Technologies to enhance self-directed learning from hypertext

Iyad Mohammed AlAgha

Technologies to Enhance Self-Directed Learning from Hypertext

Iyad Mohammed AlAgha

A Thesis presented for the degree of Doctor of Philosophy

2009

Department of Computer Science

University of Durham

Abstract

With the growing popularity of the World Wide Web, materials presented to learners in the form of hypertext have become a major instructional resource. Despite the potential of hypertext to facilitate access to learning materials, self-directed learning from hypertext is often associated with many concerns. Self-directed learners, due to their different viewpoints, may follow different navigation paths, and thus they will have different interactions with knowledge. Therefore, learners can end up being disoriented or cognitively-overloaded due to the potential gap between what they need and what actually exists on the Web. In addition, while a lot of research has gone into supporting the task of finding web resources, less attention has been paid to the task of supporting the interpretation of Web pages. The inability to interpret the content of pages leads learners to interrupt their current browsing activities to seek help from other human resources or explanatory learning materials. Such activity can weaken learner engagement and lower their motivation to learn.

This thesis aims to promote self-directed learning from hypertext resources by proposing solutions to the above problems. It first presents Knowledge Puzzle, a tool that proposes a constructivist approach to learn from the Web. Its main contribution to Web-based learning is that self-directed learners will be able to adapt the path of instruction and the structure of hypertext to their way of thinking, regardless of how the Web content is delivered. This can effectively reduce the gap between what they need and what exists on the Web. SWLinker is another system proposed in this thesis with the aim of supporting the interpretation of Web pages using ontology based semantic annotation. It is an extension to the Internet Explorer Web browser that automatically creates a semantic layer of explanatory information and instructional guidance over Web pages. It also aims to break the conventional view of Web browsing as an individual activity by leveraging the notion of ontology-based collaborative browsing.

Both of the tools presented in this thesis were evaluated by students within the context of particular learning tasks. The results show that they effectively fulfilled the intended goals by facilitating learning from hypertext without introducing high overheads in terms of usability or browsing efforts.

Acknowledgements

This thesis would never have been completed without the support of many people. In particular, I owe my respect and gratitude to:

my father (may Allah bless his soul), who gave me so much of who I am;

my mother, for always being there to love me and to provide immeasurable support throughout my lifetime;

Noor, my wife, for her love, support and endurance;

Dr. Liz Burd, first for providing me with the opportunity to do a PhD, which has been a turning point in my life, and then supporting me with ideas, advice and counsel;

Vin McIntyre, for his sincere help and support throughout my stay at Durham, and for proofreading this thesis;

the Trustees of the Durham Palestine Educational Trust, for their scholarship to study for a Master's degree at Durham, and for their endless support for me and for the Palestinian cause;

Janet Lavery, Dr. Phyo Kyaw and Dr. Andrew Hatch, for their assistance in evaluating the work in this thesis;

and, finally, my honour and respect are owed to *my people in Gaza Strip* for their great endurance and resistance to the Zionist occupation.

Copyright

Declaration

No part of the material provided has previously been submitted by the author for a higher degree in the Durham University or in any other University. All the work presented here is the sole work of the author and no-one else. The following publications were produced during the course of this thesis:

- AlAgha, I., and Burd, L., "Towards a Constructivist Approach to Learning from Hypertext", *In Proceeding of the 20th ACM Conference on Hypertext and Hypermedia*, Torino, Italy, 2009 [to appear].
- AlAgha, I., and Burd, L., "Knowledge Puzzle: A Tool for Constructivist Learning from Hypertext", *In Proceeding of the 9th IEEE International Conference on Advanced Learning Technologies*, Riga, Latvia, 2009 [to appear].
- AlAgha, I., and Burd, L., An Ontology-Based Approach for Context-Based Collaborative Browsing, *In Proceeding of the 9th IEEE International Conference on Advanced Learning Technologies*, Riga, Latvia, 2009 [to appear].
- AlAgha, I., and Burd, L., Empowering Web-Based Learning with Semantic Web Technologies: The Case of SWLinker, *In Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies*, Santander, Spain, 2008, pp. 396-398.
- AlAgha, I., and Burd, L., An Ontology based System for Reusing Components of Web Learning Resources, *In Proceedings of the 7th IASTED International Conference on Web-Based Education*, Innsbruck, Austria, 2008, pp. 281-286.
- AlAgha I., and Burd L., Personalisation of Reusable Learning Objects on the Semantic Web, *In Proceedings of the 8th Annual Conference of the Subject Centre for Information and Computer Science*, University of Southampton, 2007.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Research Overview

The Web provides access to information and knowledge sources that are practically unlimited. Its convenience and interactivity enable and encourage people to use it to learn unfamiliar topics. Hypertext resources (e.g. Web pages) have become an important educational medium. They can offer new possibilities to structure, represent, adapt and integrate various learning content and materials. Although research papers can provide more reliable resources for learning, the complexity of research papers is not always appropriate for users who wish to obtain general overview of unfamiliar topics. Moreover, the focuses of research papers are usually narrow and they rarely discuss intuitive and application issues. In contrast, many Web pages often contain intuitive descriptions of the topics or technologies. This makes these pages valuable knowledge resources, especially for non-researchers who constitue the majority of Internet users.

