that this framework supports a systematic and consistent method for assessing travel upon the Web. This travel analysis framework provides the basis of Dante, for the support of travel upon the Web.

3.1 Travel and Mobility on the Web

The notion of travel and mobility on the Web is introduced to improve the accessibility of Web pages for visually impaired and other users by drawing an analogy between

travel in the virtual and the physical world [53]. Drawing parallels proved to be effective for improving interface usability of a virtual world [35]. In the context of travel, real world mobility studies can provide insights of how visually impaired people travel. For example, real world travel and mobility studies suggest that there are differences between how sighted and visually impaired people travel. Considering these differences can help to fully support visually impaired users' journeys. Key differences which are called mobility principles [53] include:

Information flow Sighted people usually ignore details when they navigate in a physical environment. They usually do not inspect each obstruction. However, as it is not easy to assimilate complex information by non-visual means, they usually use complex information less often than simple information.

Granularity Compared to sighted people, visually impaired people have limited preview of coming objects. Therefore the use of some type of preview device is important. Additionally, visually impaired travellers orient themselves more frequently than sighted travellers do; they tend to break their route into a greater number of more complex stages. They use explicitly more environmental cues such as landmarks, paths, borders and boundaries.

Egocentricity Many visually impaired people tend to think of the real world in *egocentric* manner; the traveller associates journey, route and survey knowledge to themselves and not to the environment.

External memory and mental maps Visually impaired travellers have a tendency to create exact maps, divided into smaller and more manageable steps and with many more landmark points. External memory is not efficiently used. Visually impaired users rely less on external memory sources as they are usually not in an appropriate form.

Regularity and familiarity of the environment Visually impaired people would normally travel unassisted in areas that were familiar to them. This can be considered as the issue of predictability.

Spatial awareness Previous visual experience may affect spatial awareness. Many adventitiously visually impaired (had previous visual experience) people, are better at decoding spatial information than many congenitally visually impaired people. Spatial information is technically difficult to convey and rely on, even if visually impaired users can assimilate it.

In order to be able to adapt and transfer the findings and observations of real world studies to the virtual Web world, such as the differences explained above, they have to be captured in a model [46]. Therefore, a model of travel has been proposed. This model consists of three components: objects (e.g., landmarks), actions (e.g., landmark detection) and instruments (e.g., preview). Travellers navigate and orientate themselves by consulting memory *objects* and detecting and identifying landmarks. Consultation, detection and identification are accomplished through the *mobility instruments* of in-journey guidance, previews, probes and feedbacks. *In-journey guidance* is asking for directions and referring to objects that support external memory. *Preview* is an activity to identify objects in the environment and predict destinations. Finally, *Probing* is investigating the environment to choose the right objects to complete a task.

The model of travel aims to provide an understanding of how visually impaired people travel in an environment. Our main goal is to build Dante based upon such an understanding in order to provide better travel and mobility support for visually impaired users. However, before we can devise Dante, we need to capture the model and entire travel analysis process in a framework. Although Harper [53] has proposed a mobility analysis framework, the focus is on providing a mobility rating which shows the mobility support of the page being analysed. Fundamentally, this framework is for identifying travel objects and registering them as cues, obstacles or 'out-of-view' objects. *Cues* orientate and encourage onward navigation. *Obstacles* obstruct the progress of a traveller. *Out-of-view* objects are not obstacles because they do not inhibit travel but they are not cues either because they do not facilitate travel until they come into view. The aim of Harper's framework is to identify obstacles and turn them into cues to facilitate onward navigation and movement. In practice, this framework proved to be inappropriate and insufficiently detailed to be the basis of a tool. Although it is a useful start, we needed to create a new framework as Harper's framework is originally intended to be used by designers. It is 'craft based' and requires to be systematised. This chapter explains the new framework; first the following section explains the model of travel in detail and Section 3.3 then presents the framework.

3.2 Travel Objects

Travellers use environmental features or elements in order to make a successful journey which are called *travel objects* [73, 88]. Travellers use landmarks and memory objects to reassure themselves that they are safe to proceed and going the right way.

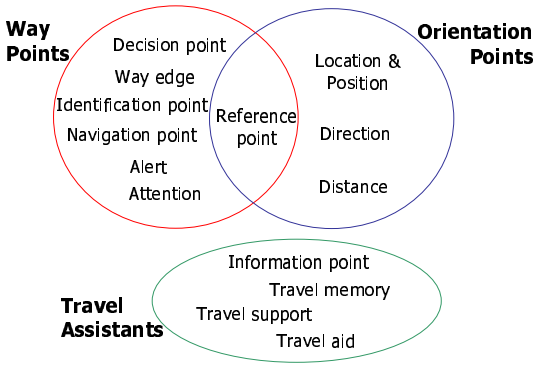


Figure 3.1: Travel Objects.

Landmarks and memory are two main classes of travel objects, and these classes also include sub-classes which are as follows [46]:

- *Landmarks*: Alert, information point, identification point, way point and way edge.
- *Memory*: Memory, alert, information point and identification point.

These classifications are dynamic and can overlap. An identification point is both a landmark and a memory object. A landmark may be classified as an information point and a way point on closer inspection. This classification depends on the context of the travel. These travel objects are an important part of the model of travel. Dante is based on this model. Fundamentally, the encapsulated process extracts travel objects from the environment in which travel takes place – a Web page. Therefore, before we can extract them, we must identify such objects and specify their characteristics. These have to be in detail and must reflect the key environmental features used by travellers. The extensions to [46] have lead to three broad categories of travel objects: (1) way points, (2) orientation and (3) travel assistants. These categories also include sub-classes which are explained in the following sections. Figure 3.1 illustrates the relationship between the categories and Table 3.1 presents these objects and examples from the real and Web world.

3.2.1 Way Points

These are the points within a journey at which a decision may be made that directly facilitates onward movement. Below, we explain the sub-classes of way points, however other classifications may also be considered as way points depending on the journey undertaken:

	Role	Real World	Web	
Way Points	**Decision Points (DP)**	Junction, corridor intersections.	Menu, check boxes, and combo box.	
	Way Edge (WE)	Wall, shorelines, hedges, fences, cliffs, railroad tracks [73], edge of lawns.	Colour boundaries, frame borders.	
	Navigation Points (NP)	Streets, walkways, canals, railroad [73], corridors and some examples specific to buildings: stairs, escalators, elevators [88].	Hyperlinks, mail to, search box.	
	Identification Points (ID)	Building name, house number, street name, campus entrance identification, floor number.	Heading, title, URL, frame title.	
	Alert (AL)	Traffic lights, signs that identify hazards, bridges, stops, and curves.	Progress bar, a search field icon can change the colour and appearance to indicate that the search is in progress, displaying or animating an hour glass.	
	Attention (AT)	Advertisements.	Animation, advertisement, banner.	
	Reference Points (RP)	Big Ben, Precinct centre bridge on Oxford road [88], sculptures, structural or decorative elements, information booths, and etc.	Logo, title banner, some aspect of the style.	Orientation
	Location & Position (LP)	"You are here" signs, signs that show the current position, reassurance signs on the motorway.	Hyperlink menus highlight the active page, some sites show the active page (e.g., home>products).	
	Direction (DR)	Signs that show the direction to cities, towns, signs with arrows.	Scrollbar direction, Back and Forward button.	
	Distance (DS)	Signs that show the distance to cities or towns.	Scrollbar (distance from top & bottom of the page).	
Travel Assistants	**Information Points (IP)**	Information or help desk, police, friends.	Search Box.	
	Travel Aid (TA)	Map, tactile map, road list, floor plan.	Site index, table of contents, site map, outline.	
	Travel Memory (TM)	Route plan, journey plan.	Previously visited links change colour, history list, bookmark.	
	Travel Support (TS)	Guided tour, visually impaired individuals may learn new routes by guiding them.	Guided tour.	

Table 3.1: Travel objects and examples.

Decision Points are the choice points where alternative paths of travel are possible. Travellers recall the direction they must travel to reach their destination and they change their direction of travel if necessary [71].

Way Edges are the environmental elements that are linear or continuous and act as

boundaries between two areas [73]. During a journey, visually impaired travellers usually make explicit references to these elements in the environment [41].

Navigation Points provide a possible route and the traveller exercises some control by choosing to follow or not to follow it. They can be considered as decision points in wayfinding, but the traveller is not choosing from a set of options; the traveller needs to decide to follow or not to follow it. They relate directly to the *paths* that are defined as channels along which people potentially move [73].

Reference Points or Landmarks are some aspects of the environment that are unique and memorable [73]. They are defined as the most salient cues in any environment [35] and are conceptually and perceptually distinct locations [66].

Increased usage of landmarks or reference points is one of the strategies used by visually impaired people to travel efficiently in their homes and communities [71]. Landmarks have to communicate some specific, identifiable features [89]. They may be primarily physical objects, but they can be sounds, odours, temperature or tactual stimuli [19, 71, 89].

Reference Point Components are directly related to the information points and are defined as two or more stimuli that, when linked, allow a traveller to determine his or her exact location [71]. A single reference point component might not be enough to identify the exact position of the traveller. Whereas, when two or more reference point components are linked, a traveller can determine his or her exact position. They are common features which do not provide precise position. However, they might help in determining one's general position.

Identification Points are identification signs that are elementary state description of a location and usually perceived when the destination is reached [88]. These points identify an object, a place or a person in the space.

Identity is what makes one part of an environment distinguishable from another; it is a characteristic that allows the traveller to differentiate parts of the environment [7]. Travellers can use identification points to validate their arrival at the destination ("this is it").

Attention These are the objects that attract traveller's attention and may change the

traveller's focus. They may be used for observations that may lead to interesting discoveries but yield most initiation control to the environment [74].

Alert These objects alert the traveller to a change in the environment or control of the journey [46]. Like the attention objects, they also attract the traveller's attention, but they usually notify approaching action or danger.

3.2.2 Orientation Points

Orientation is defined as the knowledge of one's direction and distance relative to things observed or remembered in the surroundings and keeping track of these spatial relationships as they change during locomotion [16, 19]. The concepts[1] of position or location, directionality and laterality are important cognitive components for orientation during mobility [71]. Moreover, as one moves towards a desired goal, establishing orientation and maintaining orientation are critical components of successful travel.

The knowledge about orientation suggests that a person needs information about location, distance and direction in order to be oriented in a journey. Landmarks are used to give a sense of location [62] and are defined as spatial anchors since they provide precise information about one's location [71]. Landmarks are also important for the orientation of visually impaired travellers [16].

Direction Directional information is essential to the navigator's ability to remain oriented within the environment [36, 73]. A sense of direction that is an ability to maintain direction while moving, is usually equated with a sense of orientation [88]. Directional information can be provided through the directional signs that designate direction towards a place, an object or an event in form of a name, symbol or pictograph and an arrow. They may also show which direction the traveller is moving along.

Distance The ability to make accurate distance estimations facilitates establishment and maintenance of orientation [71]. Objects that provide distance information may indicate distance from the traveller's starting position or from the traveller's destination. They may also show where one is with respect to nearby objects and the target location.

[1] Concepts are defined as mental representations, images or ideas [71].

Location or Position Landmarks provide implicit location or position information, because different travellers may need varying amounts and type of information about landmarks. However, there may be some objects in the environment which provide location or position information explicitly. These location or position objects are directly related to reassurance signs and are defined as checkpoints which are used by travellers to reassure themselves that they are on the right track [88].

3.2.3 Travel Assistants

Sighted or visually impaired travellers may all experience problems in orienting themselves from time to time in an unfamiliar or familiar environment. They use different strategies for solving this problem including consulting a map; exploring the space systematically, either alone or with a guide; or following verbal or written directions [71].

Information Points By using these objects, a traveller can directly request information. The traveller controls the type and amount of information requested and supplied, so they are an active information supply [71].

Travellers can update their spatial information by interacting with other people while travelling. This is one of the strategies used by travellers for re-orienting themselves [71]. A supportive environment can be thought of in terms of information points at frequent and regular intervals. Particularly, information points may be important for visually impaired travellers, since they compensate for not having access to distant cues that are so useful to the sighted travellers [90].

Travel Aids provide an overview of the environment. They usually place the entire environment within the traveller's view. We refer to them as passive information supply, because, unlike information points, the traveller does not control the type and amount of information. They can also be considered as secondary sources, which can be used for spatial knowledge acquisition. They may help travellers to determine their position in the environment, their direction of travel, and the relative position of other objects or places in the environment [35].

Travel Memory holds information about where the traveller has been and provides means to get back there. It can be considered as an external memory aid to supplement internal memory [46].

Travel Support A traveller may make a journey without controlling all the details of the journey, that is to say, the traveller may not actively control the journey undertaken. Travellers may make a journey by actually being guided throughout. This could be a strategy for travellers to learn the spatial relationships in an unfamiliar environment [35].

3.3 The Travel Analysis Framework

This framework is based on the model of travel. It aims to capture a systematic method to discover objects and their roles as explained in the previous section. Fundamentally, the framework is comprised of two stages:

1. Inspecting a Web page in order to create a travel object inventory;
2. Classifying each travel object in the inventory according to the role it plays in the travel process.

In Dante, the implementation of these two stages form the basis of the annotation and transformation components (see Figure 1.2 in Chapter 1). The following sections explain these two stages in detail.

3.3.1 Identification of Travel Objects

If we look at the Web landscape from the real world travel perspective, travel objects also exist in the Web landscape[2] (see Table 3.1). For example, if we analyse the home page of HSBC from the real world travel perspective (see Figure 3.2), we can identify some of these objects. The logo at the top left corner acts as a *reference point*; it is a unique and memorable feature of the page. The colour boundary on the side is a *way edge*; it is used to separate the "side content" from the "main". The heading "personal banking" is used as an *identification point* and used to identify that part of the page. As these examples demonstrate, the travel objects play an important role in the mobility of the Web users. Since in the visually impaired users' Web landscape these objects are not presented appropriately, their mobility is reduced. Travel objects should be presented in a way that they can fulfill their intended roles and ease travel on the Web.

In the first stage of the framework, Web pages are analysed to find out the provided travel objects and create a travel object inventory. The aim of identification is to filter

[2]The Web landscape is defined as the combination of the page and the agent (e.g., browser) [46].

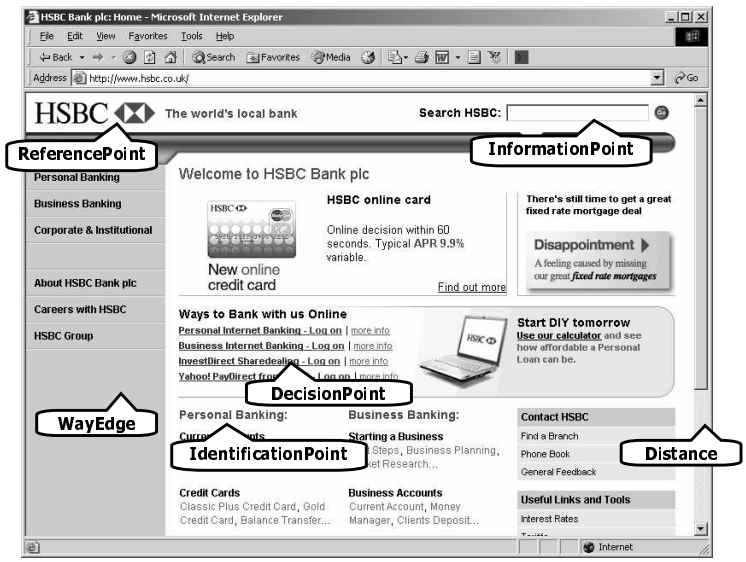


Figure 3.2: Example travel objects on the home page of HSBC (03/10/2004).

the page and find the objects that are useful in promoting the onward journey. These objects are the regions or portions of the page i.e., an HTML element, collections of elements or parts of elements depending on the rendering. In order to identify these objects, a journey should be made from top to bottom by noting which parts of the page (rendering or underlying code) are useful in promoting the onward journey. These then become travel objects.

A set of guidelines are also developed for identifying objects. These guidelines are established by investigating a large number of Web pages. The aim is to make the identification process systematic and consistent. If the process is proved to be so, the guidelines can form the basis of heuristics for travel object identification within an automated process. These heuristics can also evolve throughout the application. The guidelines are principally grouped into four. They are summarised as follows:

- **Extracting travel objects from a page** These are the fundamental strategies for extracting travel objects, and the important aspects about the environment and travel objects. E.g.,
  - *A bird's-eye view* of a page may help to spot visual groupings and draw a sketch of the page. Then the sketch can be extended by zooming in and out from these groupings and by considering their relationships.
  - *Granularity* A travel object may be atomic or composite (i.e., composed of other travel objects).

  - *Content or Context Change* When a page is analysed linearly (from top to bottom), any context or content change can be considered as a clue to differentiate objects.

- **HTML source code** Some details are hidden in the source code and some are in the rendering, so it is important to inspect both. E.g.,

  - *An image map* When the rendering of a page is analysed, it may be difficult to realise the image maps, whereby analysing source code may help to elucidate the details of the image maps.
  - *Layout tables* The details of invisible layout tables can be obtained from the source code. Sometimes groups of objects are located in a cell of the layout table, so it may help to check whether or not they can be considered as a single travel object.
  - *Image with a link* It might not be easy to recognise whether there is a link on an image or not so it might help to check the source code.

- **Using elements of a document** The general knowledge about the structure of a standard document can be used in finding out the travel objects (the role of a part of a document can be considered). For example, sections, paragraphs, titles, headings, bulleted or numbered lists are all different travel objects. Similarly, the HTML elements can also be considered, for instance, links. E.g.,

  - *Sections and headings* The heading and the content part (section) should be considered separately because it is likely that they have different roles during a journey. The headings can be obtained by checking whether the source code contains H1 through H6 tag set. However, this may not be enough because not all headings are explicitly specified by using this tag set. Different typefaces may be used to indicate headings; thus it is important to inspect both the rendering and the underlying source code.
  - *Boundaries* (e.g., line, colour and space) are used to visually divide information or sections. For example, a line boundary can be created by using a HR tag. Besides helping in recognising the context division in a page, they may also be considered as travel objects.
  - *Links* All the links on a page are candidates for being travel objects despite the fact that they could be grouped together with other objects depending on the context.

  - *Logo* If an image or icon is used to identify that particular page, then it can be considered as a travel object.
  - *Animations and Adverts* can be considered as single objects as they usually have different context.

- **Neighbourhood objects** Objects that are grouped together to provide a common function can be considered as a single travel object. E.g.,
  - *Functional dependency* If the functionality of an object depends on anther object then they can be considered as a single travel object. For instance, the search capability in a page is usually provided by a search box, a 'go' button and a label. These three objects can be considered as a single travel object because the functionality of the 'go' button depends on the search box and cannot be used on its own.
  - *Single destination* If a group of objects (e.g., an image and a paragraph) links to the same page, then that group can be considered as a single object.
  - *Common style* If objects with different content are presented with the same style (e.g., menu items), then it is likely that those objects form a composite object.
  - *Interaction* If the traveller needs to interact differently with consecutive objects then they can be considered as different objects.

3.3.2 Classification of Travel objects

This second stage aims to classify the extracted travel objects. The main use of this classification is to discover the roles of each travel object in the inventory. Every travel object has at least one role during a journey and depending on the journey, it may have more than one role or it may have different roles in different journeys. Since we cannot consider all the different possible journeys, we only consider the possible roles of travel objects in a general context.

The classification process consists of a series of questions (a questionnaire) that have to be answered for every object in the created inventory (see Table A.1 in Appendix A). These questions aim to capture the definitions of the classifications of travel objects (see Section 3.2 and Table 3.1). For example, a *Travel memory* is defined as "an object that holds information regarding where the traveller has been and

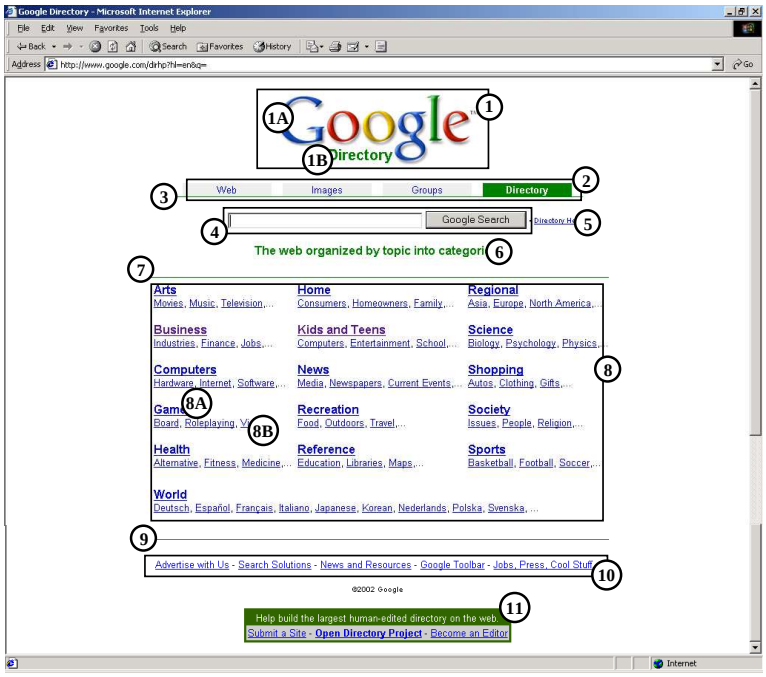


Figure 3.3: The home page of Google Directory (12/03/2002).

provides means to get back". Therefore, questions like "does this object show where the traveller has been?", "does it show previously visited places", etc., are included.

The expected answers to the questionnaire are *yes* or *no*. This is to simplify the process, make it systematic and have it in a form which is easy to automate. The results are then evaluated to infer the possible roles of the travel objects. The aim of asking every question to every object is to try to decrease the subjectivity of the approach and to provide a systematic rationale to classification.

3.4 Evaluation of the Framework

The purpose of the evaluation was to test whether the framework can be used to analyse the travel support offered within a Web page and used as a basis of a tool such as Dante.

Two evaluations were conducted: The first analysing the Google Directory page and second the home page of the Royal National Institute of the Blind (RNIB) by using the framework. The same process was followed in both evaluations and can be summarised as follows:

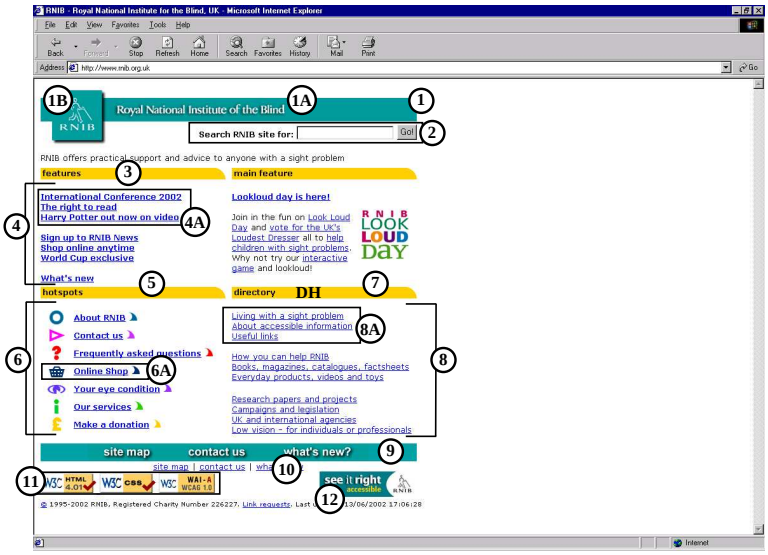


Figure 3.4: The home page of RNIB (12/06/2002).

Hypothesis If the users of the framework would be able to create a travel object inventory, classify travel objects and the results should be consistent between users, then we would be able to design a tool to support this framework.

Participants Twelve participants were used in the first evaluation (referred to as P1–P12). All the participants were experienced Web and HTML users. They had created at least one Web site before. In the second evaluation, six of the twelve participants of the first evaluation were reused.

Materials In the first evaluation, twelve participants analysed the Google Directory page (see Figure 3.3). This page was used because the design is simple, navigation based and provides many travel objects. We did not want to use a complex page and confuse the participants, because they were not familiar with the notion of travel and did not have experience in identifying travel objects. Moreover, it is obvious that it was not designed with accessibility in mind.

In the second evaluation, six participants analysed the RNIB home page (see Figure 3.4). Compared to the Google Directory page, this page has a linear structure, it is more text-driven. While the design of this page is not as simple as Google Directory, the main design focus of this page was providing accessibility.

Evaluation procedure The same procedure was followed in both evaluations. We

first explained the problem domain, the goal of the framework and discussed some accessibility issues with each participants. We also explained the notion of travel on the Web and mobility. Then we asked them to apply the framework as follows:

- Inspect the Web page to identify travel objects by using the provided guidelines;
- Answer the questionnaire to classify the identified travel objects.

After the participants completed these tasks, we asked them a set of questions concerning the overall usability and efficiency of the framework. For example, whether participants found the guidelines and examples useful, how easy they found the overall framework, how difficult they found it to answer questions as "yes" or "no", etc., (see Table A.1 in Appendix A). These questions are rated on a five point rating scale, 1=very difficult (not useful at all) and 5=very easy (very useful).

3.4.1 Results and Discussion

Both evaluations were successful in demonstrating the applicability of the framework. The participants were able to create inventories of the travel objects on both Web pages and classify the travel objects in their inventories. The created travel object inventories were highly consistent and were able to demonstrate that the encapsulated process in the framework is systematic.

Table 3.2 shows the analysis results of the Google Directory page of Figure 3.3. Similarly, Table 3.3 shows the analysis results of the RNIB home page of Figure 3.4. On average, each travel object on the Google Directory page (Figure 3.3) was extracted by 84.8% of the participants and this average increased slightly in the analysis of RNIB home page (Figure 3.4) to 88.8% (see Table 3.3). While most of the extracted travel objects were common in the inventories of all the participants, there were also some objects that were considered at a different granularity[3]. For example, although ten participants considered logo (1) on the Google Directory page as a single travel object, two participants considered parts of it ("Google" (1A) - "Directory" (1B)) in their inventories rather than considering them as a single travel object. Similarly, some objects were considered as composite objects: the participants included both the entire

[3]These objects are indicated as A and B in Tables 3.2 and 3.3.

object and parts of it in their inventories. For example, hotspots list (6) on the RNIB page was identified by all the participants and two participants also considered a part of it (6A) as well as the entire object in their inventories.

As can be seen from Tables 3.2 and 3.3, classifications of extracted travel objects from both pages were also consistent. For example, line separator (7) on the Google Directory page was identified and classified as a way edge by all participants. Participants were also able to classify objects in more than one group, for instance, the hotspots list (6) on the RNIB page was extracted by all participants. This object was then classified as a decision and navigation point by all participants and as travel memory by three participants. In both analyses, participants failed to classify travel objects as travel memory. This could be because they had to visit the links and then answer the questionnaire, but they mostly answered the questionnaire without following the links.

Generally, the classifications of the travel objects on the RNIB page (Figure 3.4) were more consistent than the Google Directory page (Figure 3.3). This could be because participants became familiar with the questionnaire and the classifications, or it could be because the role of travel objects were more precisely specified on this page because the design focus was providing accessibility.

In both evaluations, some aspects were also investigated concerning the suitability of the framework for automated and manual implementation. The following issues are related to the automated implementation:

- **Applicability of the guidelines** In order to validate potential heuristics that can be developed based on the guidelines, participants were asked if they have used guidelines and how useful they found the guidelines. In both evaluations, all the participants followed the guidelines and on average they were rated as very useful.

- **Answering questionnaire** In the second part of the framework, the participants successfully answered the questionnaire for every object in their inventories. The expected answers were *yes* or *no*, rather than on a scale of 1 to 5. This is aimed to ease the automation. Participants indicated that they found this part of the analysis easy.

	Object	**Total**	**DP†**	**WE†**	**NP†**	**RP†**	**RPC†**	**LP†**	**DR†**	**DS†**	**ID†**	**AL†**	**AT†**	**IP†**	**TA†**	**TM†**	**TS†**
1	Logo	10				10∗	2	3			5∗						
1A	“Google”	2		1		1	2						1				
1B	“Directory”	2					2				2						
2	Main menu	12	12∗		10∗			3∗	5				3		3		
3	Line separator	5		5∗													
4	Search box & go	11			8∗	3							4	10∗			
5	Help link	10		1	9∗						2				4	1∗	
6	Description	8					3	1			4∗				5		
7	Line separator	12		12∗													
8	Categories table	12	11∗		11∗	4	1		3		2			1	3	3∗	
8A	Category title	4	1		4						2				2	2	
8B	Category list	3	2		3				2							2	
9	Line separator	11		11∗													
10	Footnote menu	12	10∗		11∗				1		3		1		2	3∗	
11	Footnote	9	3∗	6∗	3∗						2		6				∗
	Average (except 1A-B, 8A-B)	**10.18 84.8%**															

† See Table 3.1.
∗ Potential roles of the travel objects.

Table 3.2: Travel objects on the home page of Google Directory.

	Object	**Total**	**DP†**	**WE†**	**NP†**	**RP†**	**RPC†**	**LP†**	**DR†**	**DS†**	**ID†**	**AL†**	**AT†**	**IP†**	**TA†**	**TM†**	**TS†**
1	Logo & banner	5		∗		5∗	2	1			1∗						
1A	Banner	1				1					1						
1B	Logo	1				1											
2	Search box & go	6			3∗	2					2			5∗			
3	Features heading	6		4∗			1		1	1	6∗						
4	Feature list	6	5∗	2	6∗				1						2	3∗	
4A	Features sub-list	1	1	1	1											1	
5	Hotspots heading	6		4∗			1		1	1	6∗						
6	Hotspots list	6	6∗	2	6∗				1				∗		3	3∗	
6A	Hotspots item	2			2											2	
7	Directory heading	6		4∗			1		1	1	6∗						
8	Directory list	6	5∗	1	6∗				1						2	4∗	
8A	Directory sub-list	1	1	1	1											1	
9	Site menu	3	3∗	3∗	3∗				1		2				2		
10	Site menu (text)	3	3∗		3∗				1						1	1∗	
11	W3C approvals	5	3∗		5∗								4∗		1		
12	RNIB logo (small)	6			6∗		∗				2		2				
	Average (except 1A-B, 4-6-8A)	**5.3 88.8%**	† See Table 3.1. ∗ Potential roles of the travel objects.														

Table 3.3: Travel objects on the home page of RNIB.

Participant	Time (Google)	Time (RNIB)
P1	2:30h	1:00h
P2	2:15h	1:15h
P3	2:10h	1:05h
P4	2:00h	1:15h
P5	1:30h	
P6	1:10h	
P7	2:15h	1:00h
P8	1:40h	
P9	3:00h	
P10	2:30h	2:00h
P11	2:45h	
P12	3:00h	
Average	2:13h	1:15h

Table 3.4: Time taken for analysing the Google Directory and the RNIB page.

The following issues are discussed with regard to the manual implementation of the framework:

- **Application time** Although the exact time to apply the framework was not recorded, each participant was asked to provide the approximate time that was spent to complete the framework. Table 3.4 shows the approximate times to complete the first and the second evaluation. As it can be seen from Table 3.4, the participants spent significantly less time analysing the RNIB home page than analysing the Google Directory page. The time difference between two analyses suggests that, after an analyst analyses three or more pages, it will take him(her) quite a short time to apply the framework. Additionally, some participants, for instance participant P2, indicated that the first analysis required a lot more concentration than the second analysis because of being unfamiliar with the framework and also with the concept of travel.

- **Understanding travel** We asked participants to rate their understanding of travel on the Web before and after the evaluations. This is important as a better understanding can lead analysts to do more accurate and efficient analysis. After the first evaluation, ten participants stated that the framework improved their understanding of travel on the Web. Participants P9 and P11 stated that the framework did not change their understanding, however it has changed their point of view regarding travel on the Web.

- **Usability of the framework** We asked participants questions concerning the

difficulty of the overall framework and the two stages of the framework individually. After the first analysis, on average, the overall framework and the first stage, were rated as easy, and the second stage was rated as moderate. After the second analysis, on average, only the rating for the second stage was changed from moderate to easy. This could be because of familiarity with the questionnaire in the second stage. These results suggest that, after several analysis, it will be very easy for users to apply the framework.

In both evaluations, although the participants did not have difficulties classifying the travel objects concerning the proposed classifications (see Table 3.1), the following suggestions and feedback were provided:

- Participant P11 mentioned that location and position should not be considered in the same class, because they suggest different granularity. For example; location (I am in Manchester), position (I am at the phone box, outside the Precinct centre in Oxford road in Manchester). In this context, some objects were providing the location information but not the position, so it was difficult to answer some of the questions concerning this class.

- In the definition of direction, we only considered the movement direction in the page (up, down, left, right), but some participants stated that it should include the journey direction as well.

- Similar to direction, in the definition of distance, we only considered the distance in a page, for instance, how far is the traveller from the bottom of the page. Some participants indicated that the journey distance should also be considered (e.g., the number of links that a traveller should follow to reach the destination).

Consequently, we modified and extended the travel object classifications by addressing these suggestions and feedbacks.

Finally, the following issues were also revealed from the evaluations:

- **Importance of the framework** The site menu (9) and site menu (text) (10) objects on the RNIB page (Figure 3.4) were important for the evaluation. Although the site menu (9) object looks like a repetition of the site menu (text) object (10), actually it is not; they have different roles in the page and the framework can draw the distinction between them. The site menu (9) object is a way edge but the site menu (text) (10) is not. Similarly, site menu (text) (10) object is a travel

memory but the site menu (9) is not. One Participant commented that these objects are good examples that illustrate the importance of the framework. Furthermore, these objects also demonstrate what is missing in text-only browsers and the way screen readers render a Web page.

- **Environment** Although the participants were informed that the environment is composed of the underlying browser and the page itself, most of the participants did not identify the travel objects provided by the browser. This could be because participants usually use the same browser and after a while they use the browser instinctively, without considering the facilities provided by the browser. When they access the Web, the only changing thing in their environment is the retrieved page. Additionally, participant P11 stated that he (she) did not consider the browser because it is the third party and difficult to change, modify or improve, but we could try to change or improve the design of the Web pages.

As a conclusion, the hypothesis was confirmed. All participants were able to create an inventory, classify objects in that inventory and the results were consistent between the participants. All the issues discussed above also demonstrate that the framework can be used as the basis of a tool.

3.5 Summary and Conclusions

In this chapter, first, the notion of travel and mobility on the Web has been introduced. Then the travel analysis framework and its evaluation has been discussed. The main goal of this framework is to identify travel objects to create an inventory and classify them concerning their roles in a typical journey. The conducted evaluation demonstrated that the framework is systematic and rigourous enough to be used as the basis of our tool, Dante. The framework provides the groundwork for the foundational components of Dante and provides the basis for the annotation and transformation components (see Chapter 5).