Learners usually navigate Web pages in self-directed ways. However, self-directed learning, in spite of its importance, places on learners the need for additional effort in all phases of the learning process. This effort, if not managed properly, can distract learners from their learning objectives and make them end up with incomplete or incoherent knowledge.

The purpose of this thesis is to present approaches to promoting self-directed learning from hypertext learning resources. The focus will be on providing solutions for problems encountered in two fundamental disciplines of learning, namely knowledge construction and knowledge aquisition. The following two sections discuss these problems.

1.2 Knowledge Construction with Hypertext

Knowledge construction with hypertext is known to be highly effective, because learners can direct their own routes through it (Conklin, 1987). Self-directed learning from hypertext implies that learners have different viewpoints and thus they may navigate different paths. Accordingly, they will have different interactions or discourses with the knowledge. However, hypertext resources, due to their static design, cannot cater for individual differences among learners. In addition, these resources cannot be restructured or interlinked to match the knowledge structure in the mind of every single learner, unless the learner is him/herself the content author. This gap between the knowledge structure in the learner's mind and the lack of a knowledge structure on the Web requires learners to make extensive cognitive and navigational effort to monitor their progress and to recall knowledge components while navigating in hyperspace.

In order to clarify the above problem, one can imagine a scenario of a student exploring the Web for the aim of learning about the topic of computer viruses. While navigating, he comes across different pages, each of which explains a different aspect of the intended topic. For example, he may obtain information from one Web page about how viruses spread. Then, from another page he learns about virus protection, while knowledge of the impact of viruses is obtained from a third page. The student constructs knowledge by thinking about the content and making semantic links between the pages he has visited. However, these pages may not be interlinked or organized on the Web in a way that the student needs since there are no direct hyperlinks between them. In addition, he cannot alter the structure of pages to cope with his needs. Therefore, the student will encounter cognitive and navigational difficulties when he starts reviewing the acquired knowledge due to the difference between what he needs and what actually exists on the Web.

It is clear from the above argument that there is an emerging need for innovative user-centric technologies that enable learners, rather than content authors, to be "active" players who can manipulate and interlink information on the Web in accordance with the progress of their own learning processes, instead of being conventional readers or viewers of information. Learners need a way to physically integrate segments of information on the Web in order to make them intelligible. Although this integration is often performed cognitively inside the human mind, if large amounts of information need to be retained the cognitive efforts needed to process it can cause learners to fail in knowledge construction. It should be emphasized here that learners should have the tools to build new associations of Web content even if they have no control over updating or adapting the Web content. The desired technologies should also work with any Web page without the need for any prior design settings or configurations.

In order to support the need for such user-centric technologies, we refer to a classic theory of pedagogy. Bloom's (1956) taxonomy identified six levels of cognitive learning, starting with simple knowledge recall, and becoming increasingly more complex, through comprehension, application, analysis and synthesis until the highest level, evaluation, is reached. While the current paradigm of learning from the Web operates at the "knowledge recall" level, the higher levels of learning expect the learner to interpret, rethink and repurpose the retrieved knowledge. Thus, an outstanding issue for Web-based learning is how to enable learners to use static hypertext resources, which provide only passive access to information, to progress to higher cognitive activities such as applying, analysing and evaluating the knowledge they glean. Resolving this issue will be a central theme in this thesis.

1.3 Knowledge Acquisition in Hypertext

People are increasingly using hypertext resources on the Web to learn unfamiliar topics. However, understanding the content of a Web page requires prior knowledge about its embedded terms. If a Web page includes terms that are unfamiliar, a learner must refer to helping resources in order to acquire the missing information and then become able to resume reading the page. These helping resources can include other Web learning resources (e.g. knowledge portals and Web-based courses), people or books that can provide assistance concerning the unknown topics. However, this activity is usually associated with the following difficulties that may affect the learner's self-motivation:

1. **Interruptions in the learning process**: It can be tedious and time consuming for learners to interrupt the learning task in order to seek help every time they encounter difficulties with interpreting the learning content. Too many interruptions in the learning process can lower the learners' engagement levels and lead to discouragement and boredom (Barr and Feigenbaum, 1982).

2. **Sub-topics or salient concepts of the topic:** When a learner tries to learn about a new topic, typically he wants to know what the prerequisite knowledge, sub-topics and/or related concepts of the topic are (Liu et al., 2003). This enables the learner to gain in-depth and more complete knowledge of the domain. However, a single Web page is often different from a normal book or survey paper in that it is unlikely to contain information about all the key concepts and/or sub-topics of the topic. This is due to the fact that the author of the page may not be interested in, and/or not expert on, every aspect of the topic. Thus, information about prerequisites and sub-topics probably needs to be discovered from multiple Web pages. However, referring to multiple pages will lead to (1) recursively.

3. **Users with similar or related interests**: If learners want to share ideas or seek assistance from other users on the Web concerning unfamiliar topics, it is necessary for them to contact users who have similar or related interests. Although hypertext resources can be accessed by several users at the same time, Web browsers are basically single-user tools. Learners are isolated when browsing the Web since they have no way of sharing online their browsing activities with other users. It would be a boon to self-directed learning if users who have similar or related interests were able to share their browsing activities (for instance, the pages they have visited).

Due to the above reasons, Web-based learners often find it difficult to interpret the contents of Web pages independently or communicate with others for help while browsing the Web. While many efforts have investigated how to improve search techniques to facilitate finding resources on the Web, few efforts have been oriented towards supporting the interpretation of their contents, which is the core objective of the learning process, without affecting learner engagement and awareness. This thesis aims to propose potential solutions for the above problems by leveraging Semantic Web technologies.

1.4 Research Contributions

This thesis provides an empirical work promoting self-directed learning from hypertext resources. Within this, the key contributions can be highlighted as follows:

1. **A tool for knowledge construction from the Web**. This thesis proposes a tool of multiple services, called the Knowledge Puzzle, to provide a constructivist approach for knowledge construction and review from hypertext resources. The proposed tool implies the following sub-contributions:

    a. **Adapting the information map on the Web to the user's information needs**. It helps the learners, rather than content authors, to repurpose and interlink segments of Web information in a way that enhances accessibility and caters for their individual needs. Such adaptive technique is distinguished by being entirely user driven, and being without any restrictions or preconditions.

    b. **Generating a hypertext format of the constructed knowledge as a support for self-review**. It helps learners represent the knowledge that they gain from the Web in a hypertext format in order to facilitate knowledge reviewing.

2. **A framework for knowledge acquisition from the Web**. This thesis proposes a framework, called SWLinker, to fulfil the aim of promoting knowledge acquisition from the Web in a way that sustains learner engagement and reduces the navigation effort. The proposed framework incorporates the following sub-contributions:

    a. **The use of semantic annotation to support interpretation and in-depth learning**. The thesis proposes an approach to on-demand semantic annotation that enables learners to associate semantics with the Web pages being browsed. Subsequently, it defines how to employ these semantics in the learning context in a way that helps learners gain a more in-depth and complete knowledge of the domain of interest.

b. **An ontology based approach for collaborative browsing.** This thesis presents, through the SWLinker framework, an ontology based approach for collaborative browsing. The approach utilizes the context behind the documents being browsed to serve as a unifying factor for sharing browsing activities across the Internet.

1.5 Criteria for Success

The five main criteria for this thesis are set out below. Their successful achievement can be measured by evaluating the impact of the proposed technologies on supporting self-directed learning from the Web. This impact can be assessed by analysing and comparing the learning outcomes when the proposed technologies are used to the outcomes when they are not used.

- **Support knowledge construction from the Web**

  This criterion will assess if the proposed technique for knowledge construction should enable learners to effectively manage, structure and evaluate the knowledge they gain from the Web with the least effort.

- **Reduce the impact of problems associated with navigation in hyperspace**

  Since the focus is on learning from hypertext resources, this criterion implies that the proposed approach should mitigate, as much as possible, the effects of disorientation and cognitive overload, the two main problems that learners experience when trying to navigate within hypertext systems.

- **Enhance and sustain learner engagement during self-directed learning**

  The on-demand annotation of Web pages should keep learners better engaged and more highly motivated by reducing the need to suspend the learning task in order to seek help.

- **Support in-depth learning while browsing learning resources on the Web**

This criterion will assess if the annotation service helps learners gain more complete and in-depth knowledge while they browse the Web.

- **Improve collaboration by bringing together users with related interests.**

  This criterion will assess if the user-matching algorithm adopted in the co-browsing service can effectively match and interlink users based on their semantically related browsing activities.

These criteria will be revisited and discussed in the final chapter of this thesis.

1.6 Thesis Outline

The structure of the thesis is as follows:

Chapter 2 provides an overview of self-directed learning from hypertext. It discusses its requirements, challenges and pedagogical foundations based on current learning theories. Then, a discussion on the stages and requirements of knowledge construction from the Web is given, presenting the background of the Knowledge Puzzle tool proposed in chapter 4. The various techniques and methods to promote navigational learning and knowledge construction from the Web are then discussed, with particular attention paid to their limitations and our view on how these limitations can be overcome.