Although the travel analysis framework provides techniques to identify travel objects and their roles, we still need to associate identified information with these objects in order to make them accessible to tools such as Dante. One possible way of doing this is annotating identified objects to make them and their roles explicit [51]. Such explicit annotations can make these objects easily accessible to automated tools. However, in order to do this we need to capture knowledge regarding these objects in a way that

enables us to annotate them explicitly. A widely known way of knowledge capturing is creating “ontologies”. An ontology allows us to create a controlled vocabulary which is computationally available. The following chapter explains our ontology, WAfA, which is created for this purpose. Fundamentally, WAfA provides a vocabulary that can be used to annotate pages to make identified objects explicitly available. Dante can then use these annotations to transform pages into a form in which the objects can play their intended roles.

Even though the model of travel provides an understanding of the requirements of visually impaired travellers [46, 125], we still need to have a good understanding of the environment itself. The combination of understanding of the user requirements and the environment can enable us to provide better travel and accessibility support. Although the travel analysis framework provides an understanding of how travel objects are used or needed to be used in a journey, we still lack the understanding of the underlying structure of the environment– that is Web pages: What are the main components or building blocks of Web pages, how do they relate to each other, what is the relationship between how these objects are presented and used and what is the importance of contextual knowledge. Therefore, a part of WAfA explained in the next chapter aims to answer these questions. In particular, a sub-ontology is created for capturing knowledge about the structure of Web pages.

Chapter 4

The WAfA Ontology

The Web is a virtual environment through which people make journeys. Moving around in such an environment requires knowledge and understanding of visual objects, their properties and purpose in that environment. The visual design of the Web provides implicit knowledge that sighted people use to move around; such knowledge needs to be made explicit in order to be accessible when the visual design cannot be seen. This chapter presents an ontology, the WAfA ontology, which aims to capture parts of the knowledge regarding that environment in a manner that enables computational support for humans and technologies that cannot see the visual design. The WAfA ontology mainly describes: structure (taxonomy of visual components that constitute a Web page), structural abstraction (the relationships between components and their connectivity) and meta-knowledge (principles and sets of rules).

Fundamentally, the WAfA ontology is composed of three sub-ontologies: authoring, mobility and contextual semantics. Authoring semantics define the structural properties of Web pages, and similarly, mobility semantics define the navigational properties. Contextual semantics define concepts to encode the context of a journey. The main goal of this ontology is to drive the knowledge driven pipeline encoded in Dante (see Chapter 5).

4.1 Motivation and the Purpose

The visual design of Web pages provide implicit knowledge that sighted people use to move around. For example, the home page of the Royal Mail[1] has a large number and variety of spatial visual cues (e.g., font, colour, layout, tables, figure, changes in the

[1]See `http://www.royalmail.com/`.

background, highlighted areas, etc., see Figure 4.1). While these cues are mainly used to create aesthetically pleasing environments and structures, they actually provide visual guidance to ease access to the environment [119]. However when we analyse the HTML source code as screen readers do (e.g., Jaws), only images (IMG tag), anchor tags (A tag) and tables (TABLE tag) can be detected. But if we carefully investigate the source code, the designer actually provides significant information about these components and the structure of the page in the comments, class and id attributes, file names and alternative text provided for images. For example, in the source code of Figure 4.1, there are comments such as "Header", "Menu", "Logo", "Topnav", etc. However, since this kind of knowledge is not provided in a *controlled* way, it is not accessible to screen readers and to other user agents, unless they use natural language processing techniques. Therefore, the encoded structural knowledge needs to be provided in a form that is interpretable by user agents. Therefore, one part of the WAfA ontology encodes concepts to describe the structural properties of Web pages.

If we further investigate Figure 4.1 and look at it from the travel perspective there are a number of travel objects on this page. For example, the logo of the Royal Mail can be considered as a unique and memorable feature of the environment [66] – a landmark or *reference point*. Distinctive colours used for sidebars can be considered as *way edges*; they are used to separate main content from the "side" content. We need to make navigational as well as structural properties of Web pages explicit and computationally accessible so that user agents can provide better travel support. The WAfA ontology is an evolving ontology for describing the foundational model of Web pages; it is concerned with the representation of concepts and relationships necessary for the modelling of the structural and navigational properties of Web pages. The main purpose of this ontology is to drive the annotation pipeline encoded in Dante.

WAfA is regarded as foundational for two reasons: 1) the structural concepts and relationships focus on providing basic components of Web pages and their relationships rather than addressing a specific type of Web page [6]; and 2) the concepts refer to vocabularies used in a number of domains such as hypermedia design [72], transcoding [2] literature [24, 58], etc., and so are generalised to all these domains. The ontology does not provide a model of the content of a Web page [10] but it aims to provide a model of the structural and navigational organisation of a Web page. However, the content and the structure are interlinked; the way a document is structured is based on

[2]**Transcoding** is the process of transforming the representation of the content into another format (see `http://www.research.ibm.com/networked_data_systems/transcoding/`).

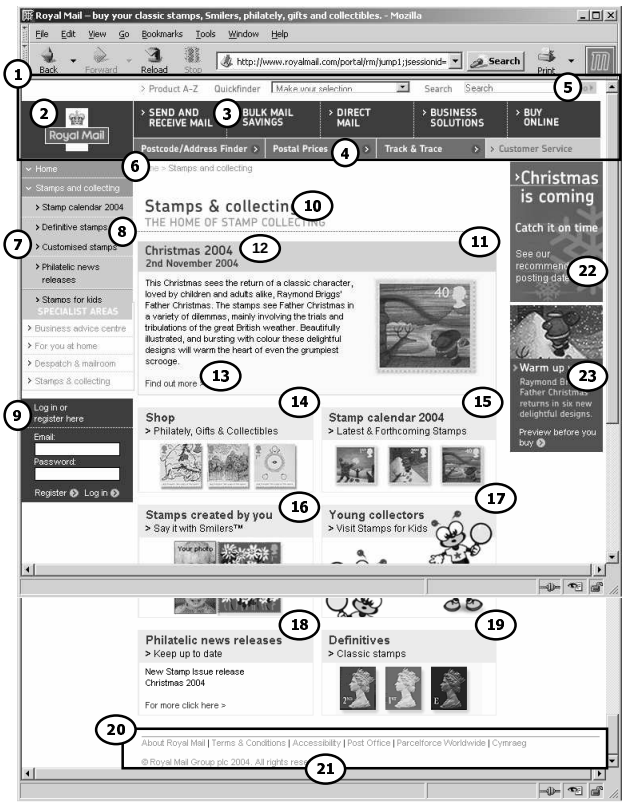


Figure 4.1: The home page of Royal Mail (03/10/2004).

the content, and the way the document is structured affects how the reader reads and comprehends the content [118]. As McKnight *et al.* [77] pointed out, readers tend to process *structure* for retention tasks and *meaning* for comprehension tasks. Therefore, our aim is neither to separate the structure from the content nor to address the meaning of the content, but to make the structural and navigational properties explicit and computationally accessible.

4.2 The Design Process

There are a number of methodologies for designing ontologies [30, 42, 75, 92, 112]. These methodologies mainly include a set of stages that occur when building ontologies, guidelines and principles to assist designers, and a life-cycle to indicate the relationships between the stages. Methodologies broadly divide into those that are stage based (e.g., TOVE [112]) and those that rely on iterative evolving prototypes (e.g., Methontology [42]). These methodologies are in fact complementary [30]. Figure 4.2

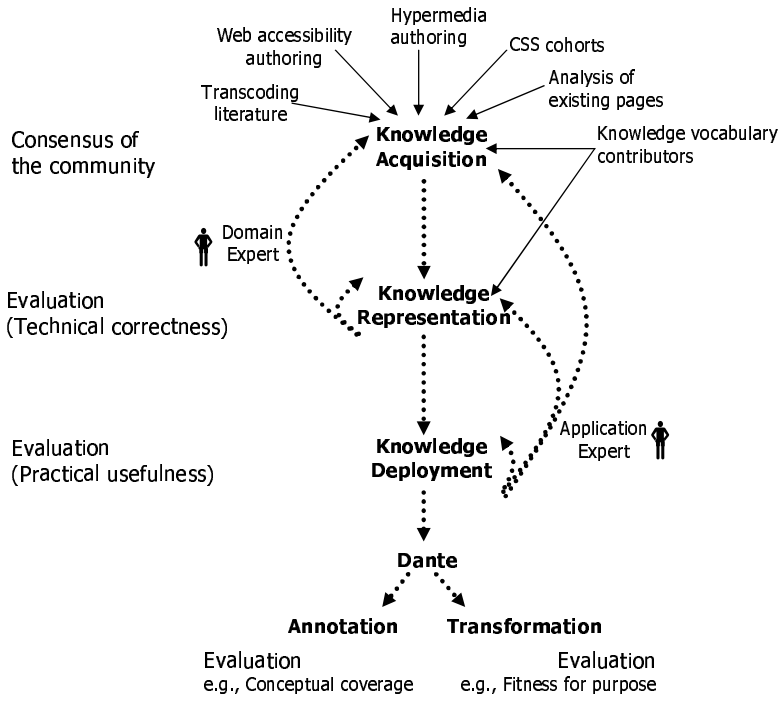


Figure 4.2: The design process of the WAfA ontology.

depicts the approach used in designing the WAfA ontology. This approach, which is based on the Methontology, has the following three stages:

Knowledge Acquisition is the process of acquiring domain knowledge from which the ontology will be built. This is a support activity in Methontology. A large part of the knowledge encoded in WAfA is extracted from the hypermedia authoring literature [72], transcoding literature [24, 58], mark-up languages [115], Web accessibility work [87, 109], voice interfaces [91], CSS cohort[3], etc. The analysis of existing Web pages (e.g., comments, id attributes, etc.) and the travel analysis framework presented in the previous chapter also contributed to the ontology.

Knowledge Representation involves *conceptualisation* and *implementation* activities. Methontology defines these as the development activities. *Conceptualisation* is the activity of identifying the key concepts, their properties and relationships. Since we used existing literature to identify concepts, we used a bottom-up approach to build the ontology; we started with most specific concepts and grouped

[3]See `http://www.w3.org/Style/CSS/`.

them into categories. The *implementation* activity is representing the conceptualisation in a formal language. The ontology was implemented using OWL [76]. Throughout the implementation stage, we also focused on evaluating the technical correctness of the ontology (e.g., consistency, redundancy, etc.) [47].

Knowledge Deployment In this stage, the WAfA ontology is deployed in Dante. The deployment process was important to evaluate and refine the ontology. For example, when we started to annotate pages in Dante, some missing concepts were identified so we turned back to the previous stages and added these concepts to the ontology. The transformation part also helped us to identify some problems with the encoded hierarchy which we later augmented to better support the transformation techniques encoded in Dante.

The overall process was iterative; a rough first version of the ontology was implemented and then deployed in Dante. This highlighted some of the issues that needed further investigation. Augmenting the ontology to address these issues helped us to refine it. Then we continued applying the same process until the ontology was in a satisfactory form to support the annotation and transformation processes encoded in Dante. The iterative process was important to ensure that the ontology was evaluated at every stage of the design and implementation processes.

4.3 The Implementation

Fundamentally, WAfA has the following three components[4]:

- *Taxonomy* Concepts are organised in a superclass-subclass hierarchy, which is also known as a taxonomy. It provides specialisation and generalisation relationships between concepts which are introduced to enable the sharing of some properties between concepts. This mainly provides the skeleton of the foundational model (e.g., `Header` *is-a* `Chunk`). Besides the *is-a* hierarchy, *part-of* relationship also provides a part-whole hierarchy in the WAfA ontology. We will discuss this in detail later.

- *Structural abstraction* describes the relationships of the concepts in the taxonomy (e.g., `Logo` *part-of* `Header`).

[4]Different typographies used in this chapter have the following associations: Names of concepts represented in the ontology are in `Courier New font`; relationships between concepts are in italics enclosed by hyphens, e.g., *part-of*.

- *Meta knowledge* provides some principles about the overall model (e.g., Disjoint (`Header`, `Footer`) which means an object cannot be a header and footer at the same time).

Fundamentally, the ontology is organised as three sub-ontologies. Table 4.1 provides an overview by presenting higher level concepts and the number of their children[5].

Authoring Semantics Encapsulate information about how the components of a Web page are *structured* and organised to form the overall structure. The concepts in this part of the ontology are subsumed by `AuthoringConcept`. Additionally, the concepts subsumed by `Connection`, `Coverage`, `Position`, `Order` and `Structure` are used to define the `AuthoringConcept`s (Section 4.3.1);

Mobility Semantics Encapsulate knowledge about the travel objects from real world mobility studies. This part of the ontology focuses on how the objects are *used* in a typical journey. Objects can have a journey role which depends on the context of the travel task and can also have one or more environmental roles depending on the underlying environment. The concepts in this part of the ontology are subsumed by the `MobilityConcept` (Section 4.3.2);

Contextual Semantics A Web journey can take place in different contexts and concepts in this group provide contextual knowledge about a journey such as the purpose of the journey being undertaken, etc. The concepts in this part of the ontology are subsumed by the `Purpose` concept (Section 4.3.3).

These sub-ontologies of the WAfA, particularly authoring and mobility semantics, provide three levels of knowledge about the page itself which is illustrated in Figure 4.3. In the first and basic level, HTML and CSS is considered to provide knowledge about the physical and presentational properties of the page components. In the second level, some HTML elements and the authoring semantics provide information about the logical structure of the page and in the last level, mobility semantics are used to provide knowledge about the navigational properties. For example, in Figure 4.1, object marked as 2, is an image in the HTML source code. By using our authoring concepts, we can specify that object as a `Logo` which provides information about the structural inventory of the page. Finally, with mobility concepts, we can represent that

[5]The complete ontology is available at `http://augmented.man.ac.uk/ontologies/wafa.owl`.

Part of the Ontology	**Concept** (Children of Thing)	**Example Children**	**Total Children**
	`Object`		
1.Authoring Semantics:	`AuthoringConcept`		
	`Atom`	`Logo, Headline, Footnote.`	43
	`Chunk`	`Header, LinkMenu, Section, Abstract.`	42
	`Node`	Represent a Web page.	0
	`Collection`	Represent a Web site.	0
Defining Authoring Semantics:	`Connection`	`AssociativeConnection, ReferentialConnection.`	3
	`Coverage`	`Inter, Intra.`	3
	`Position`	`AssociativePoint, CurrentPoint.`	2
	`Order`	`AlphabeticalOrder`	5
	`Structure`	`Hierarchy, Linear.`	4
2.Mobility Semantics:	`MobilityConcept`		
	`EnvironmentalRole`	`WayPoint, TravelAssistant.`	14
	`JourneyRole`	`Obstacle, Cue, OutOfView.`	7
3.Context Semantics:	`Purpose`	`ObjectPurpose, TravellerPurpose.`	14

Table 4.1: The overview of the WAfA ontology.

object as a `ReferencePoint` which provides implicit knowledge about the location of the traveller (see Table 4.2 for more examples).

Although our ontology is chiefly an asserted superclass-subclass (*is-a*) hierarchy, part-whole relationships also play an important role in the representation. Even though OWL does not provide any special constructs for dealing with part-whole relations, it contains sufficient expressive power that allows us to represent part-whole relationship in the WAfA ontology[6].

The formal theory of part-whole relation is called "mereology". Different part-whole relationships are identified in the literature. Winston *et al.* [117] differentiate six types: (1) component-integral object (pedal-bike), (2) member-collection (ship-fleet), (3) portion-mass (slice-pie), (4) stuff-object (steel-car), (5) feature-activity (playing-shopping) and (6) place-area (Everglades-Florida). According to this classification, in the WAfA ontology, we are representing component-integral object relationship when we talk about *part-of* relationship. An integral object is defined a particular kind of

[6]See `http://www.w3.org/2001/sw/BestPractices/OEP/SimplePartWhole/index.html`.

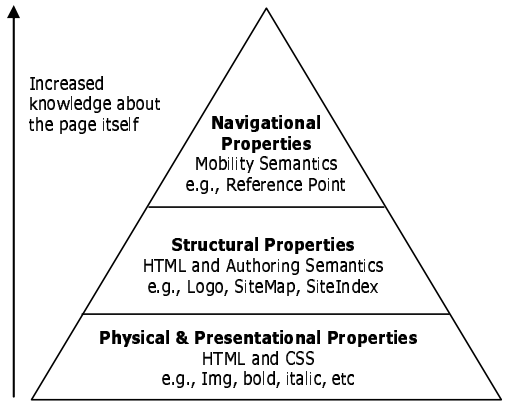


Figure 4.3: Knowledge about a Web page.

whole which is divided into components. Winston *et al.* [117] also define three relation elements to define different types of part-whole relationships: functional, homeomerous, and separable. Functional indicates that the part has a function with respect to the whole; homeomerous means that the part is identical to the other parts making up the whole; and separable means that the part can be separated from the whole. Component/integral object relation is defined as functional, nonhomeomerous and separable. Therefore, in the WAfA ontology when we define "Logo *is-part-of* Header" we mean the logo has a specific function with respect to the whole, does not resemble other parts of the header, and can be separated from the whole (Header). Some other representation details of *part-of* relationship are as follows:

Transitive property The part-whole relation is transitive and OWL allows us to specify that. For example, if A is part of B, and B is part of C, then A is part of C. For example, in Figure 4.1 at the top, `Link` labelled as "Send and receive mail" *is-part-of* `LinkMenu` (3), and `LinkMenu` *is-part-of* `Header` (1), then `Link` labelled as "Send and receive mail" *is-part-of* `Header`. But we also need to create restrictions about the direct parts of certain kinds of objects, therefore we introduced *is-direct-part-of* property which is a sub-property of the *is-part-of* and is not transitive. For example, if A is direct part of B and B is direct part of C, then we cannot say that A is direct part of C.

Inverse property OWL supports inverse relations, so we define *has-part* as the inverse of *is-part-of* which is also transitive. Therefore, if we are talking about **linkmenu** which is an instance of the `LinkMenu` class and **header** which is an

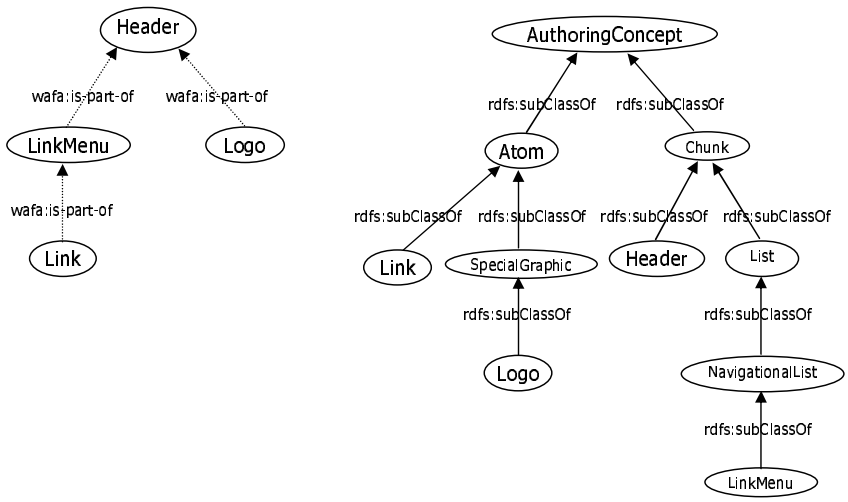


Figure 4.4: *part-of* hierarchy vs. *is-a* hierarchy.

instance of the `Header` class and if we know that **linkmenu** *is-part-of* **header**, then we can say that **header** *has-part* **linkmenu**. We also introduced *has-direct-part* property which is a sub-property of the *has-part* and is not transitive.

is-a relationship vs. part-of Although both relationships generate hierarchies, it was important to make the distinction clear. Figure 4.4 shows an example hierarchy of both by using the objects identified on Figure 4.1. For example, in the *part-of* hierarchy, `LinkMenu` *is-part-of* a `Header` and in the *is-a* hierarchy, `LinkMenu` *is-a* `NavigationalList`.

The following sections present the three subontologies in detail. A description of early work of this ontology can also be found in [122, 124].

4.3.1 Authoring Semantics for Conceptual Structuring

This sub-ontology aims to formalise the common understanding of a Web page structure. It defines a vocabulary that is already widely used within the Web community to describe components of Web pages which is not formally explained and defined [24, 58, 72, 87, 91, 109, 115]. When we analyse HTML source codes, apart from the HTML elements, designers put a lot of knowledge about the structure in their source code. But since structural elements in HTML (e.g., H1-H6) are not extensive enough, such knowledge is not encoded in a *controlled* way.

HTML[7] is a mark-up language. Mark-up is conventionally divided into three classes: presentational, structural and descriptive or semantic [29, 77]. With presentational mark-up, the visual form of the document is directly specified which controls the output interpretation by output devices (e.g., PDF). Structural mark-up provides a set of tags to represent the logical structure of a document (e.g., Latex[8], DocBook [115]). Descriptive mark-up considers a document as structured objects containing semantically interpretable parts (e.g., XML[9]). Since no directions regarding presentation are included either with the structural and semantic mark-up, the interpretation of the tags occurs entirely within the output system [77]. HTML is a Standard Generalized Markup Language (SGML)[10] application; it provides a fixed set of tags and was originally designed to present the logical structure of the document[11]. However, its element set is extended with a number of presentational elements. Therefore, HTML mixes structural (e.g., H1. . .H6, P, TITLE elements) and presentational mark-up (e.g., B, I elements). Although XHTML[12] and CSS[13] are promising approaches to separate presentation of the page from its content and logical structure, the available HTML tags support the description of structures of limited complexity. Therefore, in order to create complex structures, designers still continue to use presentational tags to achieve their design goal. Our authoring concepts can be considered as extending existing HTML elements or the vocabulary used for structural mark-up, to support better logical structuring of Web pages.

Fundamentally, the concepts in this part of the ontology refer to the basic constructs such as header, summary, abstract, footer, index, etc. According to McKnight *et al.* [77], these generic structures would be seen as landmarks that provide readers with information about where they are in a text, just as signposts, building and street names aid navigation in physical environment. These constructs are also important spatial cues that are not only important for *reading* documents successfully but also for supporting the notion of *using* the text [118]. With these concepts, we aim to describe the logical organisation of a Web page, by abstracting from the physical organisation. The physical components of a Web page would be a set of images (IMG), links (A), etc., but its logical components would be a header, a siteMap, etc.

[7] See `http://www.w3.org/TR/html4/`.
[8] See `http://www.latex-project.org/`.
[9] See `http://www.w3.org/XML/`.
[10] See `http://www.w3.org/MarkUp/SGML/`.
[11] See `http://www.w3.org/MarkUp/historical` for more information.
[12] See `http://www.w3.org/TR/xhtml1/`.
[13] See `http://www.w3.org/Style/CSS/`.

The upper level concepts of this part of the ontology, which are considered as basic units of overall structural organisation are: `Atom`, `Chunk`, `Node` and `Collection`. These concepts form the top-level categories of the *is-a* hierarchy. They also aim to support easy integration of new ontologies to the WAfA ontology. They can be considered as foundational concepts as in DOLCE [43] that can be used as a starting point to extend this ontology. The definitions of these concepts along with their OWL concrete abstract representation[14] are as follows[15]:

Atom A coherent object that cannot be logically decomposed. Physically we might be able to divide the object but the divided parts cannot be coherent and presented on their own (e.g., `Link`, `Caption`, `Footnote`, `Logo`, `Advertisement`, etc.). For example, an advertisement can be a composition of several images, but in fact they all represent only one advertisement. In the literature, they are also referred as basic objects [25].

Chunk Several objects grouped together to form a coherent unit (e.g., `Header`, `Footer`, `Section`, `Abstract`, `LinkMenu`, `SiteMap`, etc.). As pointed out by Miller [78], *chunking* is a quite natural way of presenting information and such units are important to preserve the content integrality and ease the use and recall of information. In the literature, they are also referred as fragments [58], semantic textual units [21], blocks [25, 79], composite objects [24], units or nodes [77].

```
Class(wafa:Chunk partial AuthoringConcept
 intersectionOf(
   restriction(wafa:has-direct-part
     someValuesFrom(unionOf(wafa:Chunk wafa:Atom)))
   restriction(wafa:has-direct-part
     allValuesFrom(unionOf(wafa:Chunk wafa:Atom)))))
```

This represents that a `Chunk` should have at least one part that is either `Chunk` or `Atom` and if it has a direct part then it has to be either `Chunk` or `Atom`.

Node A composition of atom(s) and chunk(s) to form a meaningful group which we use to represent a Web page.

[14] See `http://owl.man.ac.uk/2003/concrete/`.
[15] The prefix "wafa" represents the URI of the WAfA ontology.

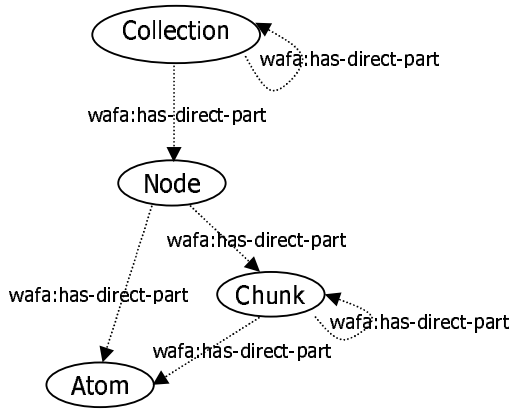


Figure 4.5: Relationships between top-level concepts of Authoring Semantics.

```
Class(wafa:Node partial AuthoringConcept
 intersectionOf(
   restriction(wafa:has-direct-part
     someValuesFrom(unionOf(wafa:Chunk wafa:Atom)))
   restriction(wafa:has-direct-part
     allValuesFrom(unionOf(wafa:Chunk wafa:Atom)))
   restriction(wafa:has-title allValuesFrom (wafa:Title))))
```

This represents that a `Node` should have at least one part that is either `Chunk` or `Atom` and if it has a direct part then it has to be either `Chunk` or `Atom`.

Collection A collection of nodes, meant to represent a Web site.

```
Class(wafa:Collection partial AuthoringConcept
 intersectionOf(
   restriction(wafa:has-direct-part
     someValuesFrom(unionOf(wafa:Node wafa:Collection)))
   restriction(wafa:has-direct-part
     allValuesFrom(unionOf(wafa:Node wafa:Collection)))))
```

This represents that a `Collection` should have at least one part that is either `Node` or `Collection` and if it has a direct part then it has to be either `Node` or `Collection`.

Figure 4.5 illustrates the relationships between these top-level concepts. *has-direct-part* property enabled us to create restrictions about the direct parts of these concepts.

For example, a `Collection` (a site) *has-direct-part* a `Node` (a page) but it cannot have a `Chunk` as its direct part. However since we have the *has-part* transitive property which is the parent property of *has-direct-part*, we can say that a `Collection` *has-part* a `Node`, and a `Node` *has-part* a `Chunk` therefore a `Collection` *has-part* a `Chunk`. Based on these top-level concepts, some other example restrictions are as follows:

(1) DisjointClasses(wafa:`Chunk` wafa:`Atom`)
(2) DisjointClasses(wafa:`Chunk` wafa:`Node`)
(3) DisjointClasses(wafa:`Node` wafa:`Atom`)
(4) DisjointClasses(wafa:`Node` wafa:`Chunk`)
(5) DisjointClasses(wafa:`Node` wafa:`Collection`)
(6) DisjointClasses(wafa:`Chunk` wafa:`Collection`)
(7) DisjointClasses(wafa:`Atom` wafa:`Collection`)
(8) ObjectProperty(wafa:*has-part* transitive inverseOf(wafa:*is-part-of*))
(9) ObjectProperty(wafa:*is-part-of* transitive inverseOf(wafa:*has-part*))

(1-7) represents that `Atom`, `Chunk`, `Node` and `Collection` are disjoint classes; an object cannot be a `Chunk`, `Atom`, `Node` and `Collection` at the same time. (4) and (5) are used to describe *has-part* and *is-part-of* properties which are transitive[16].

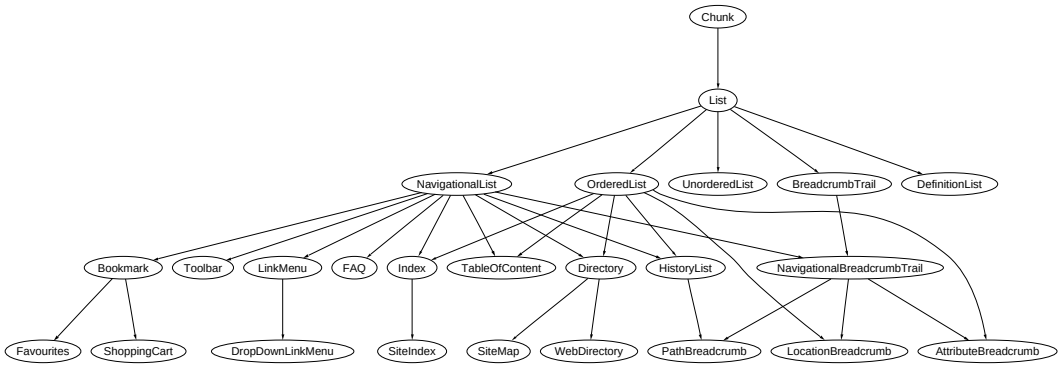


Figure 4.6: The specialisation of the concept "List".

Another example concept from this part of the ontology is the `List` concept. We particularly focus on functions and presentations of different kinds of lists and form the *is-a* hierarchy in Figure 4.6. We also have an `Order` concept which has subconcepts: `AlphabeticalOrder`, `HierarchicalOrder`, `NumericalOrder`,

[16]For detailed information, refer to `http://www.w3.org/TR/owl-features/`.

`SpatialOrder` and `TemporalOrder`. These ordering concepts are used to provide knowledge about different kinds of lists we have (see Figure 4.6). Some class definitions of the sub-concepts of the `list` are as follows:

```
(1) Class(wafa:List partial wafa:Chunk)
(2) Class(wafa:LinkMenu partial  wafa:NavigationalList)
(3) Class(wafa:NavigationalList partial wafa:List
      restriction(wafa:has-part someValuesFrom(wafa:Link)))
(4) Class(wafa:SiteMap partial wafa:Directory)
      IntersectionOf(
        restriction(wafa:has-order
          allValuesFrom(wafa:HierarchicalOrder)
        restriction(wafa:supports-movement
          allValuesFrom(wafa:CollectionWide)))))
(5) ObjectProperty(wafa:has-order range(wafa:Order))
(6) ObjectProperty(wafa:supports-movement)
```

(1) represents that a `List` is a kind of `Chunk` that has parts which can be `Atom` or `Chunk` but not `Collection` (for the definition of `Chunk`, see Page 80). Similarly, (2) represents that a `LinkMenu` is a kind of `NavigationalList` which has at least one `Link` as its part represented with (3). (4) represents that a `SiteMap` is a kind of `Directory` that has a hierarchical order and supports a collection wide movement (e.g., a Web site). (5) and (6) are used to define *has-order* and *supports-movement* properties.

Some particular concepts in this part of the ontology are also explained by using the home page of RoyalMail. Figure 4.1 highlights these objects and Table 4.2 provides documentation and hierarchical information. Descriptions of all of the concepts in this part of the ontology can be found in Appendix B.

4.3.2 Mobility Semantics for Navigational Structuring

This part of the ontology holds information about the *travel objects* based on the model of travel. The travel objects are mainly grouped into three: *way points*, *orientation points* and *travel assistants*. Objects might have a specific role in an environment, referred to as `EnvironmentRole`, and based on the context, they might have another journey role, referred to as `JourneyRole` in the ontology (see Table 4.3). Therefore, besides the travel objects we also have concepts that are about the journey roles of the

No	Authoring Concepts	Documentation	is a kind of	Mobility Concepts
1	`Header`	Is printed at the top of a page and can include a company logo, the page title, etc.	`Chunk`	`WayEdge`
2	`Logo`	An emblem or a device used to identify the page or a site.	`Special Graphic`	`Reference Point`
3,4,8	`LinkMenu`	A list of links meant to represent a menu.	`Navigational List`	`Navigation Point`
5	`SearchEngine`	Consists of a label, an edit box and a button.	`Chunk`	`Information Point`
6	`Breadcrumb Trail`	Representation of how and where information is located within a Web site.	`List`	`Location & Information Point`
10	`PageTitle`	The title of a page.	`Title`	`Identification Point`
7,22	`Sidebar`	A short, auxiliary information that is usually presented alongside the main content.	`Chunk`	`WayEdge`
12	`Heading`	Indicates what the part of the page below is about.	`Atom`	`Identification Point`
13	`Referential Link`	Provides a link between an item of information and an elaboration of that information.	`Link`	`NavigationPoint`
9,11,14,15 16,17,18,19	`Chunk`	Several objects grouped together to form a coherent unit.	`Authoring Concept`	`WayPoint`
23	`Advertisement`	A chunk that is advertisement, usually animated and graphic.	`Atom`	`Attention`
20	`Footer`	Is printed at the bottom of a page and can include copyright information, a list of links, etc.	`Chunk`	`WayEdge`
21	`Copyright`	Is a note about the copyright and is positioned at the bottom.	`Footnote`	`WayPoint`

Table 4.2: Example travel objects on the home page of Royal Mail.

objects. An object can be either an `Obstacle` or `Cue`. An `Obstacle` is an object that directly or indirectly obstructs the progress of a traveller to a specific destination and a `Cue` is an object that orientates and encourages onward navigation [56]. The journey role is context dependent, for example a graphic site map could be a cue to a sighted user but it could be an obstacle to a visually impaired user.

4.3.3 Contextual Semantics

The concepts in this part of the ontology aim to encode contextual information about a typical journey. Web users are people that have different skills, habits, motivations, intelligence, intentions, etc., that they bring to the computer when using the Web. Its highly possible that potential users will come from all walks of life and age groups [77]. Obviously, all these issues affect the experience of using the Web. We do not all interact with the text in the same way. The five things that are important for the journey experience are: (a) the current travel purpose, (b) the user, (c) the presentation form, (d) the timeliness of access and feedback, and (d) the user agent.

JourneyRole	
Cue	Orientates or encourages onward navigation.
Obstacle	Obstructs the progress of a traveller to a specific destination.
NonExistObstruction	The traveller is hindered in the completion of a successful journey because of a missing object.
ExistObstruction	The traveller is hindered in the completion of a successful journey because of existence of an object.
OutOfView	
OutOfRange	Is below the current viewable area and the objects in the viewable area do not lead onto or suggest the presence of such an object.
Invisible	Is not in the appropriate format or the environment does not provide the appropriate instrument to access them.
EnvironmentalRole	
WayPoint	
WayEdge	Linear or continuous element that acts as a boundary between two areas.
ReferencePoint	A unique and memorable feature of the environment.
DecisionPoint	A choice point where alternative paths of travel are possible.
IdentificationPoint	Provides elementary state description of a location and is usually perceived when the destination is reached.
Attention	Attracts traveller's attention and may change the traveller's focus.
Alert	An object that alerts the traveller to a change in the environment or control of the journey.
NavigationPoint	Provides a possible route and the traveller exercises some control by choosing to follow or not to follow it.
OrientationPoint	
ReferencePoint	A unique and memorable feature of the environment.
Distance	Provides distance information that may indicate distance from the traveller's starting position or from the traveller's destination.
Direction	Provides directional information that is essential to the navigator's ability to remain orientated within the environment.
Location	Provides location or position information explicitly.
TravelAssistants	
InformationPoint	Is a medium from which a traveller can directly request of information.
TravelAid	Provides an overview of the environment.
TravelSupport	Supports a journey that a traveller is actually being guided throughout.
TravelMemory	Holds information about where the traveller has been and provides means to get back there.