Chapter 3 discusses the background and the context surrounding the SWLinker framework proposed in chapter 5. It sheds some light on the Semantic Web, ontology languages and the use of ontologies in the educational domain. Then it introduces the reader in some detail to the different platforms, frameworks and tools used for semantic annotations. It also highlights some important guidelines and requirements that need to be considered when designing an annotation tool for an eLearning domain. Finally, our view is introduced and discussed.

Chapter 4 discusses the process of designing and developing the Knowledge Puzzle tool for knowledge construction from hypertext resources. First, the requirements for the tool are presented. The general design and implementation detail are outlined, providing scenarios and details of how self-direction and cognitive skills are promoted with the proposed tool. Following that, the tool's functionalities are discussed and justified from an educational perspective. Finally, the chapter highlights some of the design and implementation patterns used.

Chapter 5 presents the SWLinker, a framework that leverages Semantic Web technologies to enhance self-directed learning from hypertext resources. First, the tool is discussed and usage scenarios are presented to give an overview of the system's functionalities, highlighting the expected educational benefits. Subsequently, the system architecture, ontological foundation and implementation details are discussed. Finally, the system is compared with related work and the conclusion is presented.

Chapter 6 presents a comprehensive framework to evaluate the usefulness and the outcomes of the proposed techniques for learning from hypertext. For each phase of the evaluation a detailed analysis of the outcomes are accompanied with a thorough discussion.

Chapter 7 concludes and summarises the work contained within the thesis and suggests future work that could develop the state of the art.

Chapter 2
Self-Directed Learning from Hypertext

2.1 Introduction

In order to develop suitable methods for self-directed learning from hypertext, it is necessary to have a clear understanding of several research areas. This chapter starts by giving an overview of hypertext and its advantages for learning. Then, it introduces the requirements of self-directed learning and its applications in Web-based learning. An overview of learning theories and other pedagogical foundations is presented, highlighting their roles and applications in Web-based learning. The phases of knowledge construction from the Web are then discussed. This is followed by a detailed survey and discussion of the techniques that support Web navigation and knowledge construction, with attention paid to their limitations. Finally, this chapter presents our views on how to overcome these limitations.

2.2 Hypertext

The term "hypertext" was coined by Theodore H. Nelson in the 1960's (Nelson, 1968), who defined it as a *"... nonsequential writing--text that branches and allows choices to the reader, best read at an interactive screen. As popularly conceived, this is a series of text chunks connected by links which offer the reader different pathways"*.

Landow (1992) furthered Nelson's thinking by explaining that hypertext is *"text composed of blocks of words (or images) linked electronically by multiple paths, chains, or trails in an open ended, perpetually unfinished textuality described by the terms link, node, network, Web, and path"*.

Hypertext organizes information in sets of units connected by associative links. A hypertext system works like a database that stores materials and offers users maximal freedom to navigate through hyperspace (Chou, 1999). One of the main applications of hypertext is the delivery of Web-based learning courseware. Liu (1994) identified four major advantages of hypertext in learning:

- **Nonlinearness**, which means that there is no specific sequence of proceeding from one point to another in hypertext, a feature that accommodates individuals' different needs.
- **Associativity**, which means the ability of hypertext to represent knowledge by making meaningful connections among the ideas.
- **Flexibility**, from the learner's perspective, means that hypertext imposes no arbitrary sequence as to how to proceed. From the author's point of view, hypertext is flexible in terms of its ability to be modified and updated without affecting the entire system.
- **Efficiency**, which means that hypertext has the ability to bring different forms of information such as text, graphics, sound and animation to the screen simultaneously and by the least effort.

2.3 Difficulties of Learning from Hypertext

In order to enhance learning from hypertext, it is essential to have a thorough understanding of any associated problems. Conklin (1987) identified cognitive overload and disorientation as the main problems that users experience when trying to navigate within hypertext systems. These problems are explained in the following sections.

2.3.1 Cognitive Overload

The exploration of hypertext resources on the Web requires learners to make cognitive efforts to recall the information they have previewed and to retain the semantic relationships they have created between various pages (Schunk and Zimmerman, 2001). Conklin (1987) characterized cognitive overload as "... *the additional effort and concentration necessary to maintain several tasks or trails at one time*". The reason for cognitive overload lies in the limited capacity of human information processing. Any effort additional to reading may reduce the mental resources available for comprehension.

There are two sources for cognitive load: intrinsic and extraneous; the former cannot be avoided while the later can be reduced. Intrinsic load refers to the number of cognitive elements that need to be processed simultaneously for knowledge construction, and how well the learner handles it depends on the rationale complexity of the learning task and the expertise of the learner (Gerjets and Scheiter, 2003). However, intrinsic load cannot be avoided as it is fundamental to the very nature of the learning process (Sweller and Chandler, 1994).

On the other hand, extraneous cognitive load is defined as the "... *one that is imposed purely because of the design and organization of the learning materials rather than the intrinsic nature of the task*" (Sweller and Chandler, 1994). It occurs when learners are involved in cognitive activities that are not directed towards the fulfilment of the specified learning outcomes. Thus, course contents and requirements should be carefully designed to eliminate irrelevant cognitive activities in order to reduce extraneous cognitive load and thus facilitate learning (Sweller et al., 1998). Chandler and Sweller (1991; 1992) identified two reasons for extraneous cognitive overload: split attention and redundancy. The split attention effect occurs when the learners have to divide their attention among multiple information sources and then cognitively integrate segments of information to make up their knowledge. The redundancy effect occurs when the learners process duplicate information from different sources.

Some writers have suggested that cognitive effort can enhance learning outcomes. For example, Hübscher and Puntambekar (2002) argued that the use of too much navigation support, with the aim of reducing the cognitive overload, can be detrimental to the learner because it frees him/her from thinking. Carroll et al. (1985) observed that forcing students to discover the information necessary to learn a word processor improved their learning more than directly presenting the same information to them. Charney and Reder (1986) also reported that, when tested, students who had learned commands for a computer application without guidance solved problems more efficiently than other students who had been told how to use the commands. These findings suggest that is it not necessary to eliminate the cognitive load completely as it could be fruitful for learning to involve some metal efforts. However, the most important issue is how to control these efforts so that learners will not end up being overloaded. Such overloading may cause students to lose their motivations to learn. Therefore, one of the aims in this thesis is to control the extraneous cognitive overload associated with self-directed learning from hypertext particularly by stimulating learners to self-regulate their cognitive activities and by avoiding splitting the learner's attention during Web navigation.

2.3.2 Disorientation

The discovery of interesting information on the Web requires the reader to scan pages thoroughly and possibly to follow several links in the process of identifying material relevant to his or her interests. However, learners often fail in knowledge construction since what and why they have explored so far becomes hazy as the exploration progresses (Kashihara and Hasegawa, 2003). They may end up being disoriented or lost in hyperspace. Disorientation is due to loosing the link between the subject being searched for and the information shown on the screen. It is caused by the absence of reference points for users as they travel through the Internet (Conklin, 1987). They need to know where they came from, where they are now and how to move from one place to another. Such disorientation can cause learners to take a longer time to complete their task and they can be distracted in the process. Elms and Woods (1985) outlined the three most common causes of user disorientation as follows:

- Users not knowing where to go next.
- Users knowing where to go but not knowing how to go there.
- Users not knowing where they are within the overall structure.

Nielsen (1990) observed that even in small hypertext environments, users could lose their orientation if no orientation clues are provided, especially when the user is given a large number of choices and needs to make decisions about which links to follow and which to ignore.

2.4 Self-Directed Learning (SDL) from the Web

The term Self-Directed Learning (SDL) originated in the field of adult education (Roberson, 2005). SDL has been defined as "*a process in which individuals take the initiative, with or without the help of others, to diagnose their learning needs, formulate learning goals, identify resources for learning, select and implement learning strategies, and evaluate learning outcomes* "(Knowles, 1975). Closely related terms include self-regulated learning, self-planned learning, autonomous learning and independent learning (Hiemstra, 1996). Zimmerman (2002) claimed that SDL leads to higher learning outcomes. Long (2000) pointed out that SDL contains three dimensions: motivation, metacognition, and self-regulation. Ertner and Newby (1993) characterized three components of SDL: the planning, monitoring and evaluating of one's learning activities. Self-directed learners must be able to plan, monitor, and evaluate their learning processes and outcomes.

With the growing trend toward Web-based learning, the concept of SDL has received increasing attention. Hanna et al. (2000) stressed that SDL is a key factor in successful Web-based learning. Similarly, Guglielmino and Guglielmino (2003) believed that, although the learners' technical skills are essential for successful e-learning, capacity for self-direction is even more important. This supported the claim by Cennamo et al. (2002) that success in Web-based courses often depends on the learners' abilities to successfully direct their own learning efforts and to decide on suitable navigation paths. Learners in online environments who are skilled in self-direction become more responsible for their learning and more self-motivated (Chang, 2005).

However, these positive effects of SDL in learning from the Web have to be balanced against the problems of learning from the Web, and in particular the problems of disorientation and cognitive overload discussed in section 2.3. In addition, SDL has a number of requirements that should be considered to help learners take responsibility for their learning in a short time. Narciss et. al (2007) identified some requirements to avoid the pitfalls of self-directed learning in Web-based environments as the following:

- **Metacognitive requirements**, such as planning, monitoring, evaluating and revising. Metacognition and its components will be discussed in section 2.6.
- **Content related requirements**, such as organizing and processing the learning material and applying acquired knowledge.