Table 4.3: Mobility semantics.

This sub-ontology aims to address these issues, by particularly focusing on the purpose; this could range from the travellers' purpose (e.g., information seeking, surveying, orientation, navigation, browsing, scanning, etc.) to the travel objects' intended purpose which is in fact the designers' purpose (e.g., AidsNavigation, AidsOrientation). Section B.2 in Appendix B presents this sub-ontology in detail.

One of the possible roles that *contextual semantics* could fulfill would be to obtain enough knowledge about the traveller's purpose and to transform pages accordingly. For example, if the traveller wants to scan a page, we could try to provide an overview of the page or if he (she) wants to orientate himself (herself) in the environment (wants to learn where he (she) is in the environment) we could provide objects that support orientational information such as a title, logo, etc. Travel objects can also play different roles in different contexts, for example, for a visually impaired user, a graphic can be an

obstacle in the context of information searching but a cue in the context of orientation.

The main problem with this contextual information is that it is difficult to obtain. Typically, the traveller's purpose is not explicitly specified (or well-defined) and also the traveller can engage in many different purposes as (s)he travels through the environment. Therefore, we are not currently using this part of the ontology in Dante. However, in the future work, we are planning to incorporate it in the transformation process which will enable us to address the user requirements better.

4.4 Summary and Conclusions

This chapter has introduced the WAfA ontology that aims to provide a vocabulary to define navigational and structural properties of Web pages. One part of the ontology encodes knowledge about the objects that support movement around Web pages (navigational properties) and another part encapsulates a vocabulary that refers to hypermedia authoring literature [72], transcoding literature [24, 58], mark-up languages [115], Web accessibility work [87, 109], voice interfaces [91], CSS cohort[17], etc. (structural properties).

The main reason for creating this ontology is to encapsulate the knowledge about Web pages in a computationally available form. The ontology can be used to make properties of pages explicitly available to technologies that cannot see and access the visual properties. The following chapter explains Dante which encodes a pipeline approach driven by this ontology. The pipeline accepts Web pages, annotates visual objects with the knowledge encoded in the ontology, and transforms pages accordingly to enhance the mobility support for visually impaired users. The extensive knowledge encoded in the ontology enables Dante to transform a Web page in a way that the visual objects on that page can play their intended roles in a non-visual presentation (e.g., audio).

The WAfA ontology was designed for providing machinery support and plays a central role in Dante. Therefore, apart from the technical evaluations such as determining its consistency, completeness, redundancy [47], it would be insignificant to evaluate WAfA on its own. In this thesis, the evaluation of WAfA is done pragmatically within Dante by assessing its competency to satisfy the requirements of the knowledge-driven pipeline encoded in Dante. WAfA was iteratively and incrementally implemented, deployed, and evaluated until it was in a satisfactory form to support

[17] See `http://www.w3.org/Style/CSS/`.

annotation and transformation processes in Dante. Hence it was evaluated throughout the entire process. The details of the technical evaluation can be found in Chapter 6.

Although the main reason for creating this ontology is to improve the mobility of visually impaired users around the Web, the WAfA ontology provides a foundational model that can also be used in any other work that requires an understanding of the Web environment. For example, it can be used as a vocabulary for a markup language or for creating (or presenting) Web pages for small screen device users or as an aid for understanding the meaning of the content of Web pages [125].

Chapter 5

Dante: Annotating and Transforming Web Pages

This chapter presents Dante which is a semi-automated tool for the support of travel and mobility for visually impaired Web users. The main goal of Dante is to identify travel objects on the Web pages, discover their role(s), annotate them with the WAfA ontology and transform pages accordingly to enhance the provided mobility support. Dante uses a pipeline approach to perform the annotations. The aim is to make the travel objects and their roles explicit. Although annotations make these objects explicit, the relationship(s) between them are still implicit. Therefore, an RDF Editor is introduced to create relationship(s) between the annotated objects. Dante then uses the annotations and their relationships to transform pages by using different techniques.

5.1 Dante Approach and its Architecture

Dante is built upon the Travel Analysis Framework presented in Chapter 3. It is an experimental prototype based on Mozilla [1], which can be considered as a "proof of concept" tool. It aims to demonstrate that if the structural and navigational properties of Web pages are made computationally available, a tool can then use such information to transform pages in different ways which will provide better accessibility support for visually impaired users.

The idea behind Dante is to analyse Web pages in order to extract travel objects and annotate them with terms from the WAfA ontology. The annotations, which are a way of associating knowledge encoded in the ontology with these objects, guide the

[1] see `http://www.mozilla.org/`.

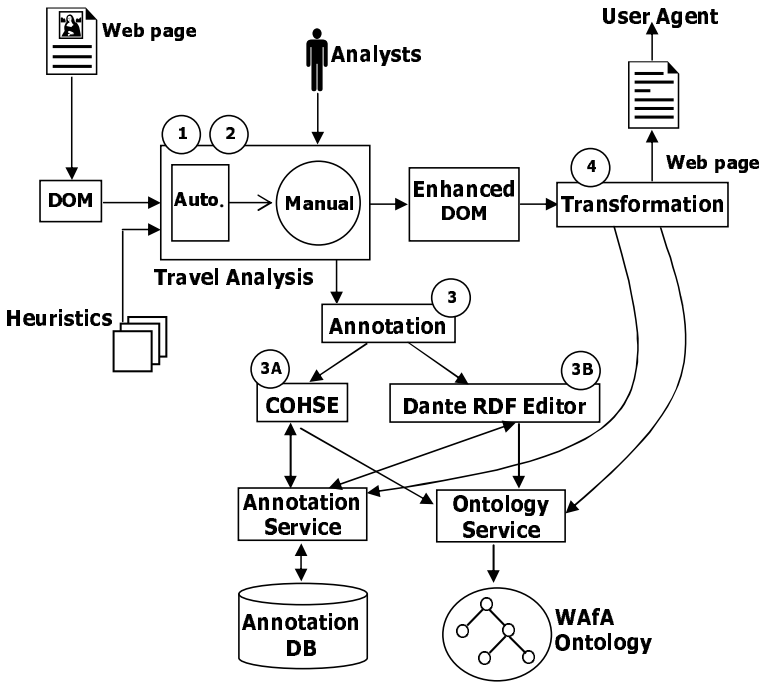


Figure 5.1: The basic architecture of Dante.

transformation of Web pages to enhance travel and mobility. For the transformations, Semantic Web technologies are used. However, unlike other examples[2] [51], a Web page is not annotated to convey the meaning, but to support mobility and convey information about the page itself including its structural and navigational properties. The architecture of Dante is depicted in Figure 5.1 which can be summarised as follows[3]:

1. Identifying and extracting travel objects;
2. Discovering their roles – navigational and structural properties;
3. Annotating the extracted objects by using the WAfA ontology (3A) and creating relationships between these annotations (3B);
4. Transforming the page with respect to these annotations and their relationship(s).

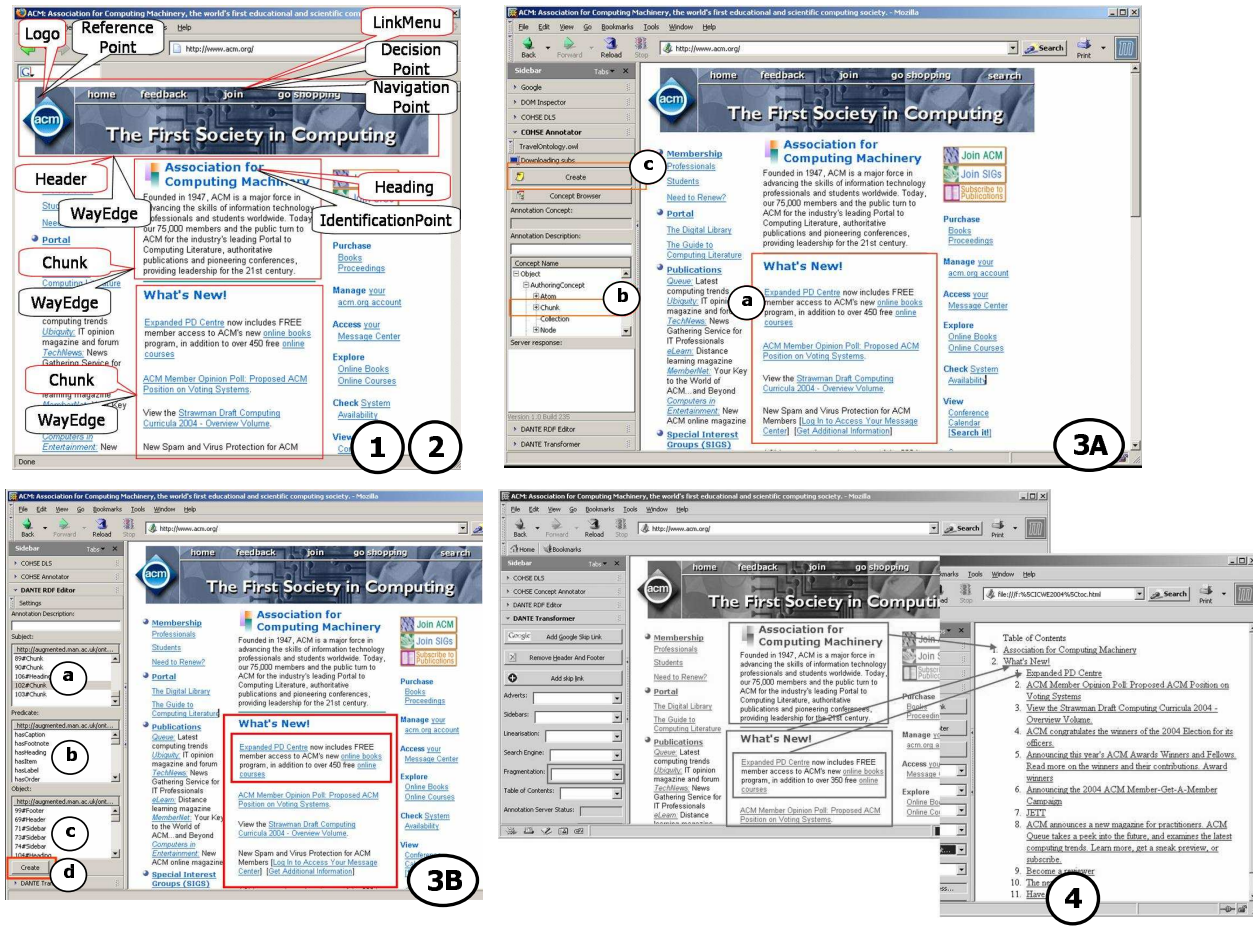
This knowledge-driven pipeline is implemented as a number of plug-ins (sidebars) to Mozilla. Mozilla is not only a browser but provides a platform for building applications [17]. In order to develop Dante plug-ins, we used the following technologies required by that platform: XUL (XML-based User-interface Language) to create the

[2]See `http://annotation.semanticweb.org`.
[3]The list numbers below represent the components number indicated in Figure 5.1.

structure and content of the user-interface, Javascript as a programming language to create the functionality, CSS (Cascading Style Sheets) to create the look and feel, XPCOM/XPConnect (Mozilla's component object model) to access C/C++ libraries from Javascript, XPInstall (Cross-Platform Install) to package the application to provide easy installation on any platform, and Gecko which is Mozilla's rendering engine to access the DOM (Document Object Model) representation of a page.

Figure 5.2 shows how the home page of ACM, which is also used as an example in Chapter 1, is transformed by Dante. This figure aims to provide an overview of the Dante pipeline. The first stage is to analyse the page by using the Travel analysis framework. In Figure 5.2 (1,2) objects such as `Logo`, `Header`, `Heading`, `LinkMenu` and `Chunk` are identified which are Authoring semantics in WAfA. The figure (1,2) also shows how these objects map to the Mobility semantics. The next stage in the pipeline is the annotation of the identified objects. In order to do that the COHSE[4] annotator is used with the WAfA ontology (Figure 5.2, 3A). COHSE stores annotations externally and then the transformation component uses these external annotations to do the transformations. Even though the annotations make the travel objects explicitly available, the relationship(s) between these objects is still missing. The Dante RDF Editor is used to create such relationship(s) (Figure 5.2, 3B). For example, a *has-heading* relationship is created between the top `Chunk` and the identified `Heading` object. In the final stage, the Dante transformer uses the annotations performed by the COHSE annotator and the relationships created by the RDF Editor to fragment the page into several pages and create a table of contents to link these fragments (Figure 5.2, 4). The following sections discuss the pipeline in detail.

[4]Conceptual Open Hypermedia Service (COHSE) (`http://cohse.man.ac.uk`).



Dante pipeline: (1,2) applying the Travel Analysis Framework, (3A) annotating with the COHSE annnotator, (3B) Creating relationships between annotations by using the Dante RDF Editor and (4) fragmenting the page into a number of pages by using the Dante transformer.

Figure 5.2: The home page of ACM transformed by the Dante pipeline.

5.2 Annotating Web Pages

Semantic annotation is one of the core technologies for the Semantic Web. It is the process of semantically enriching Web pages by adding metadata so that they are not only understandable by humans but also by machines. Fundamentally, there are two main Semantic annotation approaches: manual and automatic. Manual annotation approaches allow users to define annotations by hand. Automatic annotation approaches focus on extracting annotations automatically by using different techniques such as machine learning, natural language processing techniques, etc., [51]. We surveyed annotation tools and techniques to find out if we can use an existing tool to do the annotations in Dante[5]. However, our survey revealed that most of the existing tools and techniques focus on the annotation of the content rather than the structure of the pages.

Some tools are available to annotate pages when they are designed and created (e.g., SMORE [6]). Although this is an important approach, there are a huge number of existing Web pages, so being able to annotate them is crucial. Moreover, we wanted to demonstrate the applicability of the Dante approach before creating pages automatically annotated with the WAfA ontology. The OntoMat-Annotizer [7] is a tool that can be used with existing pages but it injects the annotations to the page. Since most of the time, Dante would work with third party pages, it is important to be able to keep annotations externally. Another project is Annotea [8] which is a W3C project that allows structured standoff RDF annotation of Web pages. RDF is a metadata framework which has a simple structure. It has three main components: the subject, predicate and object which make an RDF statement or triple. The Annotea project uses the RDF framework to create statements about part of a page [67]. For example, with Annotea a simple annotation can be created that points to this paragraph (assuming that it is an HTML document and using XML Pointer Language (XPointer[9]) which says "I do not understand what you are talking about!" (see Figure 5.3). Although Annotea provides flexibility of addressing parts of a page, its main focus is on text annotation and does not have any link to ontologies. For the Dante approach, it was important to create annotations that are somehow *controlled* by a vocabulary (i.e., the WAfA ontology) so that the transformation component can access the annotations and transform pages.

COHSE supports ontology based annotation of Web pages [45]. As opposed to

[5]See `http://annotation.semanticweb.org/tools` for the list of annotation tools.
[6]See `http://owl.mindswap.org`.
[7]See `Seehttp://annotation.semanticweb.org/ontomat/`.
[8]See `http://www.w3.org/2001/Annotea/`.
[9]See `http://www.w3.org/TR/xptr/`.

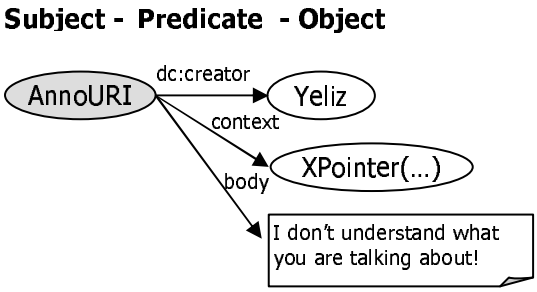


Figure 5.3: Example RDF triples of an Annotea annotation.

some other tools, COHSE keeps annotations externally. The COHSE annotator also uses XPointer expressions to identify a region of a document and is based on two services: an *Annotation service* which supports annotating resources and querying existing annotations, and an *Ontology service* which supplies operations relating to the content of an ontology (Figure 5.4). The COHSE annotator is implemented as a Mozilla sidebar which we used to do the annotations in Dante. It allows users to load an ontology and Figure 5.5 shows a screenshot of the sidebar when the WAfA ontology is loaded. Figure 5.6 and Table 5.1 show the objects identified on the home page of Java. In order to annotate these objects with the COHSE annotator (Figure 5.5), the process is as follows: First a part of the page is selected (1), then a concept is selected from the list of concepts of the loaded ontology (2) and then pressing the "create" button adds the annotation to the specified Annotation service which is a database at the back-end (3). The other components of Dante are mainly based on the extended architecture of COHSE, particularly the Annotation service. This was one of the reasons for choosing the COHSE annotator for creating annotations. It gives us the flexibility of re-using existing services and extending them to meet the requirements of Dante.

Although we are using the COHSE annotator, the underlying annotation approach

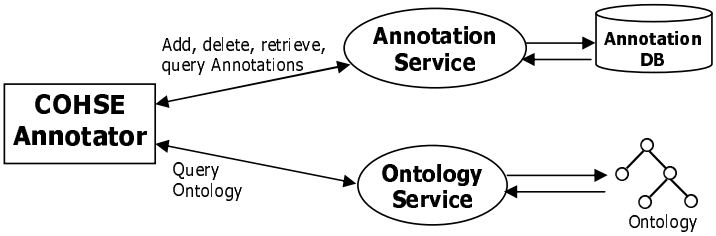


Figure 5.4: The architecture of the COHSE Annotator.

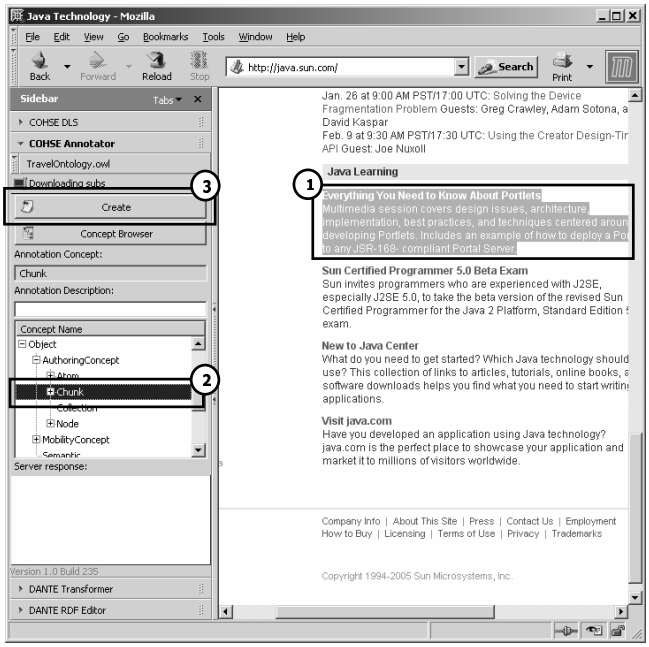


Figure 5.5: Annotating the home page of Java with the WAfA ontology.

of Dante is based on a flexible pipeline. Figure 5.7 illustrates this pipeline which can:

1. receive inputs from different sources and in many formats (e.g., DC[10], RSS[11], manual annotations done by using annotation tools such as COHSE, etc.);

2. harmonise these inputs into a canonical form by using Authoring semantics;

3. annotate pages automatically based on a heuristics by using the Authoring semantics;

4. translate between annotation vocabularies, such as from Authoring semantics to Mobility semantics, to provide extensive knowledge about these objects;

5. use such knowledge to transform Web pages into alternative forms to provide better travel support.

Fundamentally, the WAfA ontology provides a common, shared vocabulary that the annotation and the transformation components of this pipeline approach can communicate. The components numbered on Figure 5.7 relate to the main architecture of Dante

[10]Dublin Core Metadata, see `http://dublincore.org/`.
[11]See `http://blogs.law.harvard.edu/tech/rss`.

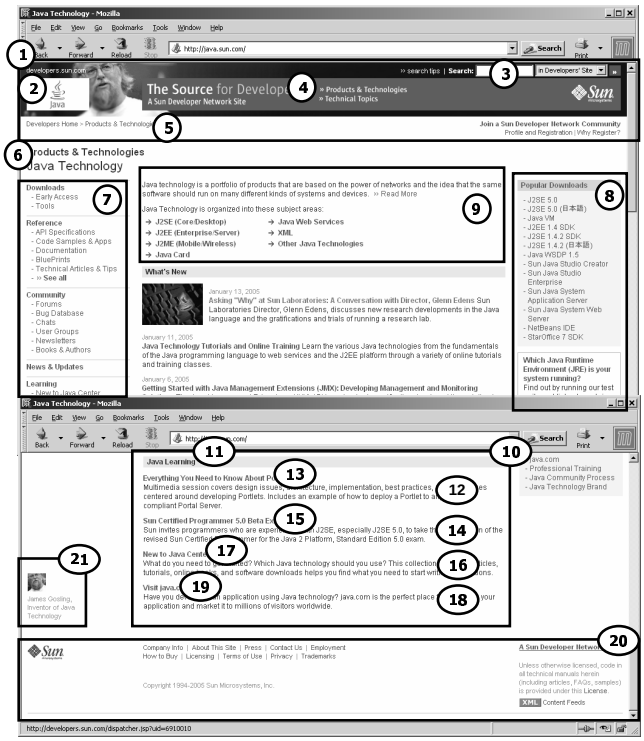


Figure 5.6: The home page of Java (15/01/2004).

No	Authoring Concept	Mobility Concept
1	Header	WayEdge
2	Logo	ReferencePoint
3	SearchEngine	InformationPoint
4	LinkMenu	Decision-NavigationPoint
5	BreadcrumbTrail	Location-InformationPoint
6	PageTitle	IdentificationPoint
7,8	Sidebar	WayEdge,Decision-NavigationPoint
9,10,12,14,16,18	Chunk	WayPoint
11,13,15,17,19	Heading	IdentificationPoint
21	Footnote	WayPoint
20	Footer	WayEdge

Table 5.1: Example annotations done on the home page of Java (see Figure 5.6).

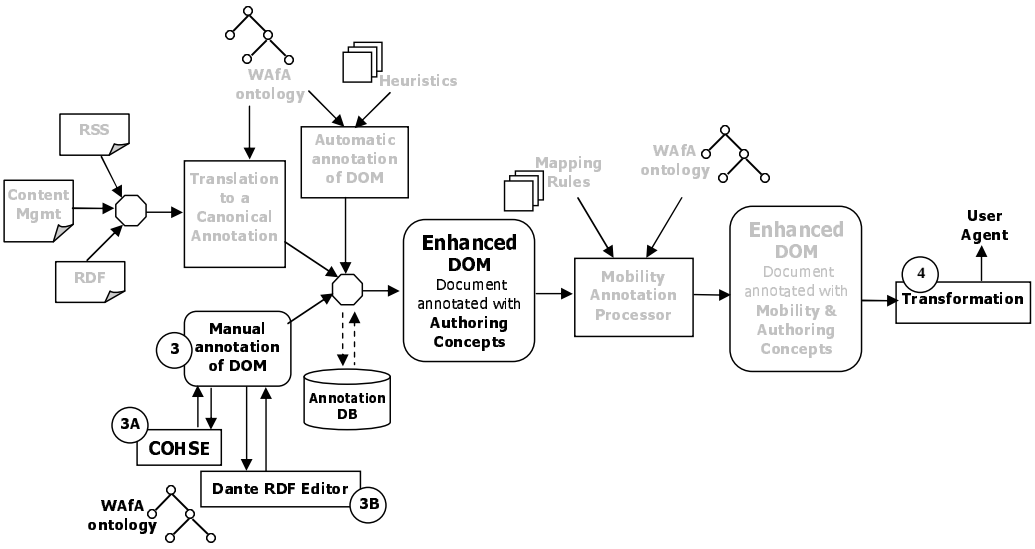


Figure 5.7: The Dante annotation pipeline (see Figure 5.1 for the parts 3 and 4).

illustrated in Figure 5.1. This pipeline approach hides how annotations are created from the transformation component. Therefore, from the transformation perspective, it becomes unimportant how the annotations were actually done; manual, automatic, generated by a content management system or encoded during the design process. In this thesis, we propose the manual annotation technique and Plessers *et al.* [95] present how WAfA annotations can be generated with a design methodology. Pages also do not need to be annotated both with Authoring and Mobility semantics. After Authoring concepts are acquired, a set of rules can be used to translate them to Mobility concepts in order to accumulate better knowledge about these objects. Detailed information regarding this process can be found in [124]. We can of course bypass the translations and use only the Authoring concepts to perform the transformations. In this case, direct translation will be done and the Mobility concepts will be implicitly used in the transformation techniques.

As a summary, this section presented the annotation component of Dante. This component is built upon a pipeline framework which allows annotations to arrive in different formats. In the prototype implementation, the COHSE annotator is used to annotate pages manually with the WAfA ontology and store them externally. The idea was first to demonstrate that the annotations with the WAfA can guide the transformation of Web pages before we actually focus on automating the annotation process. The annotations are then used by the RDF Editor to create a structural model of a page

which is explained in the next section.

5.3 The Dante RDF Editor: Modelling Web Pages

When a Web page is annotated with the WAfA ontology using the COHSE annotator, concepts are associated to parts of a Web page by using XPointers. Even though such kinds of annotations provide information about the structural characteristics of certain parts of the page, it does not provide any information about the relationship(s) between these parts which can be important. For example, in Figure 5.6, object 21 is annotated as a `Footnote`[12]. Even though this object is located at the bottom of the page, it is a `Footnote` created for the `Header`. However, when we only annotate that object as a `Footnote`, the fact that it is associated to the `Header` becomes implicit. Another example relationship is between object 10 and 12. Annotating these objects as `Chunk`s make them explicit, but again the fact that the object 12 is part of object 10 becomes implicit and computationally inaccessible. Therefore, the Dante RDF Editor is developed to make such kinds of knowledge accessible and enhance the expressiveness of the performed annotations.

The Dante RDF Editor uses the underlying architecture of COHSE by particularly extending the Annotation service (Figure 5.4). It enables users to create RDF triples to indicate the relationships between the annotations already stored by the COHSE annotator. COHSE assigns unique identifiers to each annotation. The Dante RDF Editor creates an RDF triple by using these unique identifiers in the subject and object fields and a property from the WAfA ontology as the predicate value. The Annotation service is extended and modified to store and retrieve such RDF statements. For example, the `Heading` and the `Chunk` objects, the following statement can be created: "object 10 which is a `Chunk` *has-heading* object 11 which is a `Heading`" (see Figure 5.6). In this RDF triple, object 10 is the "subject", *has-heading* is the "predicate" and the object 11 is the "object".

We have already discussed some annotation tools and approaches in the previous section. All these approaches mainly focus on annotating content of Web pages and

[12]Different typographies used in this chapter have the following associations: Names of concepts represented in the ontology are in `Courier New font`; relationships between concepts are in italics enclosed by hyphens, e.g., *part-of*.

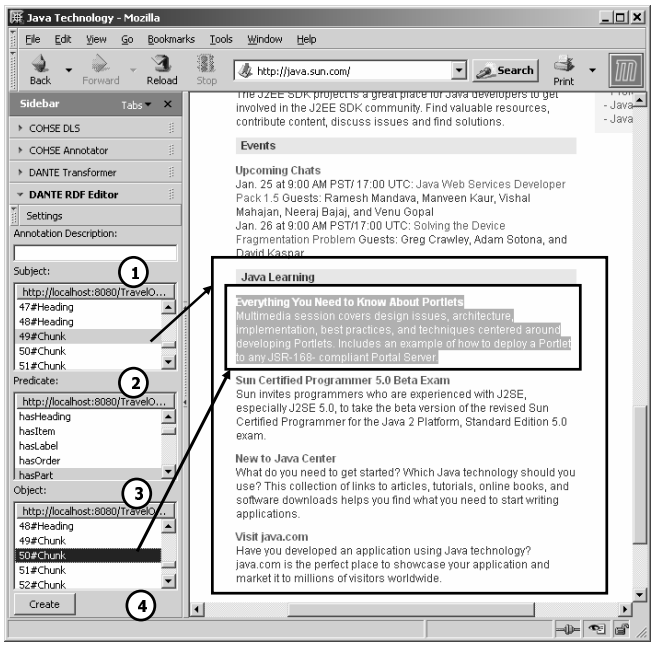


Figure 5.8: The Dante RDF Editor creating relationship(s) between annotations.

none of them discusses how to create relationship(s) between the performed annotations. Although there exist numerous RDF editors such as RDFedt [13] and RIC [14] that allow users to build complex RDF documents with triples, the focus of these tools is to create an ontology or stand-alone RDF documents. In our case, the aim is to annotate structural components of a page and their relationship(s).

Figure 5.8 shows the RDF Editor which is implemented as a sidebar. The object and subject fields list the annotations already done by the COHSE annotator. The predicate field is populated by the properties encoded in the ontology. The user can then choose a subject (1), predicate (2) and object (3), and click the "create" button to add the annotations to the specified Annotation service (4). Table 5.2 shows some triples created by using the example annotations in Table 5.1, and Figure 5.9 illustrates the triples concerning the Header object. The generated triples provide us a model of the page as opposed to SGML style hierarchical structuring of documents. As Gatzemeier *et al.* [44] pointed out, the SGML applications require a primary hierarchy for structure definitions. However, this hierarchical representation does not always represent the main structure of the document. A graph (RDF statements) as the underlying data structure of a document can reflect a non-hierarchical, interconnected nature of a

[13] See `http://www.jan-winkler.de/dev/e_rdfe.htm`.
[14] See `http://www.mindswap.org/`.

Subject	**Predicate**	**Object**
1-`Header`	*has-part*	2-`Logo`
1-`Header`	*has-part*	3-`SearchEngine`
1-`Header`	*has-part*	4-`LinkMenu`
1-`Header`	*has-part*	5-`BreadcrumbTrail`
1-`Header`	*has-footnote*	21-`Footnote`
10-`Chunk`	*has-heading*	11-`Heading`
12-`Chunk`	*has-heading*	13-`Heading`
14-`Chunk`	*has-heading*	15-`Heading`
16-`Chunk`	*has-heading*	17-`Heading`
18-`Chunk`	*has-heading*	19-`Heading`
10-`Chunk`	*has-part*	12-`Chunk`
10-`Chunk`	*has-part*	14-`Chunk`
10-`Chunk`	*has-part*	16-`Chunk`
10-`Chunk`	*has-part*	18-`Chunk`

Table 5.2: Example relationships on the home page of Java (Table 5.1).

document.

5.4 Transforming Web Pages

The transformation component of Dante is also implemented as a Mozilla sidebar. In order to reformulate the structure of the page by different techniques, it uses the model which is created by the COHSE annotator and the Dante RDF Editor. This is an experimental sidebar that allows easy deployment and testing of new heuristics. In the current implementation, each heuristic is associated to a button or a drop-down menu on the sidebar. Depending on the heuristic, this user interface can easily be extended by using the XUL (Mozilla's User Interface Language) components.

In this section, we continue using the home page of Java as an example to demonstrate the transformation heuristics encoded by Dante (Figure 5.6). This page is used because it provides good illustrations of some of the issues concerning the mobility support provided by the page. The page is annotated with the Authoring concepts. In [124], we propose that translating Authoring concepts to Mobility concepts can provide an enhanced set of annotations to perform transformations. But after experimenting with a number of Web pages and different sets of annotations, we realised that Mobility concepts are actually used to classify the identified Authoring concepts.

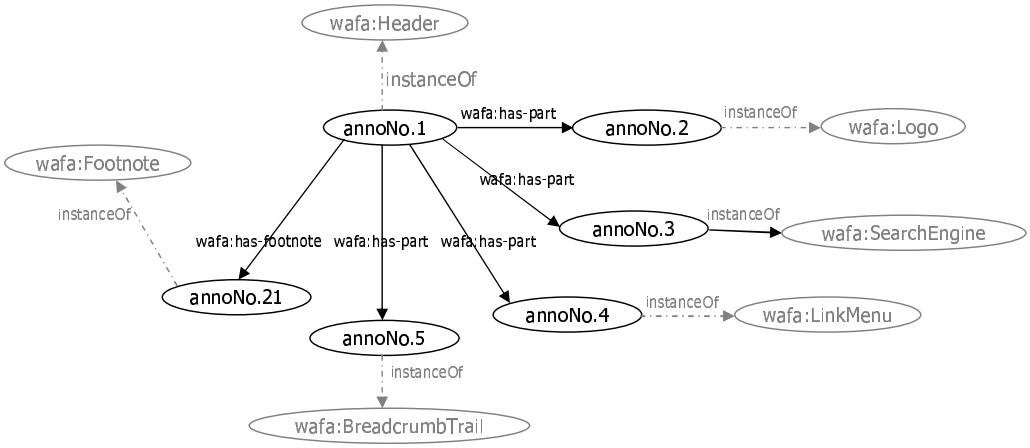


Figure 5.9: RDF Triples concerning the “Header” object on Figure 5.6.

But we did not need to use rules as these groupings were always the same. Therefore, depending on the heuristic, once we know Authoring concepts, they can easily be mapped to Mobility concepts that guide the transformations. For example, a `Heading` is an `Identification Point` and in all transformations it is treated so. Table 5.1 shows some more example mappings between the Authoring concepts identified on the home page of Java to Mobility concepts.

We are proposing different heuristics to transform pages. The following sections present them in detail particularly by explaining the motivation and the objects used. Each technique addresses certain mobility and travel issues, particularly the mobility *principles* and the mobility *instruments* (Table 5.3). The Dante transformer sidebar can be seen on Figure 5.10. After a page is loaded in Mozilla, each technique can be executed by pressing a button or choosing it from a drop-down list. These techniques are grouped into two and can be summarised as follows:

Generic heuristics: Focus on reformulating a page as a whole rather than focusing on a specific object on that page (Section 5.4.1).

T1 *Fragmenting pages physically:* A Web page is fragmented into a number of smaller and simpler pages.

T2 *Fragmenting pages logically:* A table of contents is inserted into the page to provide an overview.

T3 *Eliminating repetitions:* Some structural components such as header, footer and sidebar are removed to provide a simpler page.

T4 *Enhancing intra-page movement:* A skip link is added to skip certain elements in the page to provide better movement support within the page.

T5 *Linearising pages:* The main content of the page is linearised to provide easier access to screen reader users.

Object specific heuristics: Focus on using specific kinds of objects on a page to reformulate its structure (Section 5.4.2).

T6 *Enabling movement to the information point:* Search Engine in the page is moved to either top or bottom of the page to provide better access.

T7 *Repositioning objects:* The positions of the objects such as advertisements are changed so that they do not distract the users.

T8 *Alternative views:* Only certain components such as link menus are presented in the page.

T9 *Structuring Link Lists:* Creates a list of links in the page and organises them by using the structure of the page.