The technologies proposed in this thesis aim to promote self-directed learning from hypertext by trying to fulfil the above requirements. While metacognitive requirements will be the theme of the Knowledge Puzzle tool presented in chapter 4, the SWLinker framework, presented in chapter 5, aims to support learning content processing and organisation.

A prerequisite for successful Web-based learning is careful consideration of its pedagogical foundations so any proposal that aims to enhance learning should be based on learning theories. Therefore, the following section presents the basics of learning theories.

2.5 Web-based Learning and Learning Theories

According to previous literature reviews (Gros, 2002; Steffe and Gale, 1995; Wilson, 1998), learning theories can be related to three main commonly accepted paradigms: behaviourism, cognitive constructivism, and social constructivism. In what follows, these theories are briefly introduced focusing on how they apply in learning environments.

2.5.1 Behaviourism

The theory of behaviourism is based on the study of learners' behaviours that can be observed and measured (Good and Brophy, 1990). It views the mind as a "black box" in that its response to a stimulus can be observed quantitatively without any need to investigate or refer to the metal processes that cause that response. Subsequently, responses can be reinforced with positive or negative feedback to condition desired behaviours. In terms of instruction, behaviourism regards learning as a passive process in which knowledge is transmitted from the instructor to the learners.

Behaviourism is often applied in distance learning environments or Web-based tutoring systems where the learner is usually interacting with a computer system, either locally or remotely. An instructional designer can design the courseware in such a way that the learner's behaviours are captured and analysed as the learner moves through the instruction, with frequent positive feedback provided to reinforce the learning and remedial instruction provided when mastery is not achieved, resulting in individualized instruction (Bigus, 2004).

While behaviourism promotes stability and certainty with respect to knowledge acquisition, it is criticized for stimulating surface learning without giving enough opportunities for learners to express their own ideas during the learning task (Spiro et al., 1991). It does not adequately address the acquisition of higher-level critical thinking skills and problem-solving.

2.5.2 Cognitive Constructivism

The theory of cognitive constructivism views learning as a process of active construction rather than a product of passive transmission of knowledge (Duffy and Cunningham, 1996). In terms of instruction, learning is an active construction process in which the learners add information from the environment to their prior knowledge and experience to construct a new knowledge base. The knowledge construction process requires cognitive skills, such as analysis and reasoning, and meta-cognitive skills, such as reflection and self-assessment (Duffy and Jonassen, 1992). Cognitive constructivism investigates ways to develop the learner's cognitive skills through stimulating recall of prior learning and enhancing learner retention.

In many respects the Web can be considered an ideal medium for constructivist learning. Wilson and Lowry (2000) pointed out that there is a great potential for constructivist learning on the Web as people use the Web all the time for self-directed learning. They suggested three principles of constructivist learning that are characteristic of learning on the Web:

- Access to rich sources of information.
- Meaningful interactions with content.
- Bringing people together to challenge, support, or respond to each other.

Brown (2000) regarded the Web as "*a motivator to innovation that places the control in the hands of the user as they explore and discover*". Through interacting with hypertext resources, learners gain and take control over their own learning. They try to discover knowledge and construct meaning by self-directed inquiry, guided activity, and discovery (Landow, 1992).

2.5.3 Social Constructivism

In the theory of social constructivism, learning emerges as a social activity. It occurs as learners exercise, discuss and develop their knowledge through discussion, collaboration and information sharing (Duffy and Jonassen 1992). Knowledge is constructed from social relationships through interaction between learners and other people (e.g. instructors, peers). This means that knowledge is created as it is shared, and the more it is shared, the more it is learned. Vygotsky (1986) argued that the way learners construct knowledge, think, reason, and reflect on it, is uniquely shaped by their relationships with others. He argued that the guidance provided by more experienced peers allows the learner to engage in levels of activity that could not be managed alone. Internet communication techniques such as e-mail, forum membership and social bookmarking could well claim to enable social constructivist learning on the Web because they build virtual communities of learners collaborating to achieve their goals.

Despite the fundamental differences between these three learning theories, instructional designers point out that a learning situation often requires a mix of behaviourism, cognitive and social constructivism (Karagiorgi and Symeou, 2005). In addition, what works in some learning situations may not be appropriate for others. Thus, designers must allow circumstances surrounding the learning situation to help them decide which theory is most appropriate for that situation (Moallem, 2001).

This thesis primarily focuses on the theory of cognitive constructivism and its application as the most appropriate to learning from hypertext. It investigates the hypothesis of transforming learners from passive consumers of information into active players who can freely manipulate information to cope with the progression of their cognitive models. However, a prerequisite for applying cognitive constructivism is an understanding of metacognition and how the brain stores and retrieves information. This helps to identify the ways in which learners construct knowledge from hypertext. Metacognition and its components are discussed in the following section.

2.6 Metacognition

Flavell (1976) defined metacognition as follows:

"*Metacognition refers to one's knowledge concerning one's own cognitive processes or anything related to them, e.g., the learning-relevant properties of information or data. For example, I am engaging in metacognition if I notice that I am having more trouble learning A than B; if it strikes me that I should double check C before accepting it as fact*".

According to Flavell (1987), metacognition has the following components:

1. **Metacognitive knowledge** (also known as metacognitive awareness), which refers to knowledge that is used to process information and manage thinking processes. It is devided into three categories: knowledge of person variables, task variables and strategy variables.

2. **Metacognitive regulation,** which is the set of activites that regulates cognition and helps learners control their learning.

Brown (1987) described metacognition as "*the degree to which learners are engaged in thinking about themselves, the nature of learning tasks, and the social contexts*". She also identified four metacognitive skills as the following:

1. **Planning**, which includes the deliberate activities that manage and organise the learning task. This comprises a set of behaviours such as establishing the learning goal, the learning sequence and the learning strategies.
2. **Monitoring**, which refers to the activities that moderate the current process of learning while it is taking place. For example, the learner can ask himself/herself questions like, "Am I on the right track?", "How should I do the next part?", "What information is important to do this task?", and "Depending on the results, should I adjust my work?"
3. **Evaluation**, which involves an assessment of the results of the learning and an appraisal of the learning activity. It can assist learners with developing the necessary skills and activities from which they can draw in novel situations where it may become applicable.

4. **Revising**, which refers to modifying previous plans regarding goals and strategies, and considering other learning approaches.

Although different researchers have put forward different definitions of metacognition and its components, the short definition of metacognition as 'an awareness of one's own cognitive activity' would be commonly accepted. This thesis adopts the views of Brown (1987) who, in seeking to improve learning outcomes, paid more attention to metacognitive skills than to metacognitive knowledge, because metacognitive skills can be practically employed and emulated in Web-based learning environments. Learners in hypertext environments should be able to make effective plans for navigation that reflect their self-awareness of their skills and their understanding of the task requirements. The learners also need to be self-directed so that they can monitor their own progress in learning, evaluate their work by themselves and select appropriate strategies to complete assigned tasks effectively. Therefore, the above metacognitive skills will be the foundations of the constructivist learning framework proposed in chapter 4 of this thesis.

2.7 Knowledge Construction from the Web

The knowledge construction process has dramatically changed due to the widespread use of the World Wide Web. In the past, learners were forced to use limited resources, such as the books and newspapers available to them. In the Internet era, learners can construct knowledge by exploring the almost limitless resources of the Web in a self-directed way. As learners may follow different navigation paths through hypertext environments, they will accordingly have different structures of knowledge. Several factors influence the choice of the navigation path, such as the task and the document structure as well as the aims and the prior knowledge of the learner (Wright and Lickorish, 1990). Therefore, Web-based learning tends to rely - more than is the case with other educational media - on self-direction and a personal construction of what is meaningful.

Hypertext learning resources generally provide learners with hyperspace where they can navigate in a self-directed way. This involves navigating through a structure of hyperlinks and processing the embedded contents. While they navigate, learners construct knowledge by thinking about the content and making semantic links between the Web pages they have visited. Mitsuhara et al. (2008) made a simple model of knowledge construction from the Web, shown in Figure 2.1, which consists of three phases: search, knowledge construction and reflection (knowledge reconstruction). These phases are explained as follows:

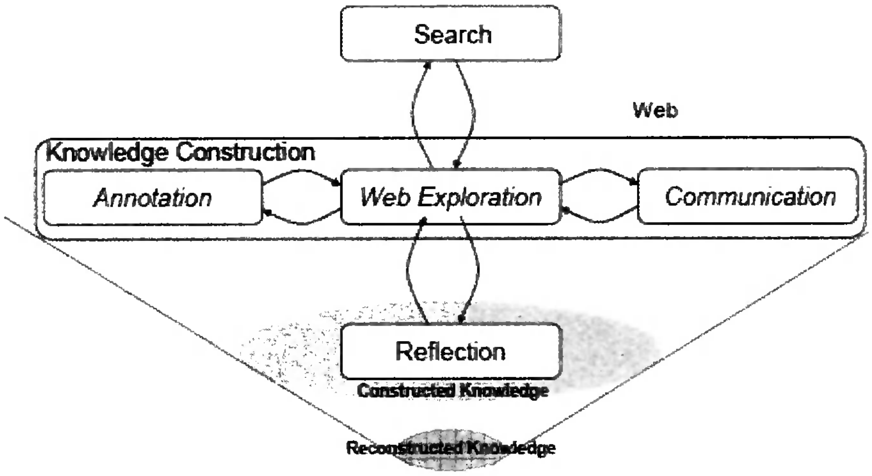

Figure 2.1: Model of knowledge construction from the Web (Mitsuhara et al., 2008)

1. **Search:** The process starts when the learner searches the Web for pages about his/her topic of interest. At this phase, a search engine is likely to be used and the learner must formulate a search query that is likely to present a list of Web pages relevant to the topic.