5.4.1 Generic Heuristics

These techniques aim to reformulate the overall page structure in such a way to enhance the overall mobility support of that page.

Fragmenting Pages Physically

Dante fragments a complex and long page into a number of smaller and simpler pages. This technique is similar to the partitioning approach proposed by [21, 25, 127] for small screen devices. But in our approach, the focus is not to support the limitations of such devices, but to support better audio navigation.

In order to fragment a page, Dante uses `Chunk` (`WayPoint`) and `Heading` (`IdentificationPoint`) annotations and also the *has-part* and *has-heading* relationships created by the RDF Editor (Section 5.3). Physical fragmentation supports *regularity* because each fragment is designed in a similar manner. Fragmented pages

Mobility	**T1**	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**
Principles									
Regularity	x			x	x				
Information Flow	x	x	x			x	x	x	x
Granularity	x	x							
Egocentricity									
External Memory						x			
Familiarity	x	x	x	x		x			
Spatial awareness	x	x			x				
Instruments									
Preview	x	x		x					x
Probe				x		x			x
Scanning	x	x	x		x		x		
Browsing	x	x					x	x	x
Searching						x			
Overview	x	x							
Differentiating page elements	x	x						x	
Orientation	x	x							
Getting contextual information	x	x							x
Accessing the focus of the page				x					
Accessing Layout (organisation)	x	x							

Table 5.3: Analyses of transformation techniques with respect to mobility.

are smaller and more manageable compared to the original page so better *information flow* is supported. In order to enable navigation between fragments, a page with a table of contents is provided. Such a table of contents also helps users to get an *overview* of the page and easily move between different parts of the page (see Table 5.3). Moreover, screen readers tend to support micro-navigation; only a small part of the page is viewable at one point. Therefore, physical fragmentation also provides good support for such micro-navigation.

Different techniques are used to create the table of contents page and link the generated fragments. The home page of Java is used as an example to explain these techniques as follows:

- *Linear fragmentation* For each `Chunk` annotation, a separate page is created. Then a linear table of contents page is created to link these fragments. Each fragment includes a link back to the table of contents. Figure 5.10 shows how the home page of Java is fragmented based on this technique.

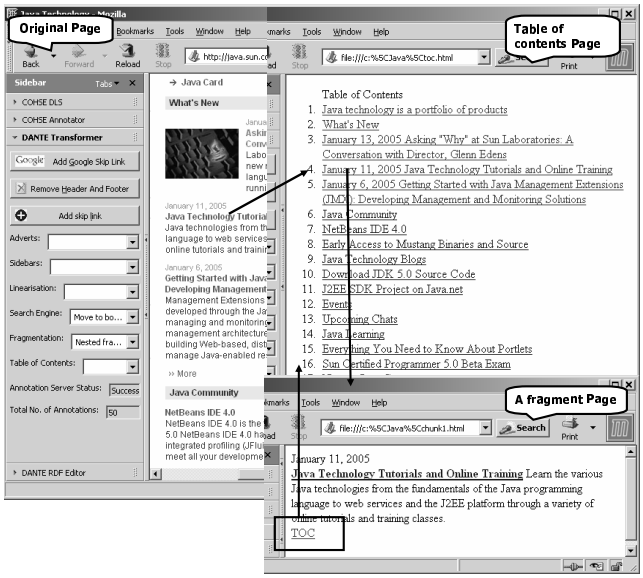


Figure 5.10: Fragmentation of the home page of Java with a linear table of contents.

- *Linear guided fragmentation* Compared to the previous technique, here each fragment is also linked to other fragments. Each fragment has three extra links: (a) a link to the previous fragment (previous), (b) a link to the next fragment (next) and (c) a link back to the table of contents page (TOC). Therefore, a kind of guided tour is created between the fragments [50]. Figure 5.11 shows the fragmentation of the home page of Java with this technique.

- *Nested fragmentation* The previous two techniques do not consider the *part-of* relationships between the chunks. Therefore, a linear table of contents is created. But in this technique, *part-of* relationships are used to create a nested table of contents. An enumerated list is used to create such a kind of table of contents. Therefore screen reader users can also easily access the parent-child relationships. Figure 5.12 shows the created table of contents concerning the *part-of* relationships.

Fragmenting Pages Logically

The fragmentation approach explained above physically divides the page into a set of pages. Similar to that approach, we also logically fragment the page. This time a TOC is inserted into the page itself that provides links to `Headings` (`Identification Point`) and `Chunks` (`WayPoint`) in the page and back to the TOC. This can be

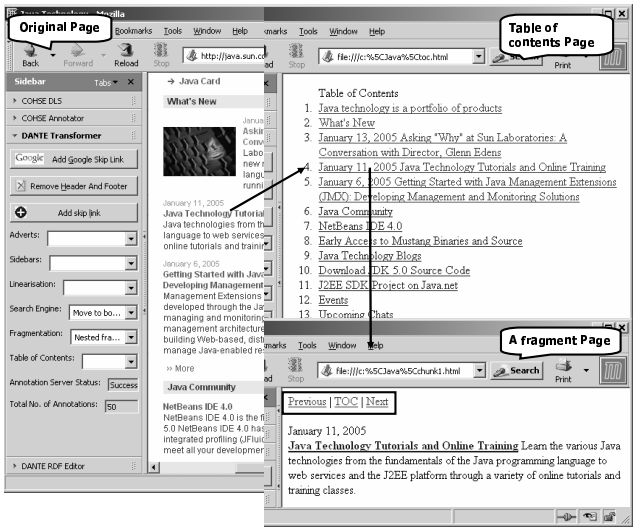


Figure 5.11: Fragmentation of the home page of Java with a linear table of contents (with guided tour support).

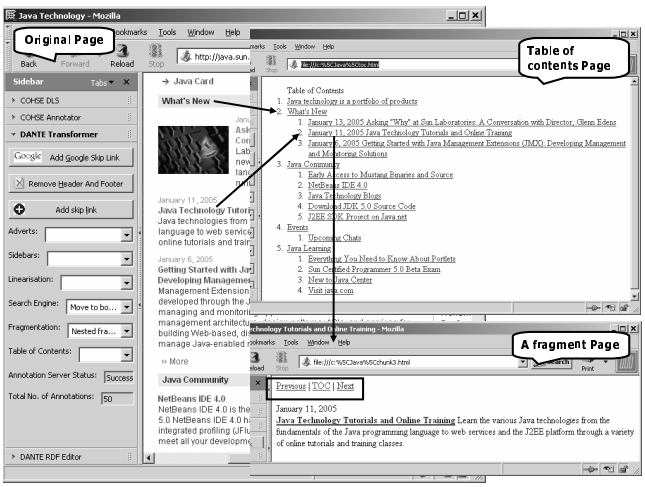


Figure 5.12: Fragmentation of the home page of Java with a nested table of contents.

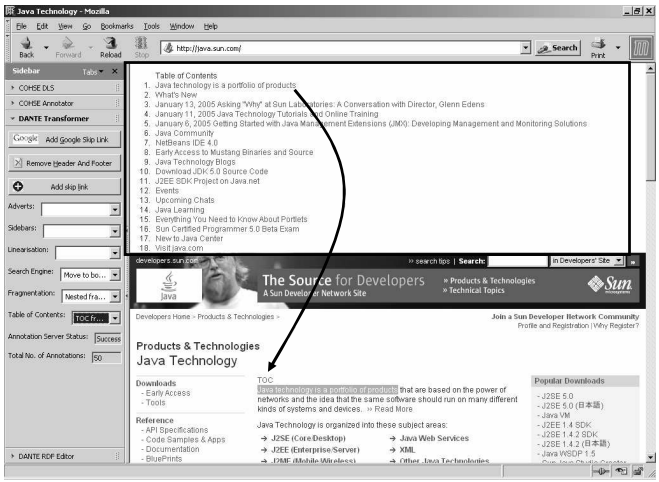


Figure 5.13: Logical fragmentation of the home page of Java.

considered as a way of providing the *bird's eye view* (overview) of the page. Based on the `Headings`s and `Chunk`s in the page, we logically fragment the page and allow users to have the preview of these logical fragments. Figure 5.13 shows how the home page of Java is logically fragmented. This is a technique to improve the intra (*within* the page) mobility support, but once we improve this, the inter (*between* the pages) and collection wide mobility support (within the site) are also improved. The created TOC can be inserted either at the top or bottom of the page.

Both fragmentation approaches (logical and physical) have pros and cons. For example, in the logical fragmentation, the user can continue to read the next chunk without returning to the TOC. However, the number of links in the page (from/ to TOC) can be too many and difficult for the user to manage. The extra added links can increase the cognitive demand. On the other hand, physically fragmented pages are simple but the user has to move between them in order to read the entire page.

Eliminating Repetitions

Some structures such as headers, footers and sidebars can easily become repetitive and not useful if the page is accessed more than once. They are kind of `WayEdges` that are usually used to differentiate the *"main"* content from the *"side"*. Sighted users tend not to read such constructs by skipping and directly focusing to the relevant part of the page. However, if you access a page with a screen reader, it is as if you have never been to that page and there is no function that enables you to skip such constructs in the

Figure 5.14: The home page of Java without header and footer.

page. Therefore the idea here is to remove header, footer and sidebars in a page so that a shorter and concise page is provided to the user (see Figure 5.14 and Figure 5.15).

Enhancing Intra-Page Movement

Skip links are popular for enhancing the navigation, and thus the mobility support provided by the page for visually impaired users. They are mainly used at the top of the page to provide a link to the main content, so that the user does not have to *read* the unnecessary information. In particular, they are used to avoid repetitions. Different strategies can be used to add skip links. Asakawa *et al.* [9] propose adding skip links to part of the page annotated as the main content. With the help of our WAfA ontology, we can have better understanding of the structure of the page. Therefore, we provide more flexibility in adding skip links.

We have three heuristics which mainly use the following three objects: (1) `Headline`, (2) `Sidebar` and (3) `Header`. If there is an object annotated as a `Headline` then we consider that object as one of the most important headings in the page so we add a skip link to easily jump to that heading. In order to avoid again repetition in the page, we add skip links to skip `Header` and the `Sidebar`. We have also assigned priorities to these heuristics, in this case we first check if there is a `Headline` annotation, then the `Header` and then the `Sidebar` annotations.

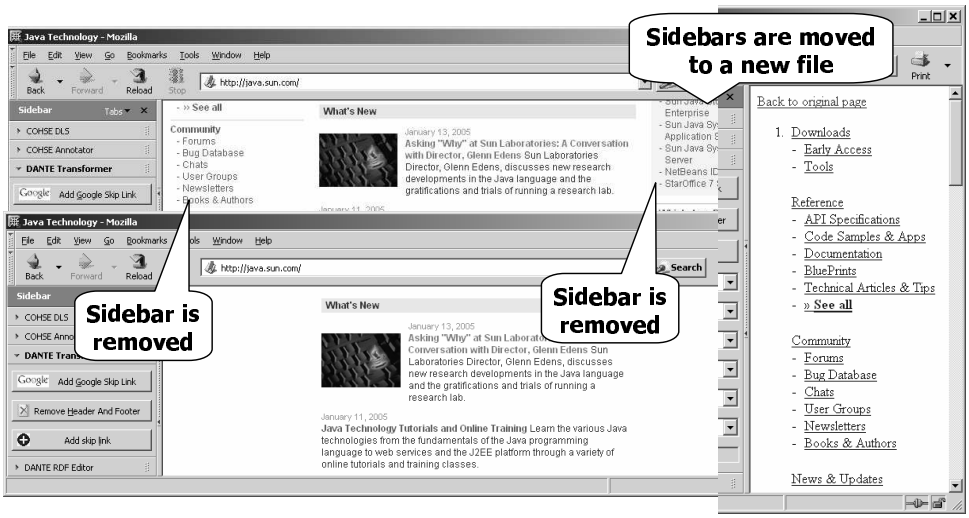


Figure 5.15: The home page of Java without sidebars.

Figure 5.16 shows how a skip link is added to the home page of Java.

Linearising Pages

When screen readers present pages in audio, they linearise the content of the page by following the source code. However, this might not always generate a proper rendering as the order in the source code might not show the logical structure of the page. Therefore, the focus here is to use the annotations to linearise the page content so that it can be better rendered with a screen reader.

Linearisation can be considered as a technique to eliminate the multi-dimensionality of pages. It is well-known that multi-dimensional structures (e.g., tables) challenges the linear interaction styles typically supported by screen readers [126]. For example, the visual rendering of the home page of Java looks like a three column table (see Figure 5.6). Therefore, the aim here is to linearise that kind of table-like structure properly. In order to do that, we again use `Chunk` and `Heading` annotations. The linearisation is supported in two ways: (i) the linearised content is inserted at the top of the page itself or (ii) the linearised content is inserted into a new page and a link is added back to the original page. Figure 5.17 shows how the home page of Java is linearised at the top of the page.

Figure 5.16: The home page of Java with a skip link.

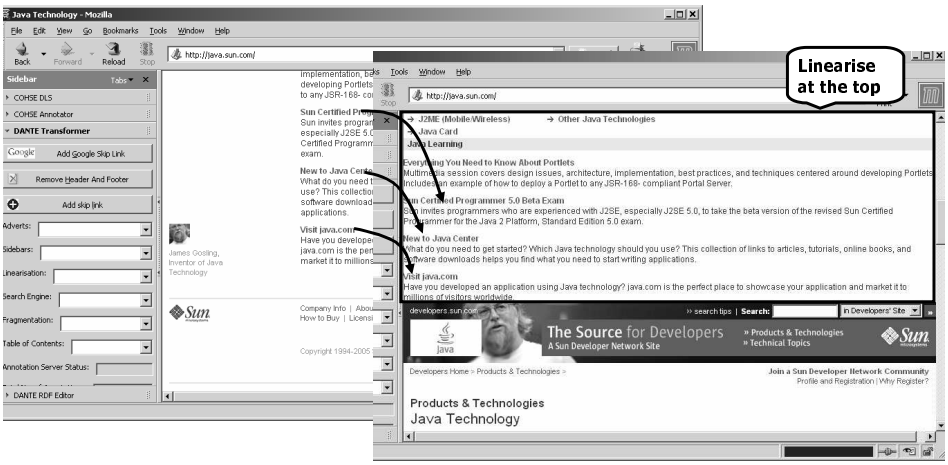


Figure 5.17: Linearisation of the main content of the home page of Java.

5.4.2 Object Specific Heuristics

As opposed to generic transformation techniques, these techniques focus on using certain kinds of objects on the page to enhance the overall mobility support.

Supporting Movement to the Information Point

During a Web journey, users can engage in different browsing activities. Depending on the current task, users use different strategies and objects in the environment. One of the objects that is widely used in case of a specific or well-defined task is the search box. A search box is a kind of `InformationPoint` that allows users to directly request information. The position of the search box can be quite important. Since sighted users can easily scan the page, the positioning might not be that important. But visually impaired users tend to read the page from top to bottom, so finding these kinds of objects cannot be easy. Some recent studies show that users have problems in finding search engines in the page and they mostly complain about the positioning of the object [33, 52]. They either think it is quite down in the page or it is in a wrong position. Therefore, we provide an option to change the positioning of the object annotated as `SearchEngine`. It can be either moved to the top of the page or to the bottom. When users need to use it, they can move it to the top of the page so that it can be accessed easily. It can also be moved to the bottom to avoid the repetition in the page. In Figure 5.18, the search engine which is at the top of the page is moved to the bottom.

Repositioning Objects

Objects that are primarily designed for visual communication can easily be distractive in a non-visual communication. A good example to these objects are advertisements (`AttentionObject`), in particular, when alternative texts are not provided and they advertise something that does not interest the user. For example, some users indicated that because of adverts it was difficult to reach the focus of the page [33]. On the other hand, these objects can support serendipitous browsing. Therefore, rather than removing them from the page, we provide two heuristics; (1) move them to the bottom of the page and (2) move them to a new file. Figure 5.19 shows the implementation of these heuristics.

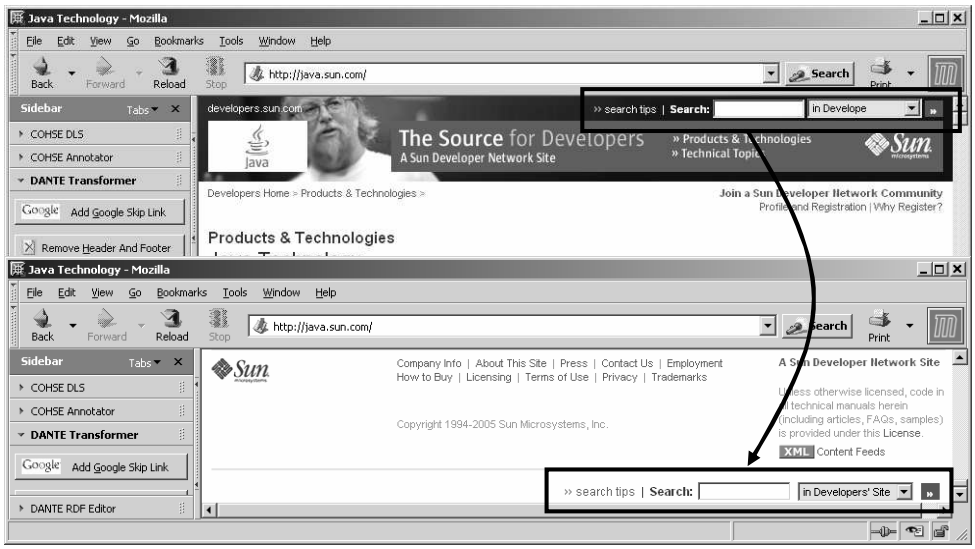


Figure 5.18: The search engine in the home page of Java is moved to the bottom.

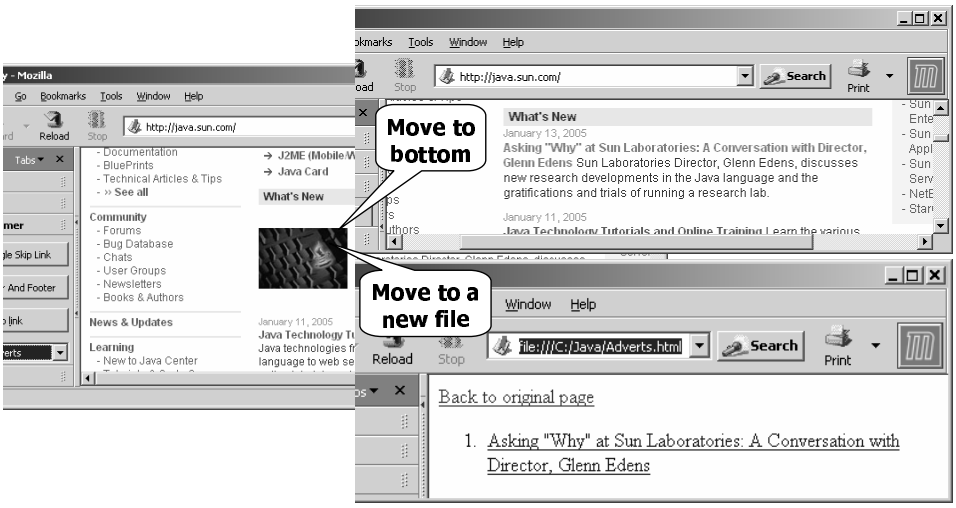


Figure 5.19: An advertisement is moved to the bottom of the page and to a new file.

Generating Alternative Views

This technique mainly uses different objects to provide alternative views. This enables users to access pages in different ways. For example, Dante uses `LinkMenu` (`Decision` and `NavigationPoint`) annotations to present only link menus in the page or as explained above only `Sidebar`s can be presented (Section 5.4.1). These alternative views can easily be extended based on the objects annotated in the page. For example, only `Header` or `Footer` or `Heading`s, etc., can be presented.

Structuring List of Links

Some screen readers provide a function for accessing the list of links in the page. They allow users to scan the page rather than read the entire page. However this technique requires links to have proper labels so that they make sense when they are read out of context, but unfortunately many links are context dependent. For instance, in the home page of Java, there are links labelled as "buy" or "see all" or "more". If they are not read with contextual information, it is almost impossible to understand where these links point to. However, the structural knowledge encoded with the annotations can be used to add context to the created list of links. This transformation technique aims to use `Chunk` and `Heading` annotations to provide structure and context to the generated list of links. This can be considered as grouping links (organising) which is well-known to aid scanning and memorability of the links [94].

5.4.3 Customisation of Heuristics

In some of the transformation heuristics, we already discussed certain options, for example being able to move the identified `SearchEngine` object to the top or to the bottom of the page. However, most of the other heuristics discussed above can also be systematically extended based on an *object-action* approach. Table 5.4 summarises basics of a framework that can be used to perform the transformations systematically. Certain actions can be performed on certain objects based on the given location parameter. For example, if a LinkMenu is identified, based on this customisation framework, possible actions can be: removing from the page, moving to the top of the page or moving to a new file. This framework provides a better and systematic customisation approach to our heuristics.

Object	**Action**	**Location**
`Header` (`WayEdge`)	move	to a new file
`Footer` (`WayEdge`)	remove	to top
`Sidebar` (`WayEdge`, `DecisionPoint`, `NavigationPoint`)	linearise	to bottom
`LinkMenu` (`DecisionPoint`, `NavigationPoint`)	fragment	.
`Chunk` (`WayEdge`)	add	.
`Headline` (`IdentificationPoint`)	duplicate	.
`Advertisement` (`AttentionObject`)	.	
`SearchEngine` (`InformationPoint`)	.	
`Link` (`NavigationPoint`)	.	
`Section` (`WayEdge`)		
`Heading` (`IdentificationPoint`)		
.		
.		
.		

Table 5.4: Customisation of some transformation heuristics.

5.4.4 Combination of Heuristics

In the current implementation of the Dante transformer, each transformation heuristic is applied separately. This gives us the flexibility of testing whether these heuristics can be combined (i.e., applied together) and if they can, whether the order of application is important or not. Table 5.5 aims to provide answers to these questions. This table also refers to some of the customisations supported by the heuristics as they might have a different impact on the combination with other heuristics. We will discuss two approaches for combining these heuristics.

In-situ same page Technically combining heuristics is a challenging issue as Dante uses annotations generated by COHSE which are referred by XPointers [5]. In case of a page reformulation XPointers can easily become obsolete. If the underlying tree structure of the page which is represented by the DOM, is changed then the XPointers based on that tree can become invalid. For example, we can have an XPointer that refers to the second table element which is a child of the body element (/html/body/table[2]...), but if the first table child is removed then this XPointer cannot be accessed or can point to the wrong part of the page depending on the other children of the body element. Therefore, the order of application is important. For example, removing the first `Footer` object and then the `Header` is a safe combination. This is because a `Footer` is at the bottom of a page, so removing corresponding elements in the DOM tree will not effect the XPointer of the corresponding `Header` elements.

Copied pages Another way to combine heuristics is to create an alternative view of the page rather than modifying the original. For example, physical fragmentation

technique does not change the original page, it creates a number of smaller pages and a table of contents without modifying the original. Therefore, this heuristic can always be combined with other heuristics as long as they do not modify the DOM tree. Similarly, moving link menus to a new file or creating a list of links in a new page, can also be easily combined with other heuristics.

Based on the customisations discussed in the previous section and the combination of issues discussed here, users can interact with these heuristics in different ways:

- Users can choose which transformation technique to apply. This is the way it is implemented in the current Dante transformer. The main advantage here is that depending on the users' task, they can decide which technique to apply. However, users have to make a decision as to whenever they access the page which is the main disadvantage.

- A structural table of contents can be created for a page depending on the annotations. Here the idea is to create a separate page for each structural component and allow users to get an overview of the structure. For example, a table of contents can be created which says "Header, Sidebar, LinkMenu, Link, Footer, etc.", these will all provide links to separate pages that include for example only the Header, or only Sidebars, etc.

- Customisation options discussed in the previous section can be turned into a preferences vocabulary that will allow users to specify which actions to apply on which objects and how. Such preferences will also have restrictions which will take into account the fact that the certain transformations cannot be combined (Table 5.5).

- All transformation heuristics can be applied on independent copies of the original page and a root page can be created to link them. In this case, we do not have to consider if the transformation heuristics can be combined or not because they all will be applied independently. For example, a root page can include links such as "Physical fragmentation, Logical fragmentation, Without Header and Footer, List of links, etc.).

These are alternative approaches to the way the Dante transformer is implemented. The main conclusion here is that knowing the structure of the page provides the flexibility of presenting the same page in different ways to meet the users' requirements. This is achieved by annotating Web pages with the WAfA ontology.

	T1	**T2A**	**T2B**	**T3A**	**T3B**	**T3C**	**T4**	**T5A**	**T5B**	**T6A**	**T6B**	**T7A**	**T7B**	**T8**	**T9**
T1-Physical fragmentation		√	√	√	√	√	√	√	√	√	√	√	√	√	√
T2A-Logical fragmentation (add TOC to top)	×		√	×	×	×	×	×	×	×	×	×	×	×	×
T2B-Logical fragmentation (add TOC to bottom)	×	√		×	×	×	×	×	×	×	×	×	×	×	×
T3A-Remove Header	×	×	×		×	×	×	×	×	×	×	×	×	×	×
T3B-Remove Footer	×	√	×	√		√	√	×	×	×	×	×	×	×	×
T3C-Remove Sidebar	×	×	×	×	×		×	×	×	×	×	×	×	×	×
T4-Add a skip link	×	×	×	×	×	×		×	×	×	×	×	×	×	×
T5A-Linearise (at top)	×	×	×	×	×	×	×		×	×	×	×	×	×	×
T5B-Linearise (to a new file)	√	√	√	√	√	√	√	√		√	√	√	√	√	√
T6A-Move SearchEngine (to top)	×	×	×	×	×	×	×	×	×		×	×	×	×	×
T6B-Move SearchEngine (to bottom)	×	×	×	×	×	×	×	×	×	×		×	×	×	×
T7A-Move Advertisement (to bottom)	×	×	×	×	×	×	×	×	×	×	×		×	×	×
T7B-Move Advertisement (to a new file)	√	√	√	√	√	√	√	√	√	√	√	√		√	√
T8-Move LinkMenus (to a new file)	√	√	√	√	√	√	√	√	√	√	√	√	√		√
T9-List of links (to a new file)	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

√- Can be combined.
×- Cannot be combined.

Note: This table represents the order of application; the row name has to be read first.

Table 5.5: Combination of transformation heuristics if in-situ same page.

5.5 Summary and Conclusions

This chapter has presented the mobility support tool, Dante, and its underlying approach and architecture. Dante is built upon the Travel Analysis Framework. Dante identifies travel objects and their roles, annotates them with the WAfA ontology and transforms pages accordingly to enhance the provided mobility support. Dante proposes a pipeline approach to annotate pages that allows annotations arriving from different sources. An RDF Editor is also presented that can be used to create relationships between the annotations. Dante then uses these annotations and their relationship(s) to transform pages in different ways.

The transformation techniques explained are simple yet efficient. In fact, they can be considered as intermediate operations that transform pages into a form which can be easily accessed with a screen reader. In an ideal case, screen readers can also provide functions based on the annotations by using the WAfA ontology directly. These transformation techniques can be considered as demonstrations of how the WAfA ontology can be used.

As explained in Chapter 2, in the literature there are different approaches to transform pages. Most of these approaches use HTML source code to transform pages and not many of them try to understand the structural and navigational properties of pages as we do. The closest approach to ours is proposed by [9, 106]. They also annotate pages with a vocabulary and transform them accordingly. However, their vocabulary is limited and they only consider structural properties of pages. Consequently, the provided transformations are also limited. For example, they provide re-ordering of objects and adding a skip link. But with our approach, even though we presented a specific set of transformation techniques, this set can easily be extended because the ontology provides extensive information about not only the structure of the page but also about the provided navigation support.

The annotations and transformations also demonstrate the practical usefulness of the WAfA ontology. The concepts and the relationships included in the ontology are good enough to be used by Dante to perform transformations. In the following chapter, a user evaluation is presented to demonstrate the utility of the transformations, which in consequence demonstrates the utility of the annotations and the WAfA ontology.

Chapter 6

The Evaluation of Dante

The evaluation of Dante aims to demonstrate that the annotations and in consequence the performed transformations ease the mobility and enhance the travel support for visually impaired users. The overall evaluation consists of two parts: user evaluation and technical evaluation.

The user evaluation aims to demonstrate that transformations improve the mobility of visually impaired users and enable them to travel easily and confidently around Web pages. In order to demonstrate this, a task-based evaluation was conducted. Ten participants were asked to perform a number of tasks on both the original and the transformed versions of a number of pages. In order to be able to assess how participants performed on both versions, we used the NASA TLX, Cooperative evaluation and a structured interview. All participants tried both the original and the transformed pages, but in order to eliminate the learning factor, half of the participants tried the original first while the other half tried the transformed version. In overall, the TLX results demonstrated that the mental workload was lower for the transformed versions. Participants' comments and the results of the structured interview also confirmed the findings of the TLX results. In summary, this user evaluation confirmed our hypothesis that transformations performed by Dante enhance the provided mobility support.

The technical evaluation aims to demonstrate the utility of the annotations. This part of the evaluation focuses on demonstrating the importance of annotations with the WAfA ontology and how they contribute to the transformations.

6.1 The User Evaluation

In the previous chapter, we explained how Dante annotates pages with the WAfA ontology and transforms them accordingly. These transformations aim to enhance the experience of visually impaired Web travellers by providing better mobility support. In order to demonstrate that we have achieved our goal, a user evaluation was carried out. A comparative task-based evaluation has been chosen. Participants were asked to perform a set of tasks on the original pages [1] and on their transformed versions. The main hypothesis of this evaluation was:

> By transforming Web pages, we improve mobility, consequently enhance the travel support provided.

If the participants would be able to complete the same set of tasks better and easier with the transformed than the original pages, then we would be able to conclude that the transformations were useful and enhanced the mobility of visually impaired users. Demonstrating that the transformations improve the mobility and travel support over and above that afforded by original pages will give a stronger indication of the value of encoding transformations. In the following sections, we first present the details of the evaluation, and then we present and discuss the results.

6.1.1 Participants

Ten Web users participated in the evaluation with ranging between 25 and 65 years (three females and seven males). Eight of the participants were blind and two of them were partially sighted. Their length of Web experience was varied. The average number of Web usage hours was also varied, ranging between half an hour to nine hours per day. Blind participants accessed pages with a screen reader (Jaws[2] and Hal[3]) using IE with Windows operating system. One of the partially sighted participants used a screen magnifier (Lunar[4]) with IE (see Table C.2 in Appendix C). Even thought, ten people participated in the evaluation, two of the participants could not complete the evaluation because they were trying to access pages with a screen reader that they were not familiar with. Lack of experience in using the available screen reader hindered them to complete the given tasks. Therefore, we are not taking into account the

[1] In this chapter, *original version* refers to the page provided by the owners which was available online at that time.

[2] See `http://www.freedomscientific.com/`.

[3] See `http://www.dolphinuk.co.uk/`.

[4] See `http://www.dolphinuk.co.uk/`.

results from these participants. In the rest of this chapter, other participants are referred to as P1–P8.

6.1.2 Materials

Three Web pages were used in the evaluation. The result page of Google[5] for the word 'Olympic' was used for training. The design of Google is widely known and popular for its simplicity. It also provides many travel objects. Therefore, transforming such a simple page with many travel objects with respect to the Travel framework, was good enough to demonstrate the purpose of the transformations.

For the main evaluation part, the home page of Manchester University[6] (see Figure 6.2) and the home page of Internet Movie Database (IMDB)[7] (see Figure 6.4) were used. Section C.1 in Appendix C presents each page along with their transformed versions used in the evaluation. Due to time limitations, we decided to use only the physical fragmentation in this evaluation rather than exploring different transformations explained in the previous chapter.

The pages used in the evaluation were particularly chosen based on the classifications proposed by [6]. They classify pages into seven according to their functionality but not to the content. Since in our work we are also interested in the structure of the pages, we have chosen pages from different categories suggested by this study. The home page of Manchester university is a typical 'Universities and Research Institutes' page and the IMDB page is a "Content and Media Site". Therefore, these pages provided us with an exemplar set.

6.1.3 Tasks

Participants were asked to perform a single task on the training page and two tasks on other pages. For the training page (Figure 6.1), the aim of the given task was to demonstrate how Dante fragments pages and introduce participants to an exemplar task. The following task was used for the training purpose:

> Assume that you searched for the word "Olympic" and reading the results.
> Can you find out what the second returned result is?

[5]See `http://www.google.com/`.
[6]See `http://www.man.ac.uk/`.
[7]See `http://www.imdb.com/`.

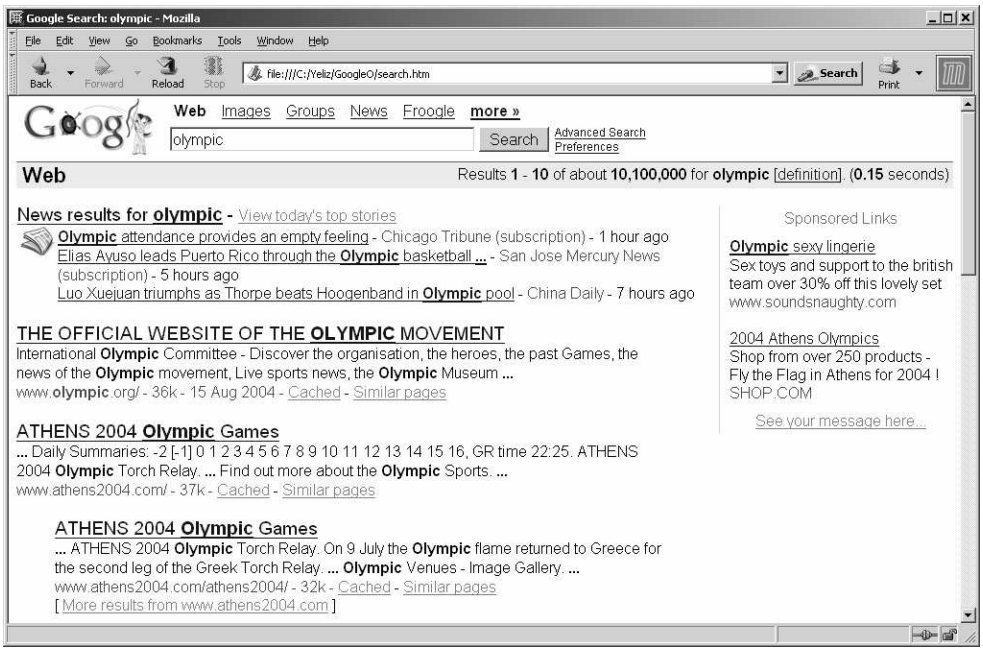


Figure 6.1: A result page of Google (17/08/2004).

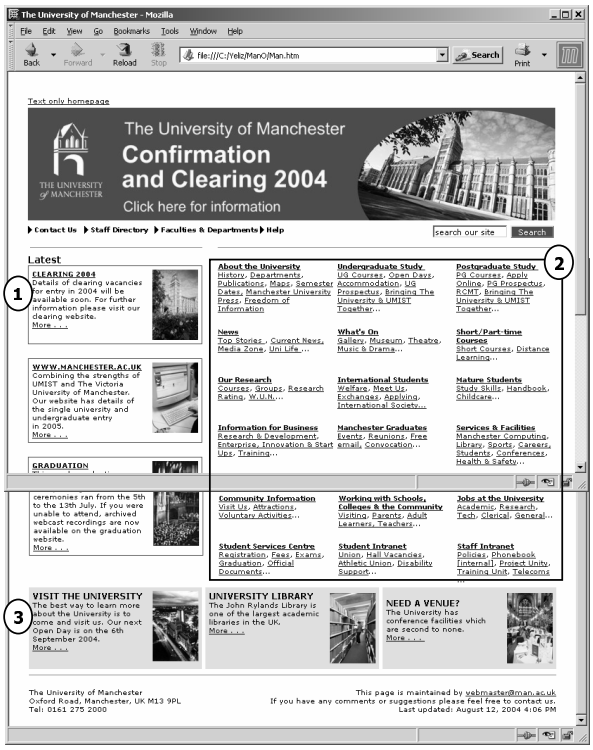


Figure 6.2: The home page of Manchester University (17/08/2004).

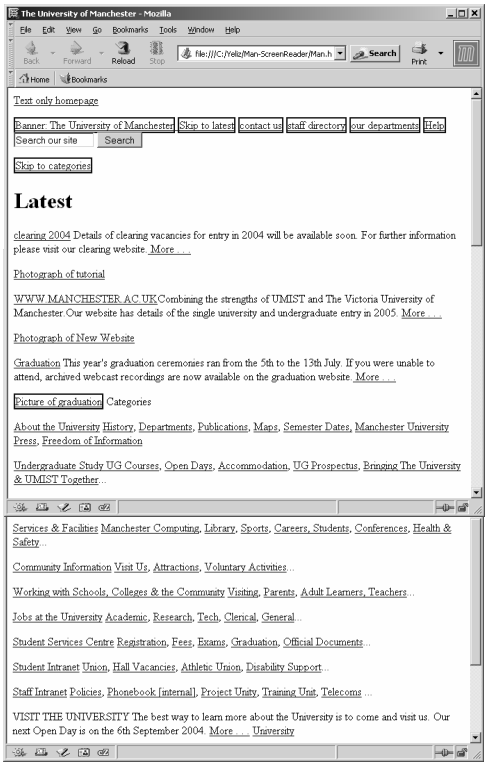


Figure 6.3: The home page of Manchester University in audio.

Figure 6.4: The home page of IMDB (17/08/2004).

Google (see Figure 6.1) visually presents search results in a well structured manner. The overall structure and layout of the page provide implicit information about the returned results, their importance, etc. The number of returned results are in a banner with a distinctive colour; sponsored links are on the right hand side with a lighter colour in order not to distract the reader; extra news results are at the top and also presented in a different style; finally the results presented more or less in the middle centre of the page where each result is presented in a separate paragraph. A screen reader user has to read the search box, number of results, sponsored links, news results in order to get to the main results section. Therefore, the given task aims to show that it is not easy and can take long time for visually impaired users to access the results. It also aims to demonstrate how physical fragmentation of pages can be useful.

Table 6.1 presents the tasks used in the main part of the evaluation. These tasks were prepared to address certain mobility and travel issues, particularly mobility *principles* (Table 6.2) and mobility *instruments* (Table 6.3). Since participants can easily transfer their knowledge about a page after completing the tasks on the original to transformed version or vice versa, we decided to slightly modify either some of the tasks or pages. For example, Figure 6.4 which is the original version, the welcoming paragraph starts with "Welcome to the Internet Movie Database, the biggest,...". In the transformed version, we slightly changed this as "This is the biggest, best, most ..." to make sure that the participants did not memorise the sentence. Some of the tasks were also slightly modified.

No	Task
The University of Manchester home page (see Figure 6.2)	
T1	The main content of this page is visually organised as three sections, can you find out what these sections are? **Transformed**[a] And how many sub-sections are there in the "categories" section?
T2	There are two links to access the home page of the university's library, can you try to find one of them and tell me which section is this link in? Can you also tell me if can find out the name of the library without actually visiting the home page of the library?
The IMDB home page (see Figure 6.4)	
T3	This page is visually laid out into three columns and actually the main content is in the middle. Can you try to find the starting point of this main content and tell me what is the first sentence? **Transformed:** Can you find out two movies that will be available on DVD tomorrow?
T4	There is a section in this page where IMDB highlights a movie for that particular date, try to access that section and try to find out what are the previous and next sections (giving their titles will be enough)?

[a]The keyword *'transformed'* indicates the version of the task that is slightly modified and used for the transformed version of the page.

Table 6.1: Tasks used in the main body of the evaluation.

The aim of the first task (T1) (Table 6.1) for *the home page of Manchester University* (Figure 6.2) was finding out if the participants could easily get an overview of the page. In its main content, this page has three visually distinctive sections (excluding the header and footer). These sections are: (a) *latest news* which is on the left, (2) *categories* which is on the right and (3) a number of referential sections below the first two sections. Distinctive colours and lines are used to create clear separation between these sections. Moreover, the categories part has a list of sub-sections where each of them has a distinctive title and a list of links. However, only one heading in this page is explicitly marked up. Therefore accessing this page in audio is not an easy task. For example, the sub-sections in the categories section are all read as a single list of links. The second task for this page (T2) aims to address link presentation issues, in particular getting a preview of the destination page. There are two links to the "John Rylands Library". One of them is provided in the section called "services and facilities" in the categories part and the other one is provided in the last section called "university library". One of these links actually provide the name of the library but not the other one. Therefore before actually visiting the page, the user would not be able to get the

Mobility	**T1**	**T2**	**T3**	**T4**
Principles				
Regularity Can the user realise the regular or repeated structure used in the page?	x		x	
Information Flow Does this page provide small manageable units of information that the user can easily use?	x		x	
Granularity Does this page provide enough landmarks and cues? and if it does, are they in the right format?			x	
Egocentricity Can the user associate objects and their positions to his (her) current position in the page?		x		
External Memory Can the user easily access objects that support external memory?	x			
Familiarity Depending on the user's previous journeys, can (s)he predict the content and structure of the page?	x		x	
Spatial awareness Can the user relate this object (e.g., link) to others in the same spatial proximity?		x		x

Table 6.2: Analyses of tasks (Table 6.1) with respect to mobility principles.

name of the library.

The home page of IMDB is also visually very well structured. The link menus and search boxes are on each side of the page and the main content is positioned in the middle column– the structure of the page makes a distinction between "main" and "side" content. However the source code actually does not reflect that. Therefore, the first task for this page (T3) aims to identify the techniques that visually impaired people use to access the main content and also find out if they can easily make distinctions between "main" and "side" content. In the main body of the page, several sections are clearly presented. The second task for this page (T4) aims to address "spatial awareness"– can users create a relationship between these sections?

As a summary, the tasks used in the evaluation were designed to address travel and mobility issues. Table 6.2 and Table 6.3 presents the relevance of these tasks to mobility principles and instruments respectively.

Mobility	**T1**	**T2**	**T3**	**T4**
Instruments				
Preview Does the link label provide enough information so that the user can decide to follow or not to follow it? e.g., can the user find out the name of the library before actually visiting the target page?		x		
Probe Can the user identify and use page elements? or Can the user make use of objects that are provided to support intra-navigation?			x	
Scanning Can the user scan the page?	x			
Browsing Can user easily browse a page?	x			
Searching If the user is looking for a known target, how (s)he behave?		x		
Overview Can the user easily get an overview of the page?	x			
Differentiating page elements Can the user notice the distinction between main components of the page such as sections?				x
Orientation Does the user know his(her) position in the page (or at least have an idea)? e.g., when the user reads a link, do s(he) know where that link is?				x
Getting contextual information Can the user access the contextual information (such as the section that a link is part of)? or can the user notice certain groupings and their parts?		x		x
Accessing the focus of the page Can the user easily access the focus of the page?			x	
Accessing Layout (organisation) If the layout or the organisation of the page plays an important role in understanding the content of the page, can the user access such information?	x		x	

Table 6.3: Analyses of tasks (Table 6.1) with respect to mobility instruments.

6.1.4 Methods

In order to assess how difficult the participants found the tasks and also to compare how well they performed with the original and transformed pages, the following three methods were used:

Cooperative evaluation is considered as an alternative evaluation technique to the strict controlled experiment paradigm [120]. In this technique, while participants perform tasks, they are encouraged to think aloud. In our evaluation, participants were encouraged to particularly comment on the commands they

issued, things they liked and did not like about pages and their preferences. Capturing such subjective data was important for this evaluation. Quantitative data such as number of commands, speed and accuracy cannot tell the whole story of the usability and utility of transformations over original pages. This is because participants' experience of using their assistive technology, previous knowledge about the pages, etc., can also easily affect such quantitative data. This evaluation sought to judge whether the participants could accomplish the tasks in a manner they wished, to their own satisfaction.

Task Load Index (TLX) is a way of measuring the workload under test conditions which was developed by NASA [57]. The workload is measured based on the following aspects of difficulty: (a) mental demand, (b) time pressure, (c) effort expended, (d) perceived performance and (e) frustration experience. Participants were asked to give numerical ratings on a scale of 0 to 20 for each of these aspects. Except from "perceived performance", a higher score indicates a more difficult task for others. The details of TLX can be found in Table C.1 in Appendix C.

Structured interview After recording the TLX scores, a small structured interview was carried out based on a questionnaire which covered the following two broad categories: orientation and movement [38]. Table 6.4 presents this questionnaire with respect to the mobility framework.

6.1.5 Procedure

Each participant performed given tasks on both versions of the same page: original (the page that was available online at that time) and transformed (the page that was transformed by Dante). Each participant used one version and then the other (e.g., first original then transformed or vice versa). Lessons learnt from the first version may transfer and affect performance and preferences for the second one (positively and negatively) [104]. In order to eliminate this memory effect, we introduced half of the participants to the original version first while the others were introduced first with the transformed version. Therefore we did not have two separate user groups but the same group tried both versions. We did not give any time limit for performing the tasks, however participants were encouraged to consider giving up if they were spending long time without feeling they were making progress.

Question	**Mobility Principles**	**Mobility Instruments**
Do you know (do you have an idea) how long this page is (we are not looking for the exact size but trying to find out if you have an idea about the size?)	-	Overview
Do you think now you have an overview of the page?	-	Overview
How difficult was it to get such an overview?	-	Overview
What do you think about the amount of information?	Granularity	-
Did you find this page complex? If yes How complex you think is this page?	Information flow	-
Was there anything particular you found repetitive in the page?	-	Accessing the focus of the page
How easy was it to access different parts of this page?	-	Probing, Differentiating page elements
How easy was it for you to figure out your position in a specific section with respect to other sections in the page?	Spatial awareness	Orientation, getting contextual information
At any point in your journey, did you think you were lost (didn't know where you are)?	Orientation, spatial awareness	-
Was there anything particularly good or bad about this page?		

Table 6.4: Post-evaluation questions with respect to mobility.

In order to be able to make a fair comparison between original and transformed pages, we did not ask participants to use Dante itself but we asked them to use the transformed pages generated by Dante. We used local copies of both original and transformed versions. Participants used their own browser and assistive technology to access these local pages. This would not only allow us to make a fair comparison, but also would eliminate the effect of the learning curve needed for the Dante. Furthermore, the main goal of this evaluation is not to assess the usability of the tool, but to assess the usability and utility of the transformations performed by Dante. Broadly the evaluation comprised of two parts: *introduction* and *main* which are explained as follows:

Introduction part First we introduced the participants to the main goal of Dante. We explained that Dante transforms Web pages into a form to enhance provided movement support. We explicitly indicated that the aim of this evaluation is not to assess the participants but to evaluate the transformation approaches encoded

in Dante. We then asked a number of questions to participants about themselves (e.g., age, experience of using Web, etc.) (see Table C.2 in Appendix C). The aim of these questions was to create a profile of our participants that are already discussed in Section 6.1.1. Finally in this part, we used the result page of Google as an example to explain how Dante transforms pages (See Figure C.2 in Appendix C). We used an example task to present the overall structure of the evaluation (Section 6.1.3). The results from this page are not considered in the overall evaluation assessment.

Main part Two pages were used in the main body of the evaluation (Section 6.1.2). Two tasks were used for each page (Section 6.1.3). The procedure followed can be summarised and explained as follows:

- Perform tasks on the original page (or transformed);
- Rate TLX factors; Structured interview;
- Perform tasks on the transformed page (or original);
- Rate TLX factors; Structured interview.

Half of the participants were first asked to perform the two tasks on the original version of the Manchester University home page. After completing these tasks, participants were asked to rate the subjective workload factors of TLX (Section 6.1.4). This was followed by the structured interview 6.1.4. Participants were asked a number of questions in order to capture their experiences of using that version of the page (see Table 6.4). After completing these questions, participants were asked to perform same tasks with the transformed version. This was followed again by TLX and structured interview. Same procedure was followed for the home page of IMDB. Other half of the participants started with the transformed version and then used the original page, but the overall structure was again same.

6.1.6 Mental Workload: TLX Results

All participants performed given tasks (see Table 6.1) on the home page of Manchester University (see Figure 6.2) and six participants performed given tasks on the home page of IMDB (see Figure 6.4).

Table 6.5 shows the average scores for each of the workload factors for the home page of Manchester University; the raw data can be seen in Table C.3 and Figure C.7

- C.12 in Appendix C. Similarly, Table 6.6 shows the average scores for each of the workload factors for the home page of IMDB; the raw data can be seen in Table C.4 and Figure C.13 - C.18 in Appendix C. Each factor was marked in the range 0-20. The overall workload was calculated from ratings in Table 6.5 and Table 6.6 as follows. For the factors *Mental Demand*, *Physical Demand*, *Temporal Demand*, *Effort* and *Frustration Level* a high score (near to 20) represented a positive result. For example, a score of 20 for *Mental Demand* implied that the demand was very high. On the other hand, a score of 20 for *Perceived Performance* would imply that the participant had not done well (performance was considered as low or poor). Therefore, in order to calculate the overall score, the *Perceived Performance* scores were complemented (subtracted from 20). Thus an overall score approaching to 20 implies that the page is difficult to use, whereas a score near zero means that the page is easy to use. Paired Student's *t*-tests were also used to assess the significance of any differences between ratings for each factor [38]. For all TLX factors, our results were statistically significant and we provide *t*-tests analysis as we discuss results for each factor.

The overall mean mental workload for the home page of Manchester University was calculated as 6.38 for the transformed page and 12.63 for the original page (t=2.58, df=5, p=0.05). This difference was also confirmed in the home page of IMDB; the overall mental workload was calculated as 14.58 for the transformed and 3.08 for the original version (t=6.22, df=5, p=0.002). The difference was higher in the IMDB page because there were a number of cases where participants gave zero to the transformed and 20 to the original version (e.g., see *Frustration level* column in Table C.4 and Figure C.18). The overall increase for the mental workload for the original version of the IMDB page suggests that participants found that page more demanding than the home page of Manchester University.

For all factors, compared to transformed version, the average given scores for the Manchester University page were almost double for the original version. This was even higher for the home page of IMDB; the average given scores were almost tripled.

As can be seen from Table 6.5 and 6.6, the highest difference among the TLX factors is for *Frustration Level* (t=5.79, df=7, p=0.001 for Figure 6.2 and t=5.3, df=5, p=0.003 for Figure 6.4). This shows that the participants were highly irritated when they performed the tasks with the original version compared to transformed. This was actually confirmed with their comments which are discussed in the next section. The lowest difference among the factors for both pages were the *Temporal demand* scores (t=2.7, df=7, p=0.031 for Figure 6.2 and t=2.9, df=5, p=0.034 for Figure 6.4). This is

Factor	Original	Transformed	Difference
Mental Demand	13.88	5.63	8.25
Physical Demand	12.75	5.00	7.75
Temporal Demand	10.38	4.50	5.88
Effort Expended	13.63	4.38	9.25
Perceived Performance	9.88[a]	4.63	5.25
Frustration Level	15.00	3.38	11.63
Mean	**12.63**	**6.38**	

Table 6.5: Summary of TLX scores for the home page of Manchester University.

[a]For all factors, except the *perceived performance* level, a low score is positive in terms of usability (Table C.3 and Figure C.7- C.12 for raw TLX scores).

not surprising since participants were not given any time limits.

The significant difference between the given *mental demand* scores for both pages suggest that pages were well transformed to meet the requirements of the given tasks. The average difference mental demand score for the Manchester University page is 8.25 (t=6.38, df=7, p=0.0004) and it is slightly higher for the home page of IMDB which is 11.17 (t=6.22, df=5, p=0.002).

Although, the definition of the *Physical demand* factor can be considered as irrelevant to our evaluation (see Table C.1 in Appendix C), we asked participants to consider the physical activities such as using the keyboard and also whether they found the tasks laborious or not. As for other factors, the difference for this factor for both pages was also quite high; for the home page of Manchester University it was 7.75 (t=3.9, df=7, p=0.006) and for the home page of IMDB it was 10.67 (t=3.91, df=5, p=0.011).

The *Effort* expended was significantly reduced in the transformed version of both pages. The difference for the Manchester University page is 9.25 (t=7.24, df=7, p=0.0002) and for the IMDB page the difference was 11.17 (t=3.96, df=5, p=0.011). Finally, *subjective performance* achieved was significantly increased in the transformed version of both pages. The difference was 5.25 (t=3.51, df=7, p=0.010) for the home page of Manchester University and 12.67 (t=6.83, df=5, p=0.001) for the IMDB page.

As a summary, the results for all factors demonstrated that the participants found the same tasks more demanding to perform on the original page than the transformed versions. Since our tasks were designed to address mobility (Section 6.1.3), these results confirmed our hypothesis that transformations improved mobility (Section 6.1).

Factor	Original	Transformed	Difference
Mental Demand	15.67	4.50	11.17
Physical Demand	13.50	2.83	10.67
Temporal Demand	11.50	2.50	9.00
Effort Expended	14.33	3.17	11.17
Perceived Performance	16.17[a]	3.50	12.67
Frustration Level	16.33	2.00	14.33
Mean	**14.58**	**3.08**	

Table 6.6: Summary of TLX scores for the home page of IMDB.

[a]For all factors, except the *perceived performance* level, a low score is positive in terms of usability (Table C.4 and Figure C.13- C.18 for raw TLX scores).

6.1.7 Results and Discussion

In this section, we present structured interview results and also our observations. Here we again discuss results by using the mobility framework. Particularly, we again refer to *mobility principles* (see Table 6.2) and *mobility instruments* (see Table 6.3). The raw data for the structured interview can be seen in Appendix C in Table C.5 and C.6 for the home page of Manchester University, and in Table C.7 and C.8 for the home page of IMDB.

Orientation and Spatial Awareness

We asked participants to comment about the orientation (i.e., if they were able to answer the question 'where am I?') and spatial awareness (i.e., if they knew their position and location in relation to other parts of the page) while they performed the tasks.

Orientation Half of the participants for the home page of Manchester University mentioned that at certain points during their journeys they did not know their position in the page. P6 indicated that "several times I felt like I could not continue and moved to top of the page to start again". When we observed participants, this was actually what most of the participants did when they actually felt lost. Moving to the top of the page provides them an explicit and unique information about their location. From there on, when they read the page, they tend to use the top of the page as a reference point; the rest of the page is compared and associated to the top of the page. On the other hand, none of the participants indicated that they were disoriented in the transformed page.

Participants' responses concerning orientation for the home page of IMDB were

more negative. For instance, P5 mentioned that "I could not figure out my position at all". Similarly, at some point P4 indicated that "I have no clue about the content of this page, I am totally lost". Participants again moved to top of this page and started to read the page from the top whenever they felt they could not continue. Our partially sighted participant also used this technique to re-orientate herself however as opposed to blind participants she did this for different purpose. For example, in the IMDB page she moved to top of the page because she was looking for the 'banner' at the top. She indicated that this is because the banner has a distinctive colour and shape (running across the page). On the other hand, none of our blind participants mentioned the banner at the top.

Spatial awareness Except for one participant, all participants thought that it was difficult to figure out their position with respect to other components of the original version of the Manchester University home page. For example, P8 indicated that "it was very difficult, for that purpose I usually use the list of links so that my screen reader gives me exact position of the item that I am reading with respect to the other items. But unfortunately in this list there were so many links with labels such as more. . . ". On the other hand, all participants found it quite easy with the transformed page. Similar results also emerged for the home page of IMDB.

Getting an Overview of a Page

Only one participant (P1) could get an overview of the page by reading the original version of the Manchester University home page. On the other hand, all participants mentioned that the table of contents of the transformed version helped them easily get an overview. For example, P4 commented that the transformed page does not have repetitions which is important to get the overview. Similar comments were also given for the home page of IMDB. For example, P6 indicated that "even though I have visited this Web site before, I did not know that those sections exist". Some participants also highlighted that the table of contents enabled them to easily see what is available on the page and to decide which part of the page they want or need to read.

Logical vs. Physical Page Size

We asked participants to give their opinions about the size of the page. The aim was not to find out if they knew the exact physical size of the page (e.g., number of words) but to find out if they can easily get an idea about the size of the page and what kind of information they use to come to the conclusion. Most of the participants indicated

that the original version of the home page of Manchester University was quite long and it was observed that participants paid quite a lot of attention to the total number of links in the page. For example, P1 mentioned that "when I entered the page my screen reader said there are 123 links which means this page is quite long". On the other hand, when we asked the same question for the transformed version, most participants found it much shorter, more structured and more manageable. However, most participants also highlighted that with the transformed version, they could not have an idea about the physical size of the page. Even though, they found the number of logical sections indicative about the size and density of the page, they thought it was not enough to find out the actual physical size of the page. For example, P7 said that after he accessed both pages, he did not think that both pages were the same size.

Most participants commented that they did not have an idea about the size of the IMDB page. For example, P1 commented that "there were so many stuff that I had to read before I reach the main content that I do not think I have an idea about the size of the page but the only thing I can say is that it is too long". All participants indicated that they found the size of the transformed page much better. Even P8 stressed that "with this transformed page, at least I know the number of sections in the page".

Most participants preferred the length of the transformed page to length of the original page. Although participants could easily get the number of logical sections in the transformed page, they could not get the actual physical size. In order to do that they had to visit each fragment individually. Participants also could not use the information that they often use to judge the page size. For example, the number of links in the generated table of contents, can give an idea about the logical number of sections of the page but not it's physical size.

Differentiating Page Elements and Movement

We asked participants how easy they found accessing different parts of the pages used in the evaluation. All participants found it difficult for both pages. P1 indicated that "he could not see any distinction between sections at all". Similarly, P4 highlighted that "it was very difficult to picture the sections and overall page in my mind". On the other hand, all participants thought that it was quite easy to differentiate sections with the provided table of contents in the transformed version. They indicated that fragmentation helped them have a good idea about the logical sections in the page. Furthermore, fragmentation helped them clearly see the distinction between the sections.

Information Flow

All participants found the amount of information in both pages quite a lot. P1 said that "when I entered the page, my screen reader told me that there are 123 links which means this page is quite long". Similarly, P4 mentioned that "awful lot of links that were immediately at the top of the page. I didn't get any information about the page and started to read immediately the list of links". However, as P8 indicated "although there are a lot of information in the page, in my opinion the question is if you can get to that information and find what you want, in this page it takes you long time to find what you are looking for". Compared to the original pages, all participants found the amount of information in the transformed versions much better. Some participants also thought the transformed version is much better structured and the sections are clearly stated, and has less clutter to go through. P8 highlighted that "although there are as much less information as possible here, you can easily find what you are looking for". Moreover, P4 also stressed that "the transformed version turned a huge page into small manageable bites". This was quite important concerning the information flow.

Directed Search

When participants were asked to perform certain tasks, we realised that expect one of the participants, all were trying to formulate a number of keywords to search in the page by the CTRL+F command. The participant who did not use this command had only a year experience of using a screen reader. P4 indicated that "I really like and use this command (*CTRL+F*), this gives me a controlled browsing feature and I do not have to read everything in the page to find something". Although most of the participants use this command, one of them indicated (P5) that "when I use Ctrl+F it is like you want to get the salt from the table and somebody takes your hand gets the salt, then you would not know what else in the table and what is around it, etc., if the black pepper is next to it and you need that you wouldn't know. You have to ask that person to take your hand again. It is exactly what is happening here". Although as this participant stressed, this command does not support serendipity browsing, it still helps users to complete certain tasks efficiently and quickly [31]. However, with our physically transformed pages, participants could not use this command. They had to move between fragments and then perform this command. We therefore need to address this issue further. A possible solution would be to have the table of contents at the top of the original page. This would allow users to have an overview of the page

and also use the commands that they are familiar with.

Visual vs. Audio Presentation of a page– Different Perception

This evaluation also highlighted the differences between audio and visual navigation that have already been discussed in Chapter 2. Figure 6.3 shows how the home page of Manchester University (see Figure 6.2) is perceived by a screen reader user in audio. The layout of the page which provides implicit information about the content is missing in such an audio presentation. For example, the original page has "side-content" and "main-content" next to each other. However, in the audio presentation the content of the page is linear. Everything in the page is read like a list. For example, one of the participants (P1) expressed this as follows for the home page of IMDB: "for a sighted user, the layout is as follows: the sidebar is on the *right*, main content is in the *middle column* and the other sidebar is on the *left*, however, for a visually impaired user, the sidebar is at the *top*, the main content is in the *middle row* and the other sidebar is at the *bottom*– these are two completely different views of the same page". Therefore, the way the table of contents was presented, was appropriate for the kind of interaction supported by screen readers. The table of contents was a linear list of links which most of our participants preferred to the original page.

When our partially sighted user was trying to perform the tasks, we observed that even though she had a limited view of the page with the screen magnifier, she was still using certain kinds of travel objects in the page (see Chapter 3). For example, when we asked her (him) to identify the sections in the home page of Manchester University, she could easily differentiate the bottom (third) section from the others. She explicitly stated that the distinctive colour and borders (lines around) [8] helped him (her) enormously identify those sections. However, none of our blind participants could access such information and therefore could not make use of it. When we also asked our partially sighted participant to find the link to the library, she identified link by using the provided image. However, none of our blind participants mentioned anything about the images in the page. Finally, our partially sighted participant indicated that the best headings for him (her) are the ones with capital letter with different colours, larger font or bold. Since blind participants cannot access the way headings are displayed, they paid attention if they were explicitly identified as headings in the audio presentation.

[8] We referred these kinds of objects as *Way Edges* (see Section 3.2 in Chapter 3).

Importance of Explicit Structural Information

We also observed that even the wrong (or inappropriate) structural markup can be useful. Participants referred highly to the structure of the page as long as their screen reader informed them. The experience of participants was also an important factor for making use of such structural information. The more experienced they were, the more they referred to the structure. This was again the issue of familiarity with the Web. For example, the list of links in the sidebars of the home page of IMDB (see Figure6.4) were positioned in tables. Even though, the table structure does not indicate anything about the fact that they are a menu, some of our participants still used tables to make the distinction between consecutive lists of links. For example, P1 said that "I could detect only a set of tables and a lot of links so I am guessing that these are different groups of navigation links". Another participant (P7) after realised that the table structure used for menus, started to move from one table to another. So this was quite important for being able to move easily and confidently in the page. Similarly, the list of links in the right hand side of the home page of Manchester University (see Figure 6.2), were actually marked up as paragraphs. So one of the participants (P3) indicated that his screen reader reads a set of paragraphs and links, therefore he concluded that they were consecutive link menus. Therefore, even if the structural markup does not explicitly provide the correct information, it can still be useful and indicative.

Environment and User's experience

In this evaluation, it was clear that a number of things had a strong influence on the mobility of the participants. The following list summarises some of these:

1. *Assistive technology* (e.g., Jaws): Different versions of the same screen reader can present the page differently. This obviously affected the performance of our participants. For example, the generated table of contents for the transformed pages were marked up as nested lists. Although the latest version of *Jaws* presents nested lists explicitly, earlier versions did not make any distinction between nested lists. Therefore, some participants could not access the logical relationship between the list items, and consequently the relationship between sections.

2. *Browser* (e.g., IE): Some screen readers work better with certain browsers. For

example, Jaws works much better with Internet Explorer than with Mozilla because of Microsoft's accessibility support[9].

3. *Experience of the user* (e.g., experienced v.s. novice user): The same version of a screen reader is used differently by users depending on their experiences. For example, novice users relied highly on the up and down arrow keys, but experienced users could easily use the supported functionalities such as the list of links, headings, etc.

4. *Purpose of the user* (e.g., browsing a page or looking for something specific): Some of our tasks were more directed (looking for something specific), in that case almost all our participants tried to use the command (*CTRL+F*) to search for the formulated keywords. However, when they need to perform a less directed task, they were trying to read the page in different ways.

Therefore, in order to improve the mobility of visually impaired users, all these factors also need to be considered.

Physical vs. Logical Fragmentation

In this evaluation, we compared physical fragmentation of the pages with the original versions. Although TLX results (Section 6.1.6) and participants' comments demonstrated that physical fragmentation improved participants' experiences and mobility, there are still some issues that need to be addressed. For example, with the table of contents, participants could not use some of the commands that they are familiar with and frequently use. These include searching for a keyword (*CTRL+F*), accessing lists of links or headings, etc. Some of the participants indicated that they would like to have the table of contents at the top of the original page, so that they can read the table of contents as well as using the commands. The partially sighted participant also indicated that having the table of contents at the top of the original page would allow him (her) still to see the images and colour contrasts provided in the page. Furthermore, some participants also commented about the movement between the fragments and table of contents. They indicated that although physically breaking page into smaller pages helped them see the distinction between sections, sometimes it was demanding to move between fragments, and the table of contents and fragments.

[9]For more information, see `http://www.microsoft.com/enable/`.

Unorthodox Reference Points

We have also observed that some users explicitly referred to certain elements of the page to complete their tasks. For example, in the Google results page, for each result Google associates two links 'cached' and 'similar pages'. We observed that some participants used these repeated links to make a distinction between the returned results. We refer to such objects as *reference points* (see Section 3.2 in Chapter 3). Even though these links were not visually distinctive (on the contrary they were visually faded), since they were repeated regularly, blind participants referred to them quite often. For example, P4 indicated that "I tab through the page until I get to the links 'cached' and 'similar pages'. These are my reference points, when I read them for the first time I know that it is the first result". These repeated words were useful to blind users. Similarly, the partially sighted participant indicated that in order to differentiate returned results from the rest of the page, she looks for the repeated structure in the page. She stressed that she visited the Google results page before and knows that the results are sequentially presented in the same style, so she looks for repeated style.

All participants at some point in the evaluation also used the top of the page as a *reference point*. Whenever they could not continue or were not sure about their location, they moved to the top of the page to re-orientate themselves and start to read the page again.

Regularity and familiarity of Environment

There were two kinds of experience which we observed that had influence on the participants' performance: (a) familiarity with the Web (being familiar with general Web page conventions) and (b) familiarity with a particular page.

Familiarity with the Web We observed that the participants relied on their previous experiences and knowledge about Web pages when they performed tasks. For example, one participant (P1) while trying to complete tasks on the IMDB home page mentioned that "at some point when I read something about languages, I thought that that kind of information is usually placed at the bottom of the page and I still could not complete the task, I thought somehow I didn't realise the content, so I moved back to the top and started again". This participant is actually referring to a convention which is common on the Web (but actually is not a standard). Different language options for a page tend to be placed at the bottom of the page. Therefore, in this case it is important to make sure that such kind of options are also at the bottom of the source code so they are

accessible in the same way to assistive technology users.

Familiarity with a particular page In particular, the effect of previous experience of using a page was quite obvious in the results page of Google (see Figure 6.1). We have observed that participants who used Google before knew about the objects presented at the top and they somehow used certain techniques to skip those objects. For instance, one of the participants (P4) indicated that "after you read the page a few times, you learn how to ignore and skip certain objects". We observed that experienced participants actually *TAB* through items at the top without actually reading them. We have also observed that participants used repeated words or links more frequently, if they have visited that page before. For example, participants who used Google before for searching, were using "cached" and "similar pages" links that Google provides for each result to differentiate them.

"How Users Read on the Web"

In this evaluation, it was also obvious that the more experienced the participants were, the less they relied on the screen reader reading the page to them. The experienced screen reader users, as sighted users [83], tend not to read the entire page word by word, but they scan the page. They use different techniques in order to do that. The following list summarises some of the observed techniques:

- Using TAB key to move link by link;
- Depending on the screen reader, using different reading modes (e.g., table mode);
- Using element by element movements (e.g., paragraph by paragraph movement);
- Using certain functionalities supported by screen readers (e.g., list of links, list of headings);
- Using Ctrl+F command to search a particular word in the page, they try to formulate a set of keywords depending on the task.

Although experienced users could easily use these techniques, novice users tended to rely on the way the screen reader renders the page and the best they could do was using the up and down arrow keys.

Summary of the Results

Fundamentally, the results presented above supported the TLX results discussed in Section 6.1.6. In particular, participants' comments suggested that they were in favour of the transformed pages. In general, their comments were more positive for the transformed version of the IMDB than the Manchester University home page. This was because they all found the IMDB page more complex and the efficiency of the physical fragmentation became more obvious for this page. The following list summarises positive outcomes:

- The orientation and spatial awareness of the participants were better on the transformed pages;
- Participants found it easier to get an overview of the page by reading the generated table of contents;
- Table of contents also helped participants have an idea about the logical structure of the page;
- Transformed version considered to be more organised, less cluttered and more manageable;
- Linear list structure of the table of contents is appropriate for the kind of interaction supported by screen readers;

However, this evaluation also highlighted certain issues that need further investigation about the transformations. These issues can be summarised as follows:

- With the fragmentation, it was not easy to get an idea about the physical size of the page;
- Participants could not use some of the commands that they are familiar with (e.g., CTRL+F) in the fragmented pages. Therefore having the table of contents at the top of the page itself (i.e., logical fragmentation) would be a better solution;
- The underlying markup of the table of contents need to be re-considered as some screen readers cannot differentiate nested lists. A possible solution would be using heading tags rather than list tags;
- Transformation can abolish some of the reference points that participants are familiar with;

- Moving between fragments, and table of contents and fragments can be demanding.

In summary, our TLX results, structured interview, and the participants' comments all demonstrated the utility of the performed fragmentations. In particular, results were discussed within the mobility framework and demonstrated that fragmentation improved the mobility of the participants.

6.2 The Technical Evaluation

It is not easy to evaluate an individual component of Dante. The user evaluation presented in the previous section focused on particular transformations. Even though the end-users provided feedback on the transformed pages, their comments were also important for the annotation component of Dante; the transformations were performed based on the annotations provided. Therefore, the following list mainly poses some questions regarding certain aspects of the annotation process and tries to answer them.

Appropriateness of annotations What kind of annotations are both necessary and sufficient for the desired transformations? This, in essence, is fitness for purpose. When we first started to experiment with annotating and transforming a page, we used to annotate every travel object in that page. However, after development and implementation of a set of transformation techniques (see Section 5.4 in Chapter 5), we started to annotate only the required annotations for those transformation techniques. For example, Table 5.1 shows the type of annotations required to perform the transformation techniques explained in Section 5.4 in Chapter 5.

Multiple ways of annotations Is it important how you represent the structure of a page? This depends on the implementation of the transformation techniques. For example, in the current implementation of the physical fragmentation, we look for `Chunk` annotations. Therefore, if something is annotated as a link menu, a new page will not be created for that part of the page. However, we could improve our implementation by considering the parent-child (*is-a*) relationships in the ontology. In this case, a link menu is a kind of list, and list is a kind of chunk, therefore a link menu can be treated as a chunk. However, if a part of the page is annotated as a chunk, that does not provide enough information to decide whether it is a header and can be removed from the page or not.

Amount of annotations How much annotation is enough? This again depends on the transformations that will be performed. For example, if annotations are needed to add a skip link to the page, then only specifying the headline in the page would be enough. Whereas if the aim is to logically fragment a page, then chunks and headings need to be annotated. If there are only two chunks in the page then only four annotations are needed (two chunks, two headings). But if there are more chunks than that, then more annotations are needed. The structure of the page is, therefore, also important.

Fitness of the WAfA ontology (completeness, expressiveness, etc.) Does the ontology provide enough or appropriate vocabulary and relationships to annotate Web pages? This is again fitness for purpose. The existing concepts and the relationships provide enough information to perform the transformations proposed in the previous chapter. However, the ontology can be easily extended for different purposes. This ontology provides a generic and flexible framework. Therefore, for specific kinds of pages (e.g., e-commerce, blogging, etc.,), specific ontologies can be modelled by extending the WAfA ontology.

Several aspects of the annotation process discussed above aims to clarify the relationship between the annotation and the transformation processes. In fact, the issues discussed above demonstrate that an evaluation of the annotation component of Dante cannot be done on its own. As a conclusion, the quality and quantity of the annotations depend on the transformation techniques that will be performed.

6.3 Summary and Conclusions

This chapter has presented the evaluation of Dante. Fundamentally, the evaluation consists of two parts: user and technical evaluation. The aim of the user evaluation was to demonstrate that the transformations enhanced the mobility of the users. Similarly, the aim of the technical evaluation was to demonstrate the utility of the annotations and the WAfA ontology.

The user evaluation was successful in demonstrating the usability of the transformations. The TLX and structured interview results, and also participants' comments show that transformations enhanced the mobility of the participants. The evaluation also provided useful information on flaws of the physical fragmentation and the generated table of contents. Mainly, the results suggest that another follow-up evaluation can

be useful to further investigate the issues highlighted in this evaluation. Particularly, the following six issues need to be addressed:

1. *Different transformations* Dante supports a number of transformations. In this evaluation, we only considered "physical fragmentation". Considering different transformations will enable us to address questions such as: When do we perform these transformations?; Does it depend on the task or users' purpose? or Does it depend on the Web page? or Does it need to be just users' preferences?

2. *Options of transformations* For some of the above transformations, Dante provides several options such as removing certain elements or changing their positions. Therefore, we still need to address questions such as Which one is better? Does it depend on the task? Which one is most preferred compared to the others? Does it only depend on the user's preferences or does it depend on the user's task?

3. *Different Web sites* Do different Web sites require different transformations. For example, fragmenting search results can be more useful than removing the header and footer in the page.

4. *Different tasks* Do different tasks require different transformations? Can participants match their tasks to transformations? Do certain tasks require certain levels of granularity?

5. *Different granularity* In this evaluation, we only compared original pages against a specific level of fragmentation. However, pages can be fragmented in different granularity and this needs to be further investigated. What is the right level of granularity for fragmentation? What is the optimum number of fragments? Does this depend on the task? Does this depend on the user's preferences? Does this depend on the kind of Web site?

6. *Experience of the user* Users access pages differently depending on their experiences. Does this mean different transformations can work better for different user groups (e.g., experienced, novice)?

Chapter 7

Summary and Conclusions

The Web has offered visually impaired people an unprecedented opportunity to have the same access to information as their sighted counterparts. However, the visual dominance on the Web has introduced an inequality. This thesis makes a significant contribution to redress this imbalance by providing a set of techniques and principles such that navigational and structural properties of Web pages can be described and used to present pages in alternative forms to provide better travel support.

This chapter concludes the thesis by summarising the work presented and assessing the extent to which the research questions set out in Chapter 1 have been addressed. First an overview of the thesis is presented, then the significance of the major results and the outstanding issues are discussed, and finally directions for future work are suggested.

7.1 Thesis Overview

For visually impaired users, travelling on the Web is difficult [61, 68, 2, 3, 33]. The objective of this thesis was to investigate techniques to improve the travel support for visually impaired Web users [53, 46]. The hypothesis was that if Web pages were analysed to identify objects that support travel, then these objects could be used to transform pages into alternative forms to enhance the travel support provided to visually impaired users.

This hypothesis has been supported by the use of a framework for the identification of structural and navigational entities and their properties of Web pages. These entities and properties were encapsulated in an ontology. This ontology, called WAfA, supports machine processing and has been used as part of a flexible pipeline approach to

annotate and transform Web pages. An experimental test-bed, Dante, has also been implemented which used this pipeline to encode a number of transformation techniques. Finally, a user evaluation method has been devised and applied which demonstrated that the travel experience of visually impaired users could be improved by adapting these techniques, and therefore, confirmed our hypothesis.

7.2 Contributions of the Thesis

The research questions stated in Chapter 1 have been successfully addressed by the following major contributions of the thesis:

(Q1) An extension and modification of the model of travel to provide a better understanding of how visually impaired people travel around Web pages [53];

(Q2) the identification of classes of objects that support travel on the Web;

(Q3) the development of a travel analysis framework to extract these objects from a Web page; the design and application of an evaluation procedure to evaluate this framework which has demonstrated that the encoded process is systematic and consistent;

(Q4) the development of the WAfA ontology that encoded knowledge about travel objects in a computationally available form;

(Q5) the identification of three levels of knowledge about Web pages: structural, navigational and contextual which were encoded as three sub-ontologies in WAfA;

(Q6) the design of a methodology which was based on a flexible knowledge-driven pipeline to annotate and transform Web pages by using WAfA and the development of an experimental prototype, Dante, based on this pipeline;

(Q7) the identification of a number of transformation heuristics that encapsulate different techniques to transform Web pages into alternative forms and the implementation of these heuristics in Dante that has shown the utility of WAfA;

(Q8) the design of an evaluation procedure to assess the travel support provided by a Web page;

(Q9) the evaluation of transformation techniques encoded in Dante based on this evaluation procedure which has shown that the travel experience of visually impaired Web users can be improved by these transformation techniques.

7.2.1 The Travel Analysis Framework

The model of travel proposed by Goble *et al.* [46] provides an understanding of how visually impaired people travel around Web pages. This model has been modified and extended to include a better understanding of the objects, called travel objects [125], used by travellers to complete a successful journey (see **Q2** on page 17). These objects have been categorised into three broad categories (1) way points, (2) orientation points and (3) travel assistants (see Table 3.1 on page 52).

A travel analysis framework has been introduced to analyse Web pages for their travel support. This framework has been based on the extended model of travel which has two stages (1) the analysis of Web pages to create an inventory of objects that support travel and (2) the classification of each object within that inventory regarding its role in a typical journey. A set of guidelines and questionnaires have also been created to support these two stages. Although Harper [53] had proposed a mobility analysis framework, his focus was on providing a mobility rating for a page. The process suggested by Harper was, however, inappropriate and insufficiently detailed to be used to inspect Web pages to extract travel objects.

In order to demonstrate that the process encoded in this new framework was systematic, an evaluation procedure was introduced and applied (see **Q3** on page 17). This evaluation was successful; the created travel object inventories were highly consistent between the participants and were able to demonstrate that the process encapsulated in the framework is systematic and consistent enough to be used as the basis of a tool [125].

7.2.2 The WAfA Ontology

In order to demonstrate the hypothesis, knowledge about travel objects need to be made explicit and computationally available. One possible way of associating information with the identified objects is annotation [51]. This requires capturing knowledge about these objects in a way that enables us to perform annotations. Ontologies

are a widely accepted way of capturing knowledge and of controlled vocabulary creation [14]. Therefore, the WAfA ontology has been created to encapsulate this knowledge in a machine processable form (see **Q4** on page 18).

Although the identified travel objects provided information regarding the travel support provided, an understanding of the underlying structure of the Web page was still lacking. Furthermore, it was observed that HTML does not provide enough information regarding the structural and navigational properties of Web pages. It provides a vocabulary that can be used to markup the physical and presentational properties, but not the navigational and structural properties relevant for travel. Hence we have proposed three levels of knowledge about Web pages: structural, navigational and contextual knowledge (see **Q5** on page 18) [124, 122, 123].

These three levels of knowledge have formed the three sub-ontologies of WAfA: authoring, mobility and contextual semantics. Mobility semantics encode knowledge about the classes of travel objects. Authoring and contextual semantics were based on the knowledge extracted from the Hypermedia authoring literature [72], transcoding literature [58, 24], mark-up languages [115], Web Accessibility work [87, 109], voice interfaces [91] and CSS[1] cohort, etc.

WAfA has been designed to provide machine interpretable semantic annotation support for making implicit travel support explicit. It has played a central role in our knowledge-driven pipeline Dante. Therefore, apart from technical evaluations, such as determining consistency, completeness and redundancy [47], it was meaningless to evaluate WAfA on its own. In this thesis, the evaluation of WAfA has been performed pragmatically within our experimental prototype, Dante, by assessing its competence in satisfying the requirements of the pipeline. WAfA was implemented, deployed, and evaluated incrementally to ensure that it was in a satisfactory form to support the annotation and transformation processes encoded in Dante.

7.2.3 Dante: Annotating and Transforming Web Pages

A knowledge-driven pipeline has been introduced to annotate Web pages with the WAfA ontology and transform them accordingly (see Figure 5.7 on page 96). This pipeline has provided the flexibility to allow annotations to arrive in different formats from different origins. It has enabled annotation and transformation processes to work

[1] See `http://www.w3.org/Style/CSS/`.

together but still to be loosely coupled enough to support the usage of alternative implementations for both components. This pipeline approach has shown how the WAfA ontology can be used to annotate Web pages (see **Q6** on page 19).

Dante, our experimental test-bed, has been designed and implemented based on this knowledge-driven pipeline. A survey revealed that most of the existing annotation tools focus on the annotation of the content rather than the structure of the pages. Furthermore, even though these annotation tools could be used to annotate Web pages, none of them addresses the issue of creating relationship(s) between annotations. Therefore, in Dante an existing tool, the COHSE annotator, has been used and another tool, called the Dante RDF editor, has been developed to create relationship(s) between annotations. The properties in WAfA and the unique identifiers generated by the COHSE annotator have been used to create RDF triples to represent relationships between annotations (e.g., *has-direct-part*, *has-heading*, etc) (see Section 5.3 on 97). These RDF triples and annotations were then used by the Dante transformer to present pages in different ways. A number of transformation heuristics have been introduced and implemented in Dante to mainly address both mobility principles and instruments (Table 5.3 on page 102).

These heuristics have shown the utility of WAfA annotations in the transformation process (see **Q7** on page 20). The experiments performed with these heuristics have also shown that annotating Web pages, both with mobility and authoring semantics, can be redundant. Annotating pages only with authoring semantics provide enough information to perform transformations. However, mobility semantics have been used implicitly in the entire process of annotation and transformation. Further experimentation with different pages has also demonstrated that the quality and the quantity of the annotations depend on both the transformation heuristics performed and the complexity of the page.

The WAfA ontology has provided a more extensive and expressive vocabulary for markup than has gone before. Furthermore, Dante is based on a model of travel and has focused on addressing the differences between audio and visual navigation, often neglected in most Web accessibility work [87, 109, 8, 9].

In order to demonstrate that the transformed versions of Web pages provide better travel support, a method was needed to evaluate the provided travel support (see **Q8** on page 20). Therefore, a task based user evaluation procedure has been developed [120, 57, 38] and has been used to evaluate Dante. In summary, participants have been asked to perform a number of tasks on both the original and the transformed versions of a

number of pages. The results have confirmed that participants perform better on the transformed pages than the original versions (see **Q9** on page 20).

7.3 Outstanding Issues

Despite the contributions arising from this work, there are some outstanding issues which are discussed below.

Manual annotations In the experimental prototype of Dante, Web pages have been annotated manually by using the COHSE annotator. Although this was used to demonstrate the utility of WAfA annotations, manual annotation is an expensive and time-consuming process. Moreover, even small changes on a page can easily invalidate the annotations (see Section 5.4.4 on page 112) and dynamically generated pages cannot be annotated. Therefore, techniques need to be further investigated to automate the annotation process (see Figure 7.1).

Design methodologies [95] presents how WAfA annotations can be generated with a design methodology called WSDM. There are a number of other methodologies for designing Web pages (WebML [23], OOHDM [100], etc.) which can also be extended with a similar approach to generate WAfA annotations.

Content Management Systems (CMS) Manage the content creation and delivery processes of a Web site. A pilot CMS can be extended to assess the viability of generating WAfA annotations automatically.

Annotation Heuristics A number of guidelines have been included in the travel analysis framework to identify travel objects. These guidelines can be used as the basis of a set of heuristics to identify travel objects automatically. Generic heuristics can be created for different kinds of pages or private heuristics can be created for specific pages.

Combination of Heuristics In Dante, transformations have been performed individually. However, as discussed in Chapter 5, some of the transformation heuristics can be applied together. A systematic technique can be further investigated to combine these heuristics. This also requires the technical limitations of annotating resources with XPointers to be addressed.

Re-implementation of the Dante pipeline The Dante pipeline has been implemented within the Mozilla platform. However, it can be re-implemented as a plug-in to other browsers such as IE. As highlighted in Chapter 6, some screen readers work better with certain browsers. For example, Jaws works better with IE. In addition to this alternative browser extension, Dante can also be implemented on the server side or as a proxy server. A browser implementation will always have the shortcoming of requiring users to have experience with that particular browser. A proxy implementation will have the advantage of being platform and browser independent. Figure 7.1 depicts how Dante can be re-implemented either on the server side, client side or as an intermediary service such as a proxy server.

User Evaluation The user evaluation conducted, has focused on a particular transformation technique. As it is discussed in Section 6.3 (page 140), it is important to have a follow-up evaluation to address different transformations, options of transformations, different Web sites, different tasks, different granularity and the experience of the participants.

7.4 Future Work

The contributions of this thesis and the limitations discussed above have suggested several promising paths for future research.

Classification of Web pages Amitay *et al.* (2003) classify Web sites into seven types depending on their functionality (e.g., corporate, search engine, etc.) [6]. Their hypothesis is that sites with a similar role exhibit similar structural patterns. A similar hypothesis would imply the viability of categorising Web pages with respect to their navigational and structural properties using the WAfA ontology.

Using Dante in a Different Context Although the Dante approach, particularly the annotation pipeline, has been proposed for visually impaired users other user groups may also benefit. For example, Web pages can be annotated and transformed to meet the requirements of small screen device users or to support internationalisation[2]. The WAfA ontology can also be extended to address the requirements of small screen devices. For example, Hori *et al.* (2000) use a

[2]See `http://www.w3.org/International/`.

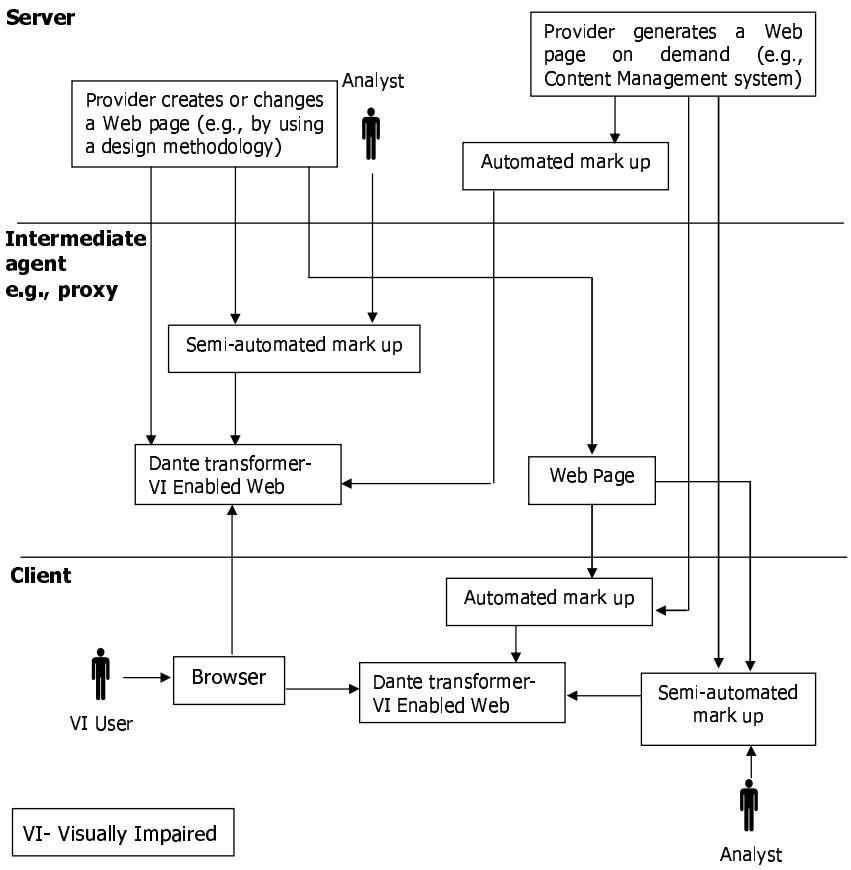


Figure 7.1: Possible applications of Dante.

vocabulary (e.g., fidelity, device capability, etc.) to annotate Web pages in order to be able to adapt for small screen devices [59]. This vocabulary can easily be integrated into the WAfA ontology.

Reasoning Support When we annotate a Web page with the WAfA ontology, we actually create an instance store for the WAfA ontology. Therefore, the functionalities of instance stores[3] can be further investigated to support better reasoning for transformation techniques.

Extending Transformation Heuristics The transformation heuristics presented can easily be extended or new heuristics can be introduced. The WAfA ontology provides extensive knowledge about the page structure which can easily be used in different ways. For example, different summarisation techniques can be applied by using certain elements (e.g., `Chunk`, `LinkMenu`, etc.). Moreover,

[3] An application that supports reasoning over individuals, see `http://instancestore.man.ac.uk/`.

transformation heuristics can also be introduced for specific kinds of pages, for example in Web directory pages, a possible transformation heuristic would be to remove all the objects in the page in order to leave only the `WebDirectory` object, this will allow users to easily access that object.

Using Contextual Semantics Although some concepts, called *contextual semantics*, have been defined in the WAfA ontology they have not been used in Dante. Therefore, a possible extension to the transformation techniques would be using this sub-ontology to capture more information about the users' purpose and then to transform pages accordingly (see Section 4.3.3 on 84).

Reuse Annotations and the Model of Web Pages [107] proposes techniques to reuse annotations for one page on another page that has a similar structure. This technique could be extended by using the object inventory created by the WAfA ontology to compare pages for their structural and navigational similarities. If these pages have a similar structure and navigation pattern, then the annotations could be migrated to those pages as well.

Even though this chapter concludes this thesis, in fact it does not provide an 'end' to the research but it provides a 'beginning' to further research. The Web plays an important role in many areas of our lives (e.g., education, employment, government, etc.) and as Henry states "an accessible Web that allows people with disabilities to actively participate in society is essential for equal opportunities in many areas" [4] [109]. This thesis provides encouraging results to realise this by solving the problems that discourage visually impaired users from accessing Web pages. The work presented in this thesis will provide a firm foundation for ongoing research into supporting better Web accessibility for visually impaired users.

[4] See `http://www.w3.org/WAI/intro/accessibility`.

Bibliography

[1] Tablin: an HTML table linearizer. `http://www.w3.org`, 2000.

[2] A report into key government web sites. Technical report, London, 2002. `http://www.iablondon.com/`.

[3] The web: Access and inclusion for disabled people. Technical report, Disability Rights Commission (DRC), 2004.

[4] WebCleaner, 2004. `http://webcleaner.sourceforge.net`.

[5] M. Abe and M. Hori. Robust pointing by XPath language: Authoring suport and empirical evaluation. In *Proceedings of the 2003 Symposium on Applications and Internet*. IEEE Computer Society, 2003.

[6] E. Amitay, D. Carmel, A. Darlow, R. Lempel, and A. Soffer. The connectivity sonar: detecting site functionality by structural patterns. In *Proceedings of the Fourteenth ACM Conference on Hypertext and Hypermedia*, pages 38–47. ACM Press, 2003.

[7] P. Arthur and R. Passini. *Wayfinding: People, Signs, and Architecture*. McGraw-Hill, 1992.

[8] C. Asakawa and T. Itoh. User interface of a home page reader. In *ASSETS'98*, pages 149–156. ACM Press, 1998.

[9] C. Asakawa and H. Takagi. Annotation-based transcoding for nonvisual web access. In *ASSETS'00*, pages 172–179. ACM Press, 2000.

[10] P. Atzeni, G. Mecca, and P. Merialdo. To weave the web. In *Proceedings of the 23rd International Conference on Very Large Data Bases*, pages 206–215, 1997.

[11] S. Bechhofer, L. Carr, C. Goble, S. Kampa, and T. Miles-Board. The semantics of semantic annotation. In *ODBASE: First International Conference on Ontologies, DataBases, and Applications of Semantics for Large Scale Information Systems*, California, 2002.

[12] S. Bechofer and C. Goble. *COHSE: Conceptual Open Hypermedia Service*, volume 96 of *Frontiers in Artificial Intelligence and Applications*, pages 193–211. IOS Press, 2003.

[13] T. Berners-Lee. *Weaving the Web*. Texere Publishing, 1999.

[14] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):34–43, May 2001.

[15] T. Bickmore and B. Schilit. Digestor: Device-independent access to the World Wide Web. In *Proceedings of the Sixth International World Wide Web Conference*, pages 655–663, 1997.

[16] B. Blasch, W. Wiener, and R. Welsh, editors. *Foundations of Orientation and Mobility*. AFB Press, American Foundation for the Blind, second edition, 1997.

[17] D. Boswell, B. King, I. Oeschger, P. Collins, and E. Murphy. *Creating Applications with Mozilla*. O'Reilly, 2002.

[18] M. R. Boyns. Design and implementation of a World Wide Web filtering system. Master's thesis, San Diego State University, 1998.

[19] M. Brambring. Mobility and orientation process of the blind. In *Proceedings of the NATO Advanced Research Workshop*, pages 493–508, 1984.

[20] S. Brown and P. Robinson. A world wide web mediator for users with low vision. In *CHI 2001 Workshop, Universal Design: Towards Universal Access in the Information Society*, 2001.

[21] O. Buyukkokten, H. Molina, and A. Paepcke. Accordion summarization for end-game browsing on PDAs and cellular phones. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 213–220. ACM Press, 2001.

[22] O. Buyukkokten, H. Molina, A. Paepcke, and T. Winograd. Power browser: Efficient web browsing for PDAs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 430–437. ACM Press, 2000.

[23] S. Ceri, P. Fraternali, and A. Bongio. Web modeling language (WebML) : a modeling language for designing web sites. In *Proceedings of the Ninth World Wide Web Conference*, 2000.

[24] J. Chen, B. Zhou, J. Shi, H. Zhang, and Q. Wu. Function-based object towards website adaptation. In *Proceedings of the Tenth International World Wide Web Conference*, Hong Kong, 2001. ACM.

[25] Y. Chen, W. Ma, and H. Zhang. Detecting web page structure for adaptive viewing on small form factor devices. In *Proceedings of the Twelfth International World Wide Web Conference*, 2003.

[26] W. Chisholm and S. Palmer. Evaluation and report language (EARL) 1.0, 2002. http://www.w3.org/TR/EARL10/.

[27] W. Chisholm, G. Vanderheiden, and I. Jacobs. Web content accessibility guidelines 1.0. World Wide Web Consortium (W3C), 1999. http://www.w3.org/TR/1999/WAI-WEBCONTENT/.

[28] Codix. *Textualize*, 2003. `http://codix.net/solutions/products/textualise/index.html`.

[29] J. Coombs, A. Renear, and S. DeRose. Markup systems and the future of scholarly text processing. *Communications of the ACM*, 30(11):933–947, 1987.

[30] O. Corcho, M. Fernandez-Lopez, and A. Gomez-Perez. Methodologies, tools and languages for building ontologies: Where is their meeting point? *Data and Knowledge Engineering*, 46(1):41–64, 2003.

[31] J. Cove and B. Walsh. Online text retrieval via browsing. *Information Processing & Management*, 24(1):31–37, 1998.

[32] K. P. Coyne and J. Nielsen. Beyond ALT text: Making the web easy to use for users with disabilities. Nielson Norman Group, 2001.

[33] J. Craven and P. Brophy. Non-visual access to the digital library: the use of digital library interfaces by blind and visually impaired people, 2003. Library and Information Commission Research Report 145.

[34] D. Dardailler. The ALT-server (an eye for an alt), 1997. `http://www.w3.org/WAI/altserv.htm`.

[35] R. Darken and B. Peterson. Spatial orientation, wayfinding and representation. In K. Stanney, editor, *Handbook of Virtual Enviroment Technology*, chapter 28. Lawrence Erlbaum Associates, Inc, 2002. `http://vehand.engr.ucf.edu/handbook/Chapters/Chapter28/Chapter28.html`.

[36] R. Darken and J. Sibert. Navigating large virtual spaces. *International Journal of Human-Computer Interaction*, 8(1):49–72, 1996.

[37] A. Dillon, C. McKnight, and J. Richardson. *Hypertext: A Psychological Perspective*, chapter Why Physical Representations are not Semantic Intentions, pages 169–191. 1993.

[38] A. Dix, J. Finlay, G. Abowd, and R. Beale. *Human-Computer Interaction, 3/E*. Prentice Hall, 2004.

[39] C. Earl and J. Leventhal. A survey of windows screen reader users: Recent improvements in accessbility. *Journal of Visual Impairment and Blindness*, 93(3), 1999.

[40] A. Edwards and R. Stevens. Visual dominance and the World Wide Web. In *Proceedings of the Sixth International World Wide Web Conference*, 1997.

[41] R. Edwards, S. Ungar, and M. Blades. Route descriptions by visually impaired and sighted children from memory and from maps. *Journal of Visual Impairment and Blindness*, 92(7):512–521, 1998.

[42] M. Fernandez-Lopez. Overview of methodologies for building ontologies. In *IJCAI99 Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends*, 1999.

[43] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider. Sweetening ontologies with DOLCE. In *EKAW*, pages 166–181. Springer-Verlag, 2002.

[44] F. Gatzemeier and O. Meyer. Making complex document structures accessible through templates. In *Proceedings of IEEE professional communication society international professional communication conference and Proceedings of the 18th annual ACM international conference on Computer documentation*, pages 509–519. IEEE Educational Activities Department, 2000.

[45] C. Goble, S. Bechhofer, L. Carr, D. Roure, and W. Hall. Conceptual open hypermedia = the semantic web? In *SemWeb2001, The Second International Workshop on the Semantic Web*, Hong Kong, May 2001.

[46] C. Goble, S. Harper, and R. Stevens. The travails of visually impaired web travellers. In *The Eleventh International ACM Conference on Hypertext and Hypermedia*, pages 1–10, 2000.

[47] A. Gómez-Pérez. Evaluation of ontologies. *International Journal of Intelligent Systems*, 16(3):391–409, 2001.

[48] T. Gruber. Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, 1993.

[49] J. Gunderson and I. Jacobs. User agent accessibility guidelines 1.0. World Wide Web Consortium (W3C), 1999. `http://www.w3.org/TR/WAI-USERAGENT/`.

[50] N. Hammond and L. Allinson. Travels around a learning support environment: Rambling, orienteering or touring. In *CHI'88 Conference Proceedings: Human Factors in Computer Systems*, pages 269–273, 1987.

[51] S. Handschuh and S. Staab. *Annotation for the Semantic Web*, volume 96 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2003.

[52] W. Harpe, J. Kingsbury, and J. Hassell. Accessibility study of BBCi: Problems faced by users with disabilities. Technical report, System Concepts Ltd., 2003.

[53] S. Harper. *Web Mobility for Visually Impaired Surfers*. PhD thesis, The University of Manchester, 2001.

[54] S. Harper, C. Goble, and R. Stevens. A pilot study to examine the mobility problems of visually impaired users travelling the web. *ACM SIGCAPH: The*

ACM Special Interest Group on Computers and the Physically Handicapped Newsletter, 68, 2000.

[55] S. Harper, R. Stevens, and C. Goble. Towel: Real world mobility on the web. In J. Vanderdonckt and A. Puerta, editors, *Computer-Aided Design of User Interfaces II*, pages 305–314. Kluwer Academic, 1999.

[56] S. Harper, R. Stevens, and C. Goble. Web mobility guidelines for visually impaired surfers. *Journal of Research and Practice in Information Technology Special Issue on HCI*, 33(2), July 2001.

[57] S. Hart and L. Staveland. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In P. A. Hancock and N. Meshkati, editors, *Human Mental Workload*, pages 139–183. North-Holland: Elsevier Science, 1988.

[58] M. Hori, G. Kondoh, and K. Ono. Annotation-based web content transcoding. In *Proceedings of the Nineth International World Wide Web Conference*, pages 197–211, 2000.

[59] M. Hori, G. Kondoh, K. Ono, S. Hirose, and S. Singhal. Annotation-based web content transcoding. In *Proceedings of the Nineth International World Wide Web Conference*, 2000.

[60] M. Hori, K. Ono, T. Koyanagi, and M. Abe. Annotation for transformation for the automatic generation of content customization metadata. In F. Mattern and M. Naghshineh, editors, *Pervasive*, volume 2414 of *Lecture Notes in Computer Science*. Springer, 2002.

[61] J. Howell. Get the message online. Royal National Institute of Blind (RNIB), 2000.

[62] R. Ingram and S. Benford. Improving the legibility of virtual environments. In *Proceedings of the Second Eurographics Conference on Virtual Environments*, 1995.

[63] M. Ivory and M. Hearst. The state of the art in automating usability evaluation of user interfaces. *ACM Computer Survey*, 33(4):470–516, 2001.

[64] M. Ivory, J. Mankoff, and A. Le. Using automated tools to improve web site usage by users with diverse abilities. *IT&Society, Special Issue on Web Navigation Skills*, 3(1):195–136, 2003.

[65] M. Jones, G. Marsden, N. Nasir, K. Boone, and G. Buchanan. Improving Web interaction on small displays. *Computer Networks (Amsterdam, Netherlands: 1999)*, 31(11–16):1129–1137, 1999.

[66] S. Jul and G. Furnas. Navigation in electronic worlds: a CHI 97 workshop. *ACM SIGCHI Bulletin*, 29(4):44–49, 1997.

[67] J. Kahan and M. Koivunen. Annotea: an open RDF infrastructure for shared web annotations. In *Proceedings of the Tenth International World Wide Web Conference*, pages 623–632, 2001.

[68] B. Kelly. Webwatch: An accessibility analysis of UK university entry points. Technical report, The University of Bath- Ariadne Issue 33, 2002. `http://www.ariadne.ac.uk/issue33/web-watch/`.

[69] A. Kennel, L. Perrochon, and A. Darvishi. WAB: World Wide Web access for blind and visually impaired computer users. *ACM SIGCAPH Computers and the Physically Handicapped*, (55):10–15, 1996.

[70] S. Kurniawan and P. Zaphiris. Usability and accessibility comparison of governmental, organizational, educational and comercial aging/health related web sites. *WebNet Journal*, 3(3), 2001.

[71] R. Long and E. Hill. Establishing and maintaining orientation for mobility. In B. Blasch, W. Wiener, and R. Welsh, editors, *Foundations of Orientation and Mobility*, chapter 2, pages 39–59. AFB Press, American Foundation for the Blind, second edition, 1997.

[72] D. Lowe and W. Hall. *Hypermedia and the Web: An Engineering Approach*. John Wiley and Sons Ltd, 1998.

[73] K. Lynch. *The Image of the City*. The MIT Press, 1960.

[74] G. Marchionini. *Information Seeking in Electronic Environments*. Cambridge Series on Human-Computer Interaction. Cambridge University Press, 1995.

[75] D. McGuinness. Ontological issues for knowledge-enhanced search. In *FOIS-98*, 1998.

[76] D. McGuinness and F. Harmelen. OWL web ontology language reference. World Wide Web Consortium (W3C), 2004. http://www.w3.org/TR/2004/REC-owl-features-20040210/.

[77] C. McKnight, A. Dillon, and J. Richardson. *Hypertext in Context*. Cambridge University Press, 1991.

[78] G. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The Psychological Review*, 63:81–97, 1959.

[79] S. Murphy. The paragraph: the weak link in technical communication? In *Proceedings of IEEE professional communication society international professional communication conference and Proceedings of the 18th annual ACM international conference on Computer documentation*, pages 477–482. IEEE Educational Activities Department, 2000.

[80] C. Musciano and B. Kennedy. *HTML and XHTML: The Definitive Guide*. O'Reilly, 2002.

[81] W. Myers. *BETSIE:BBC Education Text to Speech Internet Enhancer*. British Broadcasting Corporation (BBC) Education, 2002. `http://www.bbc.co.uk/education/betsie/`.

[82] J. Nielsen. Inverted pyramids in cyberspace. Alertbox, March 1996. `http://www.useit.com/alertbox/9606.html`.

[83] J. Nielsen. How users read on the web. Alertbox, April 1997. `http://www.useit.com/alertbox/9710a.html`.

[84] J. Nielsen. Alternative interfaces for accessibility. Alertbox, April 2003. `http://www.useit.com/alertbox/20030407.html`.

[85] J. Nielsen. Voice interfaces: Assessing the potential. Alertbox, January 2003. `http://www.useit.com/alertbox/20030127.html`.

[86] N. Noy and D. McGuinness. Ontology development 101: A guide to creating your first ontology. Ksl-01-05, Stanford Knowledge Systems Laboratory, March 2001.

[87] M. Paciello. *Web accessibility for people with disabilities*. CMP books, CMP media LLC, 2000.

[88] R. Passini. *Wayfinding in Architecture*. Van Nostrand Reinhold, New York, 1984.

[89] R. Passini, J. Delisle, C. Langlois, and G. Proulx. Wayfinding information for congenitally blind individuals. *Journal of Visual Impairment and Blindness*, pages 425–429, 1988.

[90] R. Passini and G. Proulx. Wayfinding without vision; an experiment with congenitally totally blind. *Environment and Behaviour*, 20(2):227–252, 1988.

[91] M. Perez-Quinones, R. Capra, and Z. Shao. The ears have it: A task by information structure taxonomy for voice access to web pages. In *Proceedings of the Ninth IFIP TC13 International Conference on Human-Computer Interaction*, 2003.

[92] H. Pinto, S. Staab, and C. Tempich. DILIGENT: Towards a fine-grained methodology for distributed loosely-controllled and evolving engineering of ontologies. In *Proceedings of ECAI-2004*, 2004.

[93] I. Pitt and A. Edwards. An improved auditory interface for the exploration of lists. In *Proceedings of the fifth ACM international conference on Multimedia*, pages 51–61. ACM Press, 1997.

[94] I. Pitt and A. Edwards. *Design of Speech-Based Devices - a Practical Guide*. Springer, London, Great Britain, 2003.

[95] P. Plessers, S. Casteleyn, Y. Yesilada, O. D. Troyer, R. Stevens, S. Harper, and C. Goble. Accessibility: A web engineering approach. In *Proceedings of the Fourteenth International Conference on World Wide Web*, pages 353–362, 2005.

[96] I. Ramakrishnan, A. Stent, and G. Yang. Hearsay: Enabling audio browsing on hypertext content. In *Proceedings of the Thirteenth International Conference on World Wide Web*, 2004.

[97] T. V. Raman. Emacspeak - a speech interface. In *Conference Proceedings on Human Factors in Computing Systems*, pages 66–71. ACM Press, 1996.

[98] T. V. Raman. Emacspeak - direct speech access. In *ASSETS'96*, pages 32–36. ACM Press, 1996.

[99] J. Richards and V. Hanson. Web accessibility: A broader view. In *Proceedings of the Thirteenth International World Wide Web Conference*, page ACM Press, 2004.

[100] D. Schwabe, G. Rossi, and S. Barbosa. Systematic hypermedia application design with OOHDM. In *Proceedings of the the seventh ACM conference on Hypertext*, 1996.

[101] L. Seeman. The semantic web, web accessibility, and device independence. In S. Harper, Y. Yesilada, and C. Goble, editors, *Proceedings of the international cross-disciplinary workshop on Web accessibility*, pages 67–73, 2004.

[102] J. Smith, R. Mohan, and C. Li. Transcoding internet content for heterogeneous client devices. In *Proceedings of IEEE International Conference On Circuits and Systems*, California, 1998.

[103] R. Stevens. *Principles for the Design of Auditory Interfaces to Present Complex Information to Blind People*. PhD thesis, Department of Computer Science, The University of York, York, UK, 1996.

[104] R. Stevens and A. Edwards. An approach to the evaluation of assistive technology. In *ASSETS'96*, pages 64–71. ACM Press, 1996.

[105] R. Stone and J. Dhiensa. Proving the validity and accessibility of dynamic webpages. In S. Harper, Y. Yesilada, and C. Goble, editors, *Proceedings of the international cross-disciplinary workshop on Web accessibility*, pages 45–49, 2004.

[106] H. Takagi and C. Asakawa. Transcoding proxy for nonvisual web access. In *ASSETS'00*, pages 164–171. ACM Press, 2000.

[107] H. Takagi, C. Asakawa, K. Fukuda, and J. Maeda. Site-wide annotation: Reconstructing existing pages to be accessible. In *ASSETS'02*, pages 81–88. ACM Press, 2002.

[108] H. Takagi, C. Asakawa, K. Fukuda, and J. Maeda. Accessibility designer: Visualizing usability for the blind. *ASSETS'04*, pages 177–184, 2004.

[109] J. Thatcher, P. Bohman, M. Burks, S. Henry, B. Regan, S. Swierenga, M. Urban, and C. Waddell. *Constructing Accessible Web Sites*. Glasshaus, 2002.

[110] J. Treviranus, C. McCathieNevile, I. Jacobs, and J. Richards. Authoring tool accessibility guidelines 1.0. World Wide Web Consortium (W3C), 2000. `http://www.w3.org/TR/ATAG10/`.

[111] UsableNet. *LIFT Text Transcoder*, 2003. `http://www.usablenet.com/`.

[112] M. Uschold and M. Gruninger. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 11(2):93–15, 1996.

[113] J. Vanderdonckt, A. Beirekdar, and M. Noirhomme-Fraiture. Automated evaluation of web usability and accessibility by guideline review. In *Web Engineering - 4th International Conference, ICWE 2004 Proceedings (LNCS 3140)*, pages 17–30. Springer, 2004.

[114] M. Vorburger. Altifier web accessibility enhancement tool, 1999. `http://www.vorburger.ch/projects/alt/`.

[115] N. Walsh and L. Muellner. *DocBook: The Definitive Guide*. O'Reilly, 1999.

[116] Y. Whang, C. Jung, J. Kim, and S. Chung. Webalchemist: A web transcoding system for mobile web access in handheld devices. In *Optoelectronic and Wireless Data Management, Processing, Storage, and Retrieval*, pages 102–109, 2001.

[117] M. Winston, R. Chaffin, and D. Herrmann. A taxonomy of part-whole relations. *Cognitive Science*, 11:417–444, 1987.

[118] P. Wright. Tables in text: the subskills needed for reading formatted information. In L. Chapman, editor, *The Reader and The Text*, pages 60–69. London:Heineman, 1981.

[119] P. Wright and K. Fox. Presenting information in tables. *Applied Ergonomics*, 1:331–343, 1970.

[120] P. Wright and A. Monk. A cost effective evaluation method for use by designers. *International Journal of Man-Machine Studies*, 35(6):891–9120, 1991.

[121] Y. Yesilada. Browsing tables when you cannot see them. Master's thesis, The University of Manchester, 2000.

[122] Y. Yesilada, S. Harper, C. Goble, and R. Stevens. Ontology based semantic annotation for enhancing mobility support for visually impaired web users. In *K-CAP 2003 Workshop on Knowledge Markup and Semantic Annotation*, 2003.

[123] Y. Yesilada, S. Harper, C. Goble, and R. Stevens. Dante annotation and transformation of web pages for visually impaired users. In *Proceedings of the Thirteenth International Conference on World Wide Web*, 2004.

[124] Y. Yesilada, S. Harper, C. Goble, and R. Stevens. Screen readers cannot see (ontology based semantic annotation for visually impaired web travellers). In N. Koch, P. Fraternali, and M. Wirsing, editors, *Web Engineering - 4th International Conference, ICWE 2004 Proceedings (LNCS 3140)*, pages 445–458. Springer, 2004.

[125] Y. Yesilada, R. Stevens, and C. Goble. A foundation for tool based mobility support for visually impaired web users. In *Proceedings of the Twelfth International Conference on World Wide Web*, pages 422–430, 2003.

[126] Y. Yesilada, R. Stevens, C. Goble, and S. Hussein. Rendering tables in audio: The interaction of structure and reading styles. In *ASSETS'04*, 2004.

[127] X. Yin and W. Lee. Using link analysis to improve layout on mobile devices. In *Proceedings of the Thirteenth International World Wide Web Conference*, pages 338–344, 2004.

[128] M. Zajicek, C. Powell, and C. Reeves. Evaluation of a World Wide Web scanning interface for blind and visually impaired users. In *HCI '99*, volume 2, pages 980–984, 1999.

Appendix A

The Travel Analysis Framework

A.1 The Evaluation Materials

1	How was your understanding of travel on the web before the analysis? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
2	How is your understanding of travel on the web after the analysis? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
3	How easy did you find the overall framework? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
4	How easy did you find first part which is creating the travel object inventory? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
5	How easy did you find the classification of travel objects (Answering the questionnaire) ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
6	How difficult did you find answering the questions as ‘yes’ or ‘no’? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
7	How confident were you during the analysis? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
8	How subjective did you find the questionnaire (Classifications of the travel objects)? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
9	How useful did you find the guidelines? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
10	How useful did you find the examples? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good
11	Did you use the definitions of the classifications of the travel objects? ☐ Not good at all ☐ Not very good ☐ Moderately good ☐ Quite good ☐ Very good

Table A.1: Questions for the overall evaluation of the framework.

Role	Questions		Travel Objects	
Decision Points	Does it provide a set of options (more than one) for making a decision about a journey?			
WayEdge	Does it provide a kind of border to another object?			
	Is it used to break continuity?			
	Is it used to separate two regions of the page?			
	Is it a linear or continuous element?			
	Is it a linear element that serves as dividing line between parts of the environment?			
Navigation Points	Can it be used to move from one place to another in the environment?			
	Can it be used to move either within a page or between pages?(Where can I go?)			
	Does it provide a possible route that a traveller exercises some control by choosing to follow or not to follow it?			
Reference Points (Landmarks)	Does it help the travellers place themselves within the environment? (Does it help the travellers know where they are in the environment?)			
	Can it be considered as a distinctive, recognizable or unique feature of the environment?			
	Would it be easily identified in the environment for any of the following reasons?(Size, color, architectural design, shape, physical properties, frequently used, function or meaning) (i.e Can it be easily differentiated in the environment from any of th			
	Is it a salient object (strong or prominent)?			
	Can it be considered as memorable or easy to memorize object? (After the traveller leaves the environment, is it likely that he/she will remember this object and it's location in the environment. Thus when he/she will revisit, will this object help him/he			
	Reference Point Component	Can it be combined with other travel objects which allow a traveller to determine his (her) exact (precise) position in the environment?		
		Does it provide sufficient information to identify the exact position of the traveller itself?		
		Is it adequate for determining one's position in the environment?		
Direction	Does this object show which direction the traveller is moving along? (Does it show the current direction?)			
	Does this object show which direction the traveller can move along? (Does it show the possible directions that the traveller can move along?)			
Distance	Does it show how far is the traveller from the starting point?			
	Does it show how far is the traveller from a destination?			
Location&Position	Does it **explicitly** show where the traveller is within the environment? (For example: It could be an object that shows where the traveller is in the active site, for instance,			
	Does it **explicitly** help the traveller keep track of the position in the environment?			
Identification Point	Does it identify a part of the environment or another travel object?			
	Is it used to describe something?			
	Can it be used to identify the destination on arrival?			
	Is it a distinguishable property of a part of the environment or another travel object?			
Alert	Does it show a change in the environment?			
	Does it notify the user of approaching action or danger?			
	Does it issue a warning to the traveller?			
Attention	Does it attract the traveller's attention?			
	Does it have a different context from the general context of the environment?			
	Does it change the focus of a traveller?			
Information Point	Can the traveller ask for detail information about the environment?			
	Does it provide a medium for direct request of information (two-way communication)? (Information about			
	Does the traveller control the type and amount of information request from this object?(Information about			
	Does the user need to initiate the request for information?			
Travel Aid	Can it be considered as a passive information supply? (The traveller is not involved in the information supply process) & (Information about environment)			
	Does it provide an overview of the environment?			
	Is the traveller able to control what information is revealed?			
Travel Memory	Does it show where the traveller has been?			
	Does it show previously visited places?			
	Does it show how the traveller can get back to where he (she) was?			
	Does the object record information about the travel undertaken so far?			
Travel Support	Does it provide an automated journey?			

Figure A.1: The travel object classification questionnaire.

Appendix B

The WAfA Ontology

B.1 Authoring Semantics

The root concept for this sub-ontology is `AuthoringConcept`. This concept has four children: `Atom`, `Chunk`, `Node` and `Collection`. Table B.1 shows a hierarchical view of the high-level concepts of this part of the ontology. The concept `Node` represents a Web page and has only one child which is `HomePage` and the concept `Collection` represents a Web site. The concepts `Atom` and `Chunk` subsume a number of concepts which are explained in Section B.1.1 and Section B.1.2 respectively. There are also a number of concepts that are included in the ontology to define `AuthoringConcept`s. These concepts are: `Connection`, `Coverage`, `Position`, `Order` and `Structure`. Each of these concepts and their children are explained in Section B.1.3[1].

B.1.1 The Concept 'Atom'

This concept represents a coherent object that cannot be logically decomposed. Table B.2 shows a hierarchical view of the children of this concept which are explained as follows (The prefix "wafa" represents the URI of the WAfA ontology):

Advertisement Is usually a graphical, animated (dynamic) object that is used to advertise another page and provides link to that page.

```
Class(wafa:Advertisement partial wafa:Atom
  restriction(wafa:has-part someValuesFrom wafa:Link))
```

[1]For some of the descriptions, WordNet (`http://www.cogsci.princeton.edu/~wn/`) and Dictionary.com (`http://dictionary.reference.com/`) are used.

Part of the Ontology	**Concept**(Children of **`Thing`**)	**Page No.**
	`Object`	
Authoring Semantics:	`AuthoringConcept`	
	`Atom`	Page 165
	`Chunk`	Page 170
	`Node`	
	`Collection`	
Defining Authoring Semantics:	`Connection`	Page 176
	`Coverage`	Page 176
	`Position`	Page 177
	`Order`	Page 178
	`Structure`	Page 178

Table B.1: The overview of the Authoring Semantics.

AdvertisementBanner A banner that is used to advertise another page (`Node`) or site (`Collection`). This concept is also a child of the `Banner` concept.

Caption A title, short explanation or description accompanying an illustration for example a table or a figure.

FigureCaption A caption particularly used to describe or identify a figure.

Class(wafa:`FigureCaption` partial wafa:`Caption`
restriction(wafa:*is-caption-of* allValuesFrom wafa:`Figure`))

TableCaption A caption particularly used to describe or identify a table.

Class(wafa:`TableCaption` partial wafa:`Caption`
restriction(wafa:*is-caption-of* allValuesFrom wafa:`Table`))

Heading Indicates what the passage below is about (e.g., H1-H6 HTML tags).

Headline Is a widely known concept in newspapers; it is usually set in large type to indicate an important or sensational piece of news. In the context of Web pages, it is the highlighted heading which can be the latest updated or the most important heading in the page.

SectionHeading Is a heading that is used to identify a particular section.

Class(wafa:`SectionHeading` partial wafa:`Heading`
restriction(wafa:*is-heading-for* allValuesFrom wafa:`Section`))

Concept(Children of **`Atom`**)	**Page No.**
`Advertisement`	Page 165
`AdvertisementBanner`	Page 165
`Caption`	Page 166
`FigureCaption`	Page 166
`TableCaption`	Page 166
`Heading`	Page 166
`Headline`	Page 166
`SectionHeading`	Page 166
`Label`	Page 166
`Link`	Page 168
`Note`	Page 168
`Citation`	Page 168
`Footnote`	Page 168
`Copyright`	Page 168
`NB`	Page 168
`PS`	Page 168
`Separator`	Page 168
`Boundary`	Page 169
`Banner`	Page 169
`AdvertisementBanner`	Page 165
`TitleBanner`	Page 169
`Space`	Page 169
`SymbolSeparator`	Page 169
`SpecialGraphic`	Page 169
`Banner`	Page 169
`Icon`	Page 169
`Logo`	Page 169
`Title`	Page 169
`TitleBanner`	Page 169

Table B.2: The overview of the concept “Atom”.

Label Identifying marker attached to an object; usually used with a form element.

Link Is used to provide a link to objects (`Atom`,`Chunk` or `Node`) and has a URI[2] definition normally used in conjunction with HTML anchor (A) element. This concept subsumes the following concepts:

- **ReferentialLink** Provides a link between an item of information and an elaboration or explanation of that information [72].
- **AssociativeLink** Is used to maintain semantic relationship between the linked elements. A good example is a link that is used for cross-referencing [72].
- **StructuralLink** Is used to maintain the structure of the document, but it does not provide any semantic relationship between the linked information [72].
- **SkipLink** A link that enables users, in particular visually impaired users, to avoid certain areas that are considered as obstacles or not of interest. For example, a skip link is usually provided to skip a header or sidebar [109].
- **ToTextOnlyPage** A Link that can be used to move to the text only version of a page.

Note A comment that is usually added as supporting information to the main content of the document- additional information.

Citation A reference or quotation; a short note recognised a source of information or of a quoted passage.

Footnote In printed documents, it is a note placed below the main body of the text on a page or can be defined as a note to explain, comment on or provide references for an information in a document.

Copyright Is copyright note that is usually located at the bottom of a Web page.

```
Class(wafa:Copyright partial wafa:Footnote
  restriction(wafa:has-part someValuesFrom wafa:CopyrightSymbol))
```

NB Nota bene. Usually used to direct attention to something particularly important.

PS Postscript. A note appended to a letter after a signature.

[2]See `http://www.w3.org/Addressing/`

Separator Is usually used to separate one object from the others. It can also be considered as a delimiter.

Boundary Indicates the limit or extend of other objects. It usually determines the limits of an area and indicates a border or limit.

Banner Has a variety of rectangular or square shapes and is meant to catch the immediate attention of the visitor. It can include text, graphic and animation. This concept has two children: `AdvertisementBanner` and `TitleBanner`.

Space This concept represents a white space that is usually used to separate items of a list or paragraphs.

SymbolSeparator An arbitrary sign that has acquired a conventional significance. On Web pages, a set of special characters are used for specific purposes. For example, 'vertical bar' is usually used to separate a link from others in a link menu. This concept subsumes the following concepts:

CommaSeparator Represents the punctuation mark ',' that is used to separate elements.

CopyrightSymbolSeparator Represents the symbol: 'c' inside a circle.

DashSeparator Represents the punctuation mark '-'.

TriangleLeftSeparator Represents the triangle symbol pointing left.

TriangleRightSeparator Represents the triangle symbol pointing right.

VerticalBarSeparator Represents the "vertical bar" character.

SpecialGraphic Images with a special purpose conveying additional meaning.

Icon A graphic symbol (usually a simple picture) that denotes a concept in a graphical system.

Logo An emblem or device used to orientate users to their environment.

Title An identifying name given to a `Node` (e.g., a Web page).

TitleBanner Is usually at the top and presents the title of the page. It usually runs across the full page and provides a kind of boundary at the top. This concept is also a child of the `Banner` concept.

B.1.2 The Concept 'Chunk'

Several objects grouped together to form a coherent unit. Table B.3 shows a hierarchical view of the children of this concept which are explained as follows:

Figure A diagram or picture illustrating textual material.

```
Class(wafa:Figure partial wafa:Chunk
  restriction(wafa:has-caption allValuesFrom wafa:FigureCaption))
```

Footer Is typically positioned at the bottom of a page and includes information about the designer or the company, copyright information and a list of links.

```
Class(wafa:Footer partial Chunk
  IntersectionOf(
    restriction(wafa:is-positioned-at allValuesFrom wafa:Bottom)
    restriction(wafa:has-direct-part someValuesFrom
      unionOf(wafa:Logo wafa:Link wafa:LinkMenu wafa:Copyright))))
```

RunningFooter A footer that is positioned at the bottom of every page in a site or on a group of pages in a site.

Header Is typically printed at the top of a page and includes, for example, a company logo, the title of the page, list of links and sometimes a search engine.

```
Class(wafa:Header partial Chunk
  IntersectionOf(
    restriction(wafa:is-positioned-at allValuesFrom wafa:Top)
    restriction(wafa:has-part someValuesFrom
      unionOf(wafa:Logo wafa:Banner wafa:SearchEngine
        wafa:LinkMenu wafa:SkipLink))))
```

RunningHeader A header that is positioned at the top of every page of a site or on a group of pages in a site.

List Contains an array of items.

BreadcrumbTrail Representation of where and how information is located within a Web site. It allows a user to link to major categories of information along a continuum of sequential order.

Concept(Children of **Chunk**)	**Page No.**
Figure	Page 170
Footer	Page 170
RunningFooter	Page 170
Header	Page 170
RunningHeader	Page 170
List	Page 170
BreadcrumbTrail	Page 170
NavigationalBreadcrumbTrail	Page 170
AttributeBreadcrumb	Page 173
LocationBreadcrumb	Page 173
pathBreadcrumb	Page 173
DefinitionList	Page 173
NavigationalList	Page 173
OrderedList	Page 175
UnorderedList	Page 175
ReferentialChunk	Page 175
SearchEngine	Page 175
Section	Page 175
Abstract	Page 175
Sidebar	Page 175
Summary	Page 175
Abstract	Page 175
PageSummary	Page 175
SiteSummary	Page 176
Table	Page 176
DataTable	Page 176
LayoutTable	Page 176

Table B.3: The overview of the concept “Chunk”.

Concept(Children of **`NavigationalList`**)	**Page No.**
`Bookmark`	Page 173
`  Favourites`	Page 173
`  ShoppingCart`	Page 173
`Drectory`	Page 173
`  SiteMap`	Page 174
`  WebDirectory`	Page 174
`FAQ`	Page 174
`HistoryList`	Page 174
`  pathBreadcrumb`	Page 173
`Index`	Page 174
`  SiteIndex`	Page 174
`LinkMenu`	Page 174
`  DropDownLinkMenu`	Page 174
`NavigationalBreadcrumbTrail`	Page 170
`  AttributeBreadcrumb`	Page 173
`  LocationBreadcrumb`	Page 173
`  pathBreadcrumb`	Page 173
`TableOfContent`	Page 174

Table B.4: The overview of the concept “NavigationalList” (see Table B.3).

Concept(Children of **`OrderedList`**)	**Page No.**
`AttributeBreadcrumb`	Page 173
`Directory`	Page 173
`  SiteMap`	Page 174
`  WebDirectory`	Page 174
`HistoryList`	Page 174
`  pathBreadcrumb`	Page 173
`Index`	Page 174
`  SiteIndex`	Page 174
`LocationBreadcrumb`	Page 173
`TableOfContent`	Page 174

Table B.5: The overview of the concept “OrderedList” (see Table B.3).

NavigationalBreadcrumTrail Provides information about the location of a page with respect to the overall structure of a site. The main difference between this and the `BreadcrumbTrail` concept is that the latter does not include links.

LocationBreadcrumb Conveys the position of the current page within the site hierarchy. A page has the same breadcrumb trail, no matter how users get there.

```
Class(wafa:LocationBreadcrumb partial
  restriction(wafa:has-order someValuesFrom wafa:HierarchicalOrder))
```

AttributeBreadcrumb Some sites use breadcrumbs like navigational features as meta-information to describe components of the site. Links are provided in a breadcrumb manner instead of a flat list of keywords.

```
Class(wafa:AttributeBreadcrumb partial
  restriction(wafa:has-order someValuesFrom wafa:HierarchicalOrder))
```

PathBreadcrumb Shows the path the user has taken within the site to get to the current page. A page can have different breadcrumbs based on how users get there.

```
Class(wafa:PathBreadcrumb partial
  restriction(wafa:has-order someValuesFrom wafa:TemporalOrder))
```

DefinitionList Is used to present the definitions of a list of items (e.g., HTML DL element)

NavigationalList A list of links (items have to provide navigation support). Links do not need to be presented in a specific order.

```
Class(wafa:NavigationalList partial wafa:List
  restriction(wafa:has-Part someValuesFrom Link))
```

Bookmark Used to mark a page or a specific place in a page for later retrieval.

Favourites List of links bookmarked on personal pages for later retrieval. Typically users bookmark pages for their future journeys and as a reference for others.

ShoppingCart Is synonymous to a shopping basket. It is usually used to bookmark a set of pages or items on a shopping site so that they can be used in the payment process.

Directory supports organisation of links in a hierarchical structure. Since it has a specific order, it is also a child of `OrderedList`.

```
Class(wafa:Directory partial
  unionOf (wafa:NavigationalList wafa:OrderedList)
  restriction(wafa:has-order someValuesFrom wafa:HierarchicalOrder))
```

SiteMap Provides a well structured view of pages in a site. It usually shows the categories of links, ordered in a specific sequence used to show the structure and overview of the site.

WebDirectory Provides links to different web sites in a well structured way (e.g., Google directory page).

FAQ Is a list of questions that are frequently asked about a specific topic along with their answers.

HistoryList Provides links to the pages that the user has visited before. It usually presents the links in a particular time order therefore it is also a child of the `OrderedList`.

```
Class(wafa:HistoryList partial
  unionOf (wafa:NavigationalList wafa:OrderedList)
  restriction(wafa:has-order someValuesFrom wafa:TemporalOrder))
```

Index Provides links to pages in an alphabetical order based on keys (or keywords), each of which identifies a page. This concept has only one child which is `SiteIndex`.

```
Class(wafa:Index partial
  unionOf (wafa:NavigationalList wafa:OrderedList)
  restriction(wafa:has-order someValuesFrom wafa:AlphabeticalOrder))
```

SiteIndex provides an alphabetical list of links to pages within a site.

LinkMenu A list of links that are meant to represent a menu. This concept has one child: `DropDownLinkMenu` which is meant to represent a menu where options appear below the item when the user clicks on them.

TableOfContent Outlines the content of a site or a page in a structured manner.

```
Class(wafa:HistoryList partial
  unionOf (wafa:NavigationalList wafa:OrderedList)
  restriction(wafa:has-order someValuesFrom
    unionOf (wafa:HierarchicalOrder wafa:SpatialOrder)))
```

OrderedList Is a list of items in a specific order (e.g., HTML OL tag). This concept subsumes a number of concepts which are: `Directory`, `HistoryList`, `Index`, `AttributeBreadcrumb`, `LocationBreadcrumb` and `TableOfContent` (see Table B.5).

```
Class(wafa:OrderedList partial  wafa:List
  restriction(wafa:has-order someValuesFrom wafa:Order))
```

UnorderedList Is a collection of related items that have no special sequence or order (e.g., HTML UL tag).

ReferentialChunk Is typically used to provide a summary of a page and a link to that page. It provides referential information [72]. It typically contains a heading, a small paragraph of text and a small image that is used to illustrate the content. It is used to introduce the target page and provide link so that the user can follow in order to get further information.

```
Class(wafa:ReferentialChunk partial  wafa:Chunk
  intersectionOf(
    restriction(wafa:has-part someValuesFrom wafa:ReferentialLink)
    restriction(wafa:support-connection
      someValuesFrom wafa:ReferentialConnection)))
```

SearchEngine Consists of an edit box, a button and is usually identified by a `Label`. It could be used to search the site or the Web.

Section A self-contained part of a page.

Abstract A sketchy summary of the main points of an argument or theory.

Sidebar Is similar to frames and typically used to provide navigational features. It can also be referred to as a navigation bar and usually positioned either on the left or right side of the page.

Summary A brief statement that represents the main points in a concise form.

PageSummary A chunk that provides a summary of the page.

SiteSummary A chunk that provides a summary of a site.

Table Is a tabular array. A set of data arranged in rows and columns.

DataTable A table used to organise data rather than layout of a page.

LayoutTable A table used to organise layout of a page.

Toolbar A list of actions (or verbs) represented by icons.

B.1.3 Concepts to describe Authoring Semantics

This section presents concepts that are mainly introduced in the ontology to describe `AuthoringConcept`s. These concepts are: `Connection`, `Coverage`, `Position`, `Order` and `Structure`. Table B.6 provides a hierarchical view of these concepts which are explained as follows:

Connection The context of the connections (links) between objects (`Collection`, `Node`, `Chunk` or `Atom`) [72]:

- **AssociativeConnection** Connecting (linking) objects based on the similarity of their context. Instantiation of a semantic relationship between information elements or can be defined as creating links based on meaning and cross referencing of objects.
- **ReferentialConnection** Connecting (linking) objects based on an elaboration of the information or can be defined as providing a link between an item of information and an elaboration or explanation of that information. The item at one end of a referential link exists because of the existence of the other item (e.g., a link from a word to its definition).
- **StructuralConnection** Connecting (linking) objects based on the maintenance of the structure of the docuverse or can be defined as providing a form for the information space which allows users to develop an understanding of its scale and their location within the space. This kind of connection does not imply any semantic relationships between linked information.

Coverage The range over which mobility concepts are applied or to extend to which something is covered.

Concept(Children of `Thing`)	**Page No.**
`Connection`	Page 176
`AssociativeConnection`	Page 176
`ReferentialConnection`	Page 176
`StructuralConnection`	Page 176
`Coverage`	Page 176
`CollectionWide`	Page 176
`Inter`	Page 177
`Intra`	Page 177
`Position`	Page 177
`Bottom`	Page 177
`Side`	Page 177
`Top`	Page 177
`Order`	Page 178
`AlphabeticalOrder`	Page 178
`HierarchicalOrder`	Page 178
`NumericalOrder`	Page 178
`NumericalOrder`	Page 178
`SpatialOrder`	Page 178
`TemporalOrder`	Page 178
`Structure`	Page 178
`Hierarchy`	Page 178
`Linear`	Page 178
`Matrix`	Page 178
`Network`	Page 178

Table B.6: The overview of the concepts used to describe "AuthoringConcepts" (see Table B.1).

CollectionWide Covers mobility WITHIN a `Collection` (e.g., site).

Inter Covers mobility BETWEEN `Node`s (e.g., pages).

Intra Covers mobility WITHIN a `Node` (e.g., page).

Position These concepts can be used to show the position of the objects within a page. The aim of this concept is not to provide the actual physical position but to make the implicit logical position information explicit.

Bottom Bottom of a `Node` (e.g., page).

Top Top of a `Node` (e.g., page)

Side The object is positioned either at the right or left of the page, but that is not really important, the important thing is the notion of being on the side.

Order These concepts can be used to indicate the order of organisation of the components of an object.

AlphabeticalOrder Components are arranged in an alphabetical order.

HierarchicalOrder Components are arranged in a hierarchical order.

NumericalOrder Components are arranged in a numerical order.

SpatialOrder The physical ordering of the components is meant to represent the logical ordering of the components.

TemporalOrder Components are ordered in a time based manner, the sequence of the items happens in a specific order in a time span.

Structure These concepts are concerned with maintaining the structure of the docuverse either as a `Collection`, `Node`, `Chunk` or `Atom` of information. For example, the structures defined below (e.g., linear) can be used either to represent the complete structure of a site (`Node` structure in a `Collection`) or they could be an internal structure of a page (`Chunk` or `Atom` structure in a page).

Hierarchy Each chunk of information (`Collection`, `Node`, `Chunk` or `Atom`) is connected such that superior nodes are connected to interior nodes (provided connections tend to show parent-child relationship).

Linear Each chunk of information (`Collection`, `Node`, `Chunk` or `Atom`) is connected in a linear structure such that it can be followed in a stepwise manner (like a guided tour).

Matrix Each chunk of information (`Collection`, `Node`, `Chunk` or `Atom`) is connected to all its neighbours in a matrix structure.

Network Each chunk of information (`Collection`, `Node`, `Chunk` or `Atom`) is connected to other chunks based on their associative context (like graph) and does not maintain a rigid structure.

Property	**Page No.**
has-footnote	Page 179
is-footnote-for	Page 179
has-order	Page 179
has-part	Page 180
has-direct-part	
has-title	Page 180
has-caption	Page 180
has-label	Page 180
has-heading	Page 180
has-structure	Page 180
has-summary	Page 180
is-summary-of	Page 180
is-part-of	Page 181
is-direct-part-of	
is-caption-of	Page 181
is-label-for	Page 181
is-heading-for	Page 181
is-positioned-at	Page 181
supports-movement	Page 181

Table B.7: The overview of the properties used to describe “AuthoringConcepts”.

B.1.4 Properties of Authoring Semantics

This section explains the properties or structural abstractions defined in the ontology. These properties are particularly introduced in the ontology to define Authoring Semantics and can be explained as follows:

has-footnote To specify that an object has a particular footnote associated with it.

```
ObjectProperty(wafa:has-foootnote
  inverseOf(wafa:is-footnote-for)
  range(wafa:Footnote))
```

is-footnote-for To indicate that a footnote is associated with a particular object.

```
ObjectProperty(wafa:is-foootnote-for
  inverseOf(wafa:has-footnote)
  domain(wafa:Footnote))
```

has-order To indicate the order that objects are presented.

```
ObjectProperty(wafa:has-order
  range(wafa:Order))
```

has-part To specify parts of an object.

```
ObjectProperty(wafa:has-part
  inverseOf(wafa:is-part-of))
```

has-direct-part Introduced to be able to create restrictions about the direct parts.

has-title To specify the title of an object.

```
ObjectProperty(wafa:has-title
  range(wafa:Title))
```

has-caption To indicate the caption of an object.

```
ObjectProperty(wafa:has-caption
  range(wafa:Caption))
```

has-label To specify a label for an object.

```
ObjectProperty(wafa:has-label
  range(wafa:Label))
```

has-heading To indicate the heading of an object.

```
ObjectProperty(wafa:has-heading
  range(wafa:Heading))
```

has-structure To specify the underlying structure of an object.

```
ObjectProperty(wafa:has-structure
  range(wafa:Structure))
```

has-summary To specify the summary of an object.

```
ObjectProperty(wafa:has-summary
  inverseOf(wafa:is-summary-of))
  range(wafa:Summary))
```

is-summary-of Inverse of *has-summary* property. It can be used to indicate which object does the summary object summarise.

```
ObjectProperty(wafa:is-summary-of
  inverseOf(wafa:has-summary))
  domain(wafa:Summary))
```

is-part-of Inverse of *has-part* property.

```
ObjectProperty(wafa:is-part-of
  inverseOf(wafa:has-part))
```

has-direct-part Introduced to be able to create restrictions about the direct parts.

is-caption-of Inverse of *has-caption* property.

```
ObjectProperty(wafa:is-caption-of
  inverseOf(wafa:has-Caption))
  domain(wafa:Caption))
```

is-label-for Inverse of *has-label* property.

```
ObjectProperty(wafa:is-label-for
  inverseOf(wafa:has-Label))
  domain(wafa:Label))
```

is-heading-for Inverse of *has-heading* property.

```
ObjectProperty(wafa:is-heading-for
  inverseOf(wafa:has-heading))
  domain(wafa:Heading))
```

is-positioned-at To indicate the logical position of an object.

```
ObjectProperty(wafa:is-positioned-at
  range(wafa:Position))
```

supports-movement To indicate the mobility coverage of an object.

```
ObjectProperty(wafa:supports-movement
  range(wafa:Coverage))
```

B.1.5 Meta-Knowledge about Authoring Semantics

The following list shows some meta-knowledge included in the ontology about concepts in the Authoring Semantics part. For example, (4) indicates that an object cannot

Concept(Children of `Purpose`)	**Page No.**
`ObjectPurpose`	Page 182
`AidsExternalMemory`	Page 182
`AidsNavigation`	Page 183
`AidsOrientation`	Page 183
`TravellerPurpose`	Page 183
`Browsing`	Page 183
`CasualBrowsing`	Page 183
`OpportunisticBrowsing`	Page 183
`systematicBrowsing`	Page 183
`Monitoring`	Page 183
`Observing`	Page 183
`Querying`	Page 183
`Scanning`	Page 183
`Searching`	Page 183

Table B.8: The overview of the concepts used to describe the concept “Purpose”.

be a `Node` and an `Atom` at the same time. Similarly, (8) indicates that an object cannot be a `Chunk` and `Atom` at the same time.

(1) DisjointClasses(wafa:`Top` wafa:`Bottom`)
(2) DisjointClasses(wafa:`Collection` wafa:`Node`)
(3) DisjointClasses(wafa:`Chunk` wafa:`Collection`)
(4) DisjointClasses(wafa:`Node` wafa:`Atom`)
(5) DisjointClasses(wafa:`Footnote` wafa:`Heading`)
(6) DisjointClasses(wafa:`Header` wafa:`Footer`)
(7) DisjointClasses(wafa:`Collection` wafa:`Atom`)
(8) DisjointClasses(wafa:`Chunk` wafa:`Atom`)
(9) DisjointClasses(wafa:`Chunk` wafa:`Node`)

B.2 Contextual Semantics

This section describes the concepts in the contextual semantics. All the concepts defined here are subsumed by the concept `Purpose`. Table B.8 shows the hierarchical view of the children of this concept that are explained as follows:

ObjectPurpose Shows the intention of an object in an environment rather than the purpose of a traveller (reader).

AidsExternalMemory Knowledge is the ability to recognise environmental and spatial information from sources of learnt knowledge either from external sources like maps, or directly from mental recall. External memory is the provision of these external sources of learnt knowledge (e.g., maps, guides) and so the purpose of the object is to support such external memory.

AidsNavigation The purpose of the object is to help user answer the question "where can I go?"

AidsOrientation The purpose of the object is to help user answer the question "where am I?".

TravellerPurpose Shows the purpose or intention of the traveller.

Browsing Is an activity that a traveller can engage in while travelling a journey or can be defined as movement through an information space. It is the task of looking to see what is available in the world [66].

CasualBrowsing The user has no specific goal and is not focused. The activity is purely random, unstructured and unsystematic [31].

OpportunisticBrowsing The user has a less definite target. The activity is less systematic and more predictive [31].

SystematicBrowsing The user has a specific goal or target. The activity is focused, structured and systematic [31].

Monitoring Focuses on attributes of interest to the information seeker [74]. For example, while reading a text related to a specific topic, a monitoring strategy "listens" for concepts related to another topic of interest.

Observing Used when the information seeker assumes that he(she) is in a promising neighbourhood and react the stimuli from the neighbourhood. For example, advertisements in newspapers attract the browser's attention as he(she) applies observational tactics. Observations may lead to interesting discoveries but yield initiation control to the environment [74].

Querying Defined as using a search engine to submit a description of the object being sought and receiving relevant content of information [66].

Scanning An activity that compares sets of well-defined objects with an object that is clearly represented in the information seeker's mind [74].

Searching This activity is the task of looking for a known target [66].

Appendix C

The User Evaluation

C.1 The Evaluation Materials

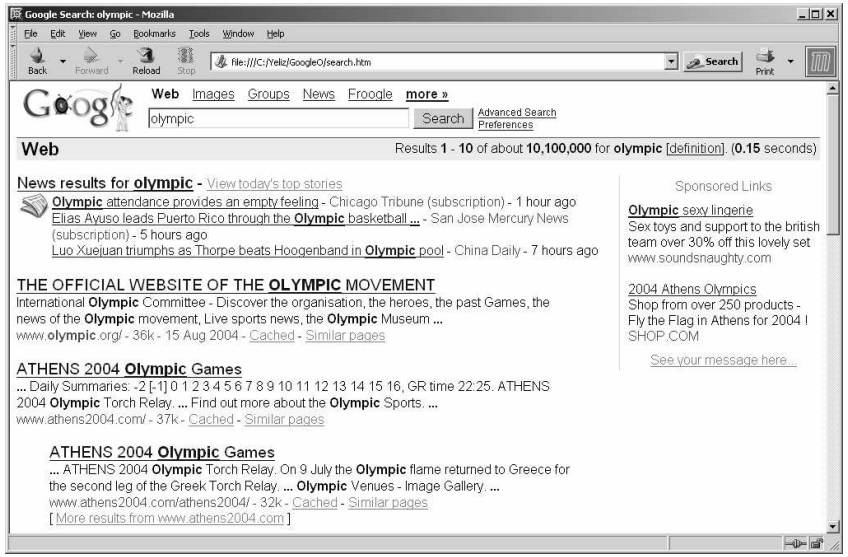


Figure C.1: A result page of Google (17/08/2004).

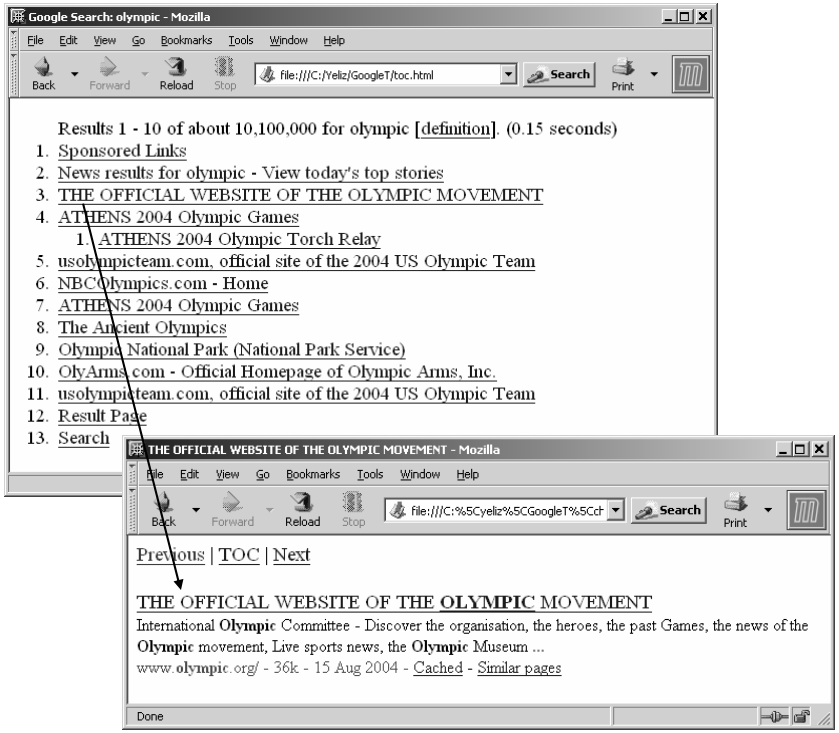


Figure C.2: A physical fragmentation of C.1.

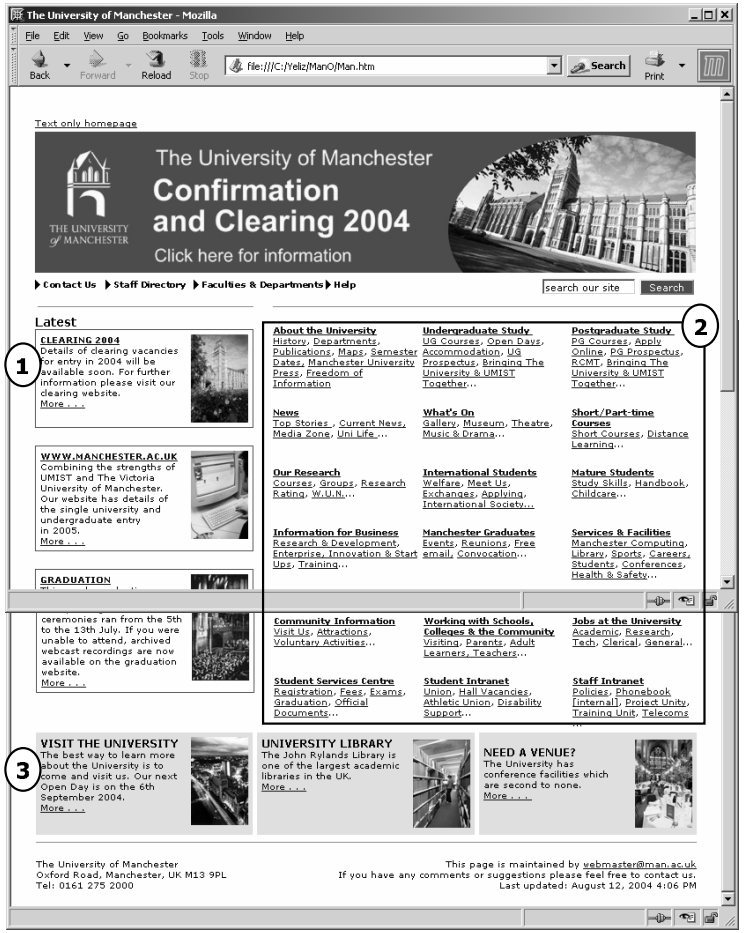


Figure C.3: The home page of Manchester University (17/08/2004).

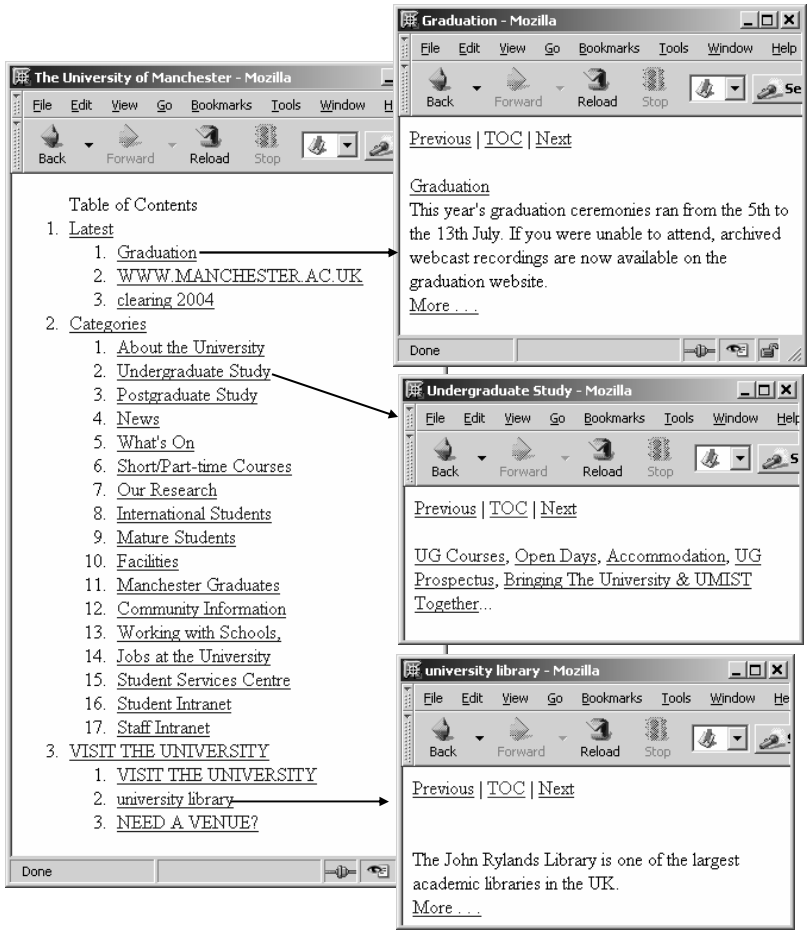


Figure C.4: A physical fragmentation of C.3.

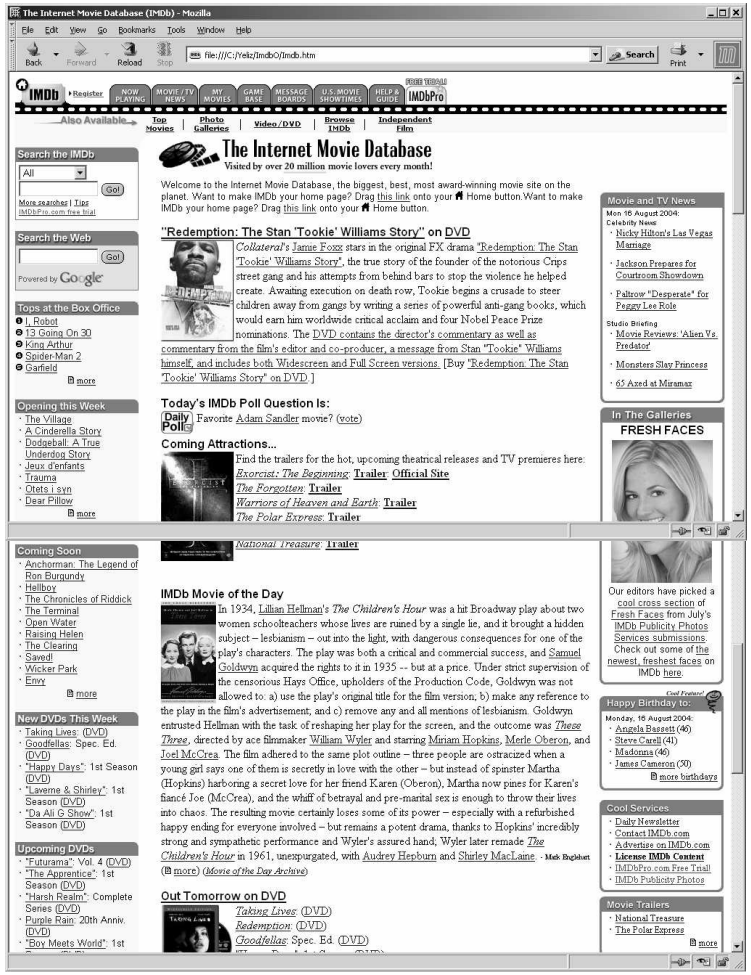


Figure C.5: The home page of IMDB (17/08/2004).

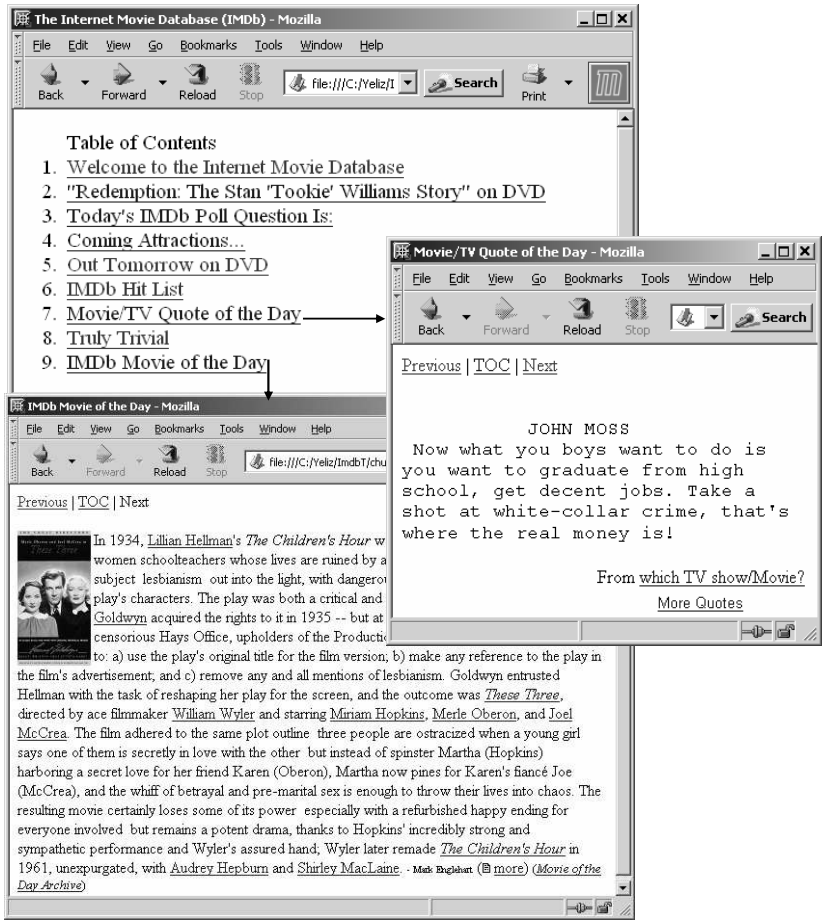


Figure C.6: A physical fragmentation of C.5.

C.2 The Evaluation Methods

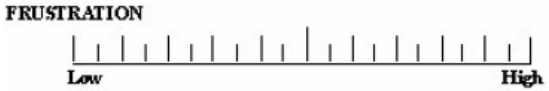
Title	Endpoints	Descriptions	Metrics	Nasa TLX Metrics
Mental Demand	Low/High	How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?	0- Low 20-High	
Physical Demand	Low/High	How much physical activity was required (e.g., pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?	0- Low 20-High	
Temporal Demand	Low/High	How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?	0- Low 20-High	
Effort	Low/High	How hard did you have to work (mentally and physically) to accomplish your level of performance?	0- Low 20-High	
Performance	Good/Poor	How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goa	0- Good 20- Poor	
Frustration Level	Low/High	How insecure, discouraged, irritated, stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?	0- Low 20-High	

Table C.1: NASA Task Load Index (TLX) details.

C.3 The Evaluation Results

1	Name and surname	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
2	Gender	M	F	M	F	M	F	M	M	M	M
3	Occupation	Learning advisor	x	Part-time	x	Physotherapist	Student	Self-employed	Retired	Lecturer	x
4	Age range (<16, 16-30, 31-45, 46-60, >60):	16-30	31-45	16-30	31-45	>60	16-30	31-45	>60	46-60	46-60
5	You must be familiar with the Web (WWW) (Yes/No):	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
4	How do you describe your vision (blind, sighted, partially sighted):	Blind	Blind	Blind	Blind	Blind	Partially sighted	Blind	Blind	Blind	Partially sighted
5	Which browser and operating system do you use?	IE& W-XP	IE& W-XP	IE& W-2000	IE& W-XP	IE& W-XP	IE& W-98	IE& W-XP	IE& W-2000	IE& W-XP	IE& W-XP
6	Which assistive technology do you use?	Jaws5.0	Jaws5.0	Jaws4.0	Hal	Jaws3.7	Lunar	Jaws3.7	Hal	Super Nova	Hal
8	Does participant wish to remain anonymous?	No	No	No	No	No	Yes	No	No	No	No
9	Does participant give permission to use material from this evaluation within reports?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

x– Unemployed
W-XP- Windows XP
IE- Internet Explorer

Table C.2: Profiles of the participants.

C.4 TLX Scores

TLX:	Mental Demand		Physical Demand		Temporal Demand		Effort		Performance		Frustration Level	
O/T	O	T	O	T	O	T	O	T	O	T	O	T
P1	13	8	10	4	7	6	10	3	8	2	13	5
P2	15	12	17	15	15	12	20	12	5	3	20	5
P3	14	3	14	4	10	6	13	2	6	2	8	0
P4	15	7	2	4	0	0	12	4	4	2	15	10
P5	17	8	12	4	0	0	6	3	16	10	8	2
P6	12	5	15	2	15	4	15	3	12	3	16	3
P7	10	2	15	5	18	5	15	5	10	10	20	0
P8	15	0	17	2	18	3	18	3	18	5	20	2
Mean:	13.88	5.63	12.75	5.00	10.38	4.50	13.63	4.38	9.88	4.63	15.00	3.38
StDev:	2.17	3.89	4.95	4.17	7.42	3.85	4.44	3.20	5.14	3.46	5.04	3.29

Note: For all factors, except perceived performance level, a low score is positive in terms of usability ('O'= Original page and 'T'= Transformed page).

Table C.3: Individual and mean rating for TLX factors for Figure C.3 and C.4.

TLX:	Mental Demand		Physical Demand		Temporal Demand		Effort		Performance		Frustration level	
O/T	O	T	O	T	O	T	O	T	O	T	O	T
P1	16	4	15	2	16	4	17	4	14	2	17	0
P4	18	9	15	4	0	0	16	4	18	4	20	0
P5	8	3	3	3	3	3	3	3	16	7	14	7
P6	14	5	10	4	12	3	10	3	9	2	7	2
P7	18	1	18	0	18	0	20	0	20	0	20	0
P8	20	5	20	4	20	5	20	5	20	6	20	3
Mean:	15.67	4.50	13.50	2.83	11.50	2.50	14.33	3.17	16.17	3.50	16.33	2.00
StDev:	4.27	2.66	6.16	1.60	8.24	2.07	6.65	1.72	4.22	2.66	5.16	2.76

Note: For all factors, except perceived performance level, a low score is positive in terms of usability ('O'= Original page and 'T'= Transformed page).

Table C.4: Individual and mean rating for TLX factors for Figure C.5 and C.6.

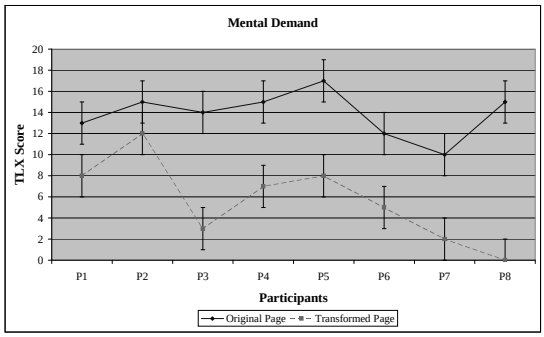


Figure C.7: Comparing TLX *Mental Demand* for Figure C.3 and C.4. [O_{Mean}=13.88 T_{Mean}=5.63].

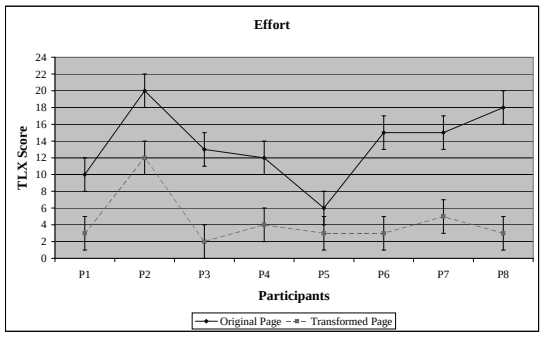


Figure C.10: Comparing TLX *Effort* for Figure C.3 and C.4. [O_{Mean}=13.63 T_{Mean}=4.38]

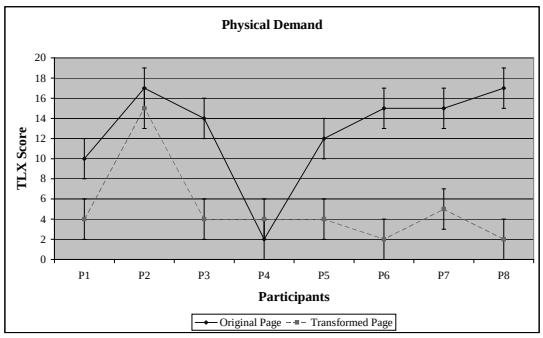


Figure C.8: Comparing TLX *Physical Demand* for Figure C.3 and C.4. [O_{Mean}=12.75 T_{Mean}=5.00]

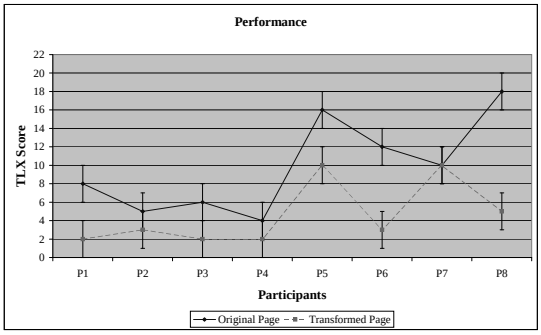


Figure C.11: Comparing TLX *Performance* for Figure C.3 and C.4. [O_{Mean}=9.88 T_{Mean}=4.63]

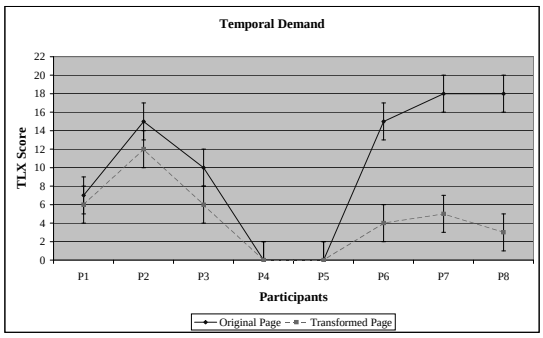


Figure C.9: Comparing TLX *Temporal Demand* for Figure C.3 and C.4. [O_{Mean}=10.38 T_{Mean}=4.50].

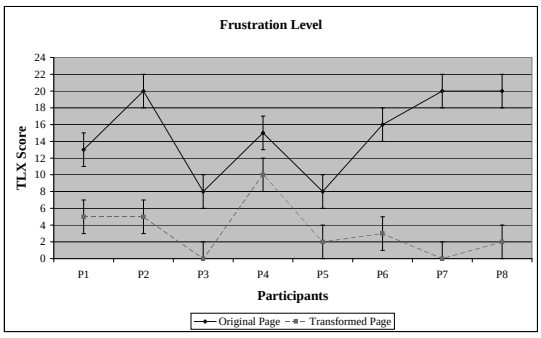


Figure C.12: Comparing TLX *Frustration Level* for Figure C.3 and C.4. [O_{Mean}=15.00 T_{Mean}=3.38]

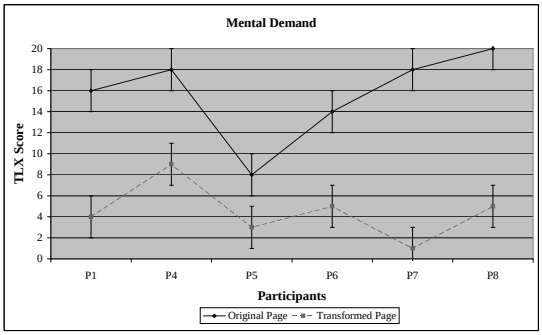


Figure C.13: Comparing TLX *Mental Demand* for Figure C.5 and C.6. [O_{Mean}=15.67 T_{Mean}=4.50].

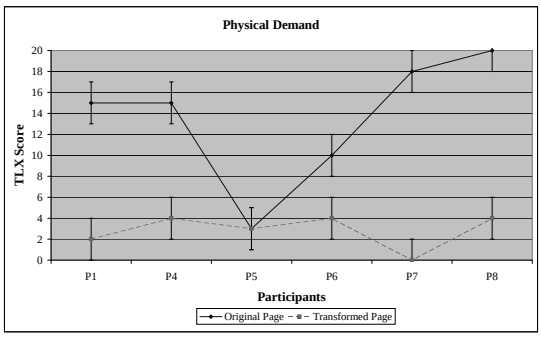


Figure C.14: Comparing TLX *Physical Demand* for Figure C.5 and C.6. [O_{Mean}=13.50 T_{Mean}=2.83]

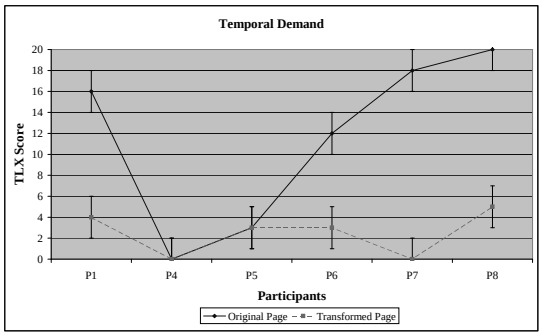


Figure C.15: Comparing TLX *Temporal Demand* for Figure C.5 and C.6. [O_{Mean}=11.50 T_{Mean}=2.50].

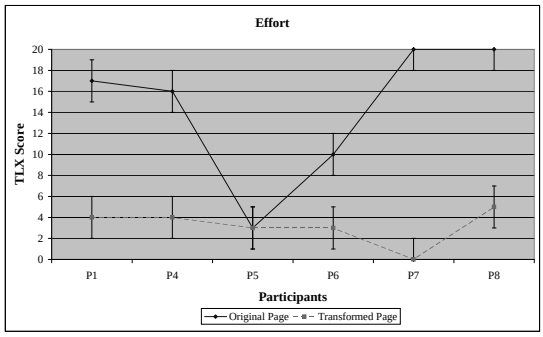


Figure C.16: Comparing TLX *Effort* for Figure C.5 and C.6. [O_{Mean}=14.33 T_{Mean}=3.17]

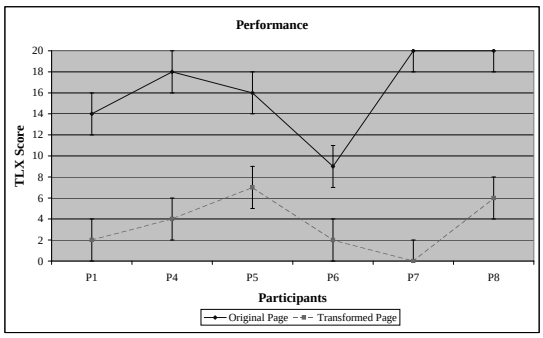


Figure C.17: Comparing TLX *Performance* for Figure C.5 and C.6. [O_{Mean}=16.17 T_{Mean}=3.50]

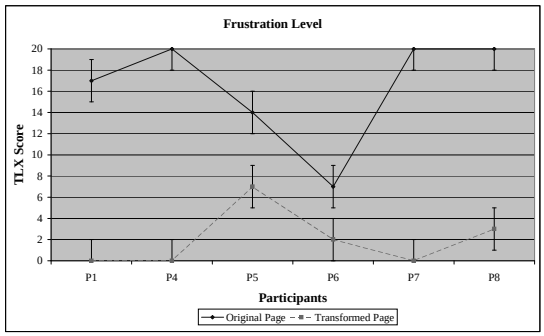


Figure C.18: Comparing TLX *Frustration Level* for Figure C.5 and C.6. [O_{Mean}=16.33 T_{Mean}=2.00]

Questions	P1	P2	P3	P4	P5	P6	P7	P8
1. Do you know (do you have an idea) how long this page is (we are not looking for the exact size but trying to find out if you have an idea about the size?)	Yes, Quite long.	Quite long.	Long and lots of information.	Long, lots of links.	Quite long.	Similar size.	Quite long page.	No idea, don't know.
2. Do you think now you have an overview of the page?	Very limited.	Not really.	Yes.	Not really.	No.	Not really	No.	Not really.
3. How difficult was it to get such an overview?	Difficult.	Difficult.	Difficult, lots of effort.	Very difficult.	Difficult.	Very difficult.	Difficult.	Very hard work.
4. What do you think about the amount of information?	Too much.	Too much.	Quite a lot.	lot of links.	Too much and too much details to deal with.	Quite a lot .	Too much.	A lot of information.
5. Did you find this page complex? If yes How complex you think is this page?	Complex.	Yes.	Neither easy nor difficult.	Complex.	Complex.	Very complex.	Yes.	Yes.
6. Was there anything particular you found repetitive in the page?	No.	No.	No.	No.	So many links and cannot differentiate.	No.	Lots of repeated links.	Very long list of links.
7. How easy was it to access different parts of this page?	Quite long.	Very complex.	No.	Difficult to picture.	Very difficult.	Complicated.	Yes.	Difficult.
8. How easy was it for you to figure out your position in a specific section with respect to other sections in the page?	Not difficult.	Very difficult.	Difficult.	Difficult.	Difficult.	Difficult.	Not that difficult.	Very difficult.
9. At any point in your journey, did you think you were lost (didn't know where you are)?	No.	Yes.	Not straightforward.	No.	Yes,moved to top of the page.	No.	Yes, moved to top of the page.	No.

Table C.5: Summary of the structured interview results for the original version of home page of Manchester University (see Figure C.3).

Questions	P1	P2	P3	P4	P5	P6	P7	P8
1. Do you know (do you have an idea) how long this page is (we are not looking for the exact size but trying to find out if you have an idea about the size?)	Yes, different.	Quite long but manageable.	Yes, well structured.	Not long.	Not the actual size.	Similar size.	Not long, not the same page and size.	Not the physical size but the logical.
2. Do you think now you have an overview of the page?	Yes, can scan the page.	Yes	Yes, well structured.	Yes, not repetitions.	Yes.	Not really.	Easier to get an overview.	Much simpler content.
3. How difficult was it to get such an overview?	Very easy.	Easy.	Easy.	Easy.	Easy.	Easy.	Easy.	Easy.
4. What do you think about the amount of information?	Not too much, just right amount.	More structured.	Not much.	Less clutter to go through.	Not much	Not much.	Much better.	Not as much as possible but useful.
5. Did you find this page complex? If yes How complex you think is this page?	Wasn't complex.	Not really.	Very simple.	Simple.	Not complex.	Simple.	No.	No.
6. Was there anything particular you found repetitive in the page?	No.	No.	No.	No.	No.	No.	No.	No.
7. How easy was it to access different parts of this page?	Easy.	No.	Very easy.	Easy.	Easy.	Quite easy.	No.	Easy.
8. How easy was it for you to figure out your position in a specific section with respect to other sections in the page?	Very easy.	Easy.	Easy.	Easy.	Not that difficult.	Easy.	Easy.	Easy.
9. At any point in your journey, did you think you were lost (didn't know where you are)?	No.	No.	Not at all.	No.	No.	No.	No.	No.

Table C.6: Summary of the structured interview results for the transformed version of the home page of Manchester University (see Figure C.4).

Questions	P1	P4	P5	P6	P7	P8
1. Do you know (do you have an idea) how long this page is (we are not looking for the exact size but trying to find out if you have an idea about the size?)	No.	Really long.	No.	No.	Very long.	No, but guess that is long.
2. Do you think now you have an overview of the page?	Yes.	No.	No.	Not really.	Rough idea.	No.
3. How difficult was it to get such an overview?	Difficult.	No.	Very difficult.	Yes.	Not great.	Difficult.
4. What do you think about the amount of information?	Too much.	Not useful.	Too much.	Hard to read.	Too much.	Too much.
5. Did you find this page complex? If yes How complex you think is this page?	Yes, too complex.	Yes.	Very complex.	Very very complex.	Yes.	Difficult.
6. Was there anything particular you found repetitive in the page?	Yes, lots of tables and links at the top.	Yes.	Not complex.	No.	Lots of tables without headings.	No.
7. How easy was it to access different parts of this page?	Very difficult	Difficult.	Very difficult.	Links were not clear.	Difficult.	Yes.
8. How easy was it for you to figure out your position in a specific section with respect to other sections in the page?	Relatively yes, too long.	Very difficult.	No, very difficult.	Difficult.	Difficult.	Difficult.
9. At any point in your journey, did you think you were lost (didn't know where you are)?	Yes.	Yes, at some point didn't have any clue.	Couldn't figure out my position at all.	Yes.	Yes.	No.

Table C.7: Summary of the structured interview results for the original version of the home page of IMDB (see Figure C.5).

Questions	P1	P4	P5	P6	P7	P8
1. Do you know (do you have an idea) how long this page is (we are not looking for the exact size but trying to find out if you have an idea about the size?)	No.	Not long at all.	Very wage idea.	Yes.	Much much better.	At least, I know sections.
2. Do you think now you have an overview of the page?	Yes.	Yes.	Yes.	Yes, easily.	Yes.	Yes.
3. How difficult was it to get such an overview?	Easy.	No.	Difficult.	No.	Precise.	Easy.
4. What do you think about the amount of information?	User-friendly version.	Clean.	A lot.	Less confusing.	Fine.	Fine.
5. Did you find this page complex? If yes How complex you think is this page?	Less complex, usable navigation mechanism.	No.	Complex.	No.	Simpler.	Simple.
6. Was there anything particular you found repetitive in the page?	No, sections were clearly stated.	No.	Simple.	No.	No.	No.
7. How easy was it to access different parts of this page?	Very easy.	Easy.	Difficult.	Movement was difficult.	Not so difficult.	No.
8. How easy was it for you to figure out your position in a specific section with respect to other sections in the page?	Quite different from the original.	Easy.	No.	Not difficult.	Easy.	Easy.
9. At any point in your journey, did you think you were lost (didn't know where you are)?	No	No.	No.	No.	No.	No.

Table C.8: Summary of the structured interview results for the transformed version of the home page of IMDB (see Figure C.6).