2. **Knowledge Construction:** This phase can be divided into three activities: Web exploration, annotation, and communication. Web exploration, which starts immediately after a successful Web search, is the fundamental activity of knowledge construction from the Web. It involves visiting Web pages by following hyperlinks and reading the content of visited pages. While navigating the Web, learners may need to mark (highlight) important information so that they can recognize the information of interest afterwards. Another possible activity is to annotate or attach their own notes to particular page contents. Such annotations can be very useful to knowledge construction and to organising that knowledge when the Web pages are revisited. Learners can, of course, add knowledge from resources independent of the Web, including human experts, to knowledge gained from the content of their

visited Web pages. A student who has a question or some difficulty often constructs knowledge in communication with peers, teachers, and/or people connected through popular internet communication methods such as e-mail, bulletin board, and chat (instant messenger).

3. **Reflection:** Knowledge construction in hyperspace requires learners to reflect on their navigation process. At this phase, a learner reconstructs his/her constructed knowledge by reflection methods. Reflection gives individuals opportunities to modify misconceptions or to improve inadequate understanding. One method of reflection is that the learner revisits Web pages in order to rethink the contents already explored. Empirical studies that have analysed the navigation process in hyperspace have shown that revisiting pages to rethink the knowledge gained is a common activity (Kashihara et al., 2000). In a seminal study of how users explore the Web, Tauscher and Greenberg (2001) presented the following statistics on the types of actions users may carry out:

   - 58% of pages visited are revisits,
   - 90% of all user actions are related to navigation,
   - 30% of navigation actions are through the 'Back' button,
   - less than 1% of navigation actions use a history mechanism

   A fairly obvious conclusion from such statistics is that, for most people, revisiting pages seems to be crucial for the knowledge reconstruction process. Furthermore, knowledge reflection requires learners to rethink not only the pages visited but also the reasons why they were visited since these reasons have a great influence on how they will shape their knowledge structure.

2.8 Learning Support through Web Navigation

Linard and Zeiliger (1995) pointed out that navigation occurs at two levels: in hyperspace and in the knowledge space. A clear distinction between these two levels of navigation should be made when discussing the relationship between learning and navigation issues in hypertext:

- **Navigation in hyperspace**: involves users steering a course through a structure of hyperlinks and moving from one Web page to another.

- **Navigation in the knowledge space:** involves users constructing knowledge and having some understanding of the topography of the domain in which they are immersed, how page contents are semantically related and which information contributes to their knowledge of the area under consideration. Navigating at this level can be a planned activity or it can be a more situated activity (i.e. responding to the environment). Navigation in the knowledge space involves decision-making and is inseparable from a consideration of context (Jul and Furnas, 1997).

Learning from the Web implies that the two navigation processes should take place at the same time. Many researchers have studied and proposed methods for facilitating Web navigation and reducing the problems associated with the two levels of navigation. In what follows these methods are reviewed, with special attention paid to discussing their relevance to self-directed learning from the Web.

2.8.1 Adaptive Hypermedia

The goal of adaptive hypermedia research is to improve the usability of hypermedia applications by personalizing them (Brusilovsky, 1996). Adaptive hypermedia can be useful in any application area where users of a hypermedia system have essentially different goals and knowledge and where the hyperspace is reasonably large. Adaptive hypermedia techniques provide a certain level of intelligence to hypermedia systems in the sense that they have the ability to "understand" the user and to adapt their behaviour to the user's needs. Having knowledge about the users, adaptive hypermedia can support them in navigation by limiting the browsing space and suggesting the most relevant links to follow, thus decreasing search and navigation time. They also seek to enhance comprehension of the content by presenting the most relevant information on a page and hiding information that is not relevant. Their aim is thus to solve the disorientation and cognitive overload problems. Adaptation is achieved by collecting information about the users while they interact with the system, and adapting the application based on this gathered information. This information is stored in the so-called user-model. Adaptive hypermedia provides two forms of adaptation:

1. Content adaptation (Wu et al., 1998), or adaptive presentation (Brusilovsky, 1999) - the system presents the content in different forms according to the domain and instructional model (e.g. concepts, their relations, prerequisites, etc) and the information from the user-model.

2. Link adaptation (Wu et al., 1998) or adaptive navigation (Brusilovsky, 1999), which means modifying the availability and/or appearance of links that appear on the Web page according to the information taken from the user-model. The two most popular forms of link adaptation are:

    i. **Link annotation**, which aims to provide additional information about the page to which the link points by using suitable visual indicators such as colour, text, or symbol, all determined by information from the user-model. For example, links to advanced topics can be shown in a different colour to indicate that the user is not ready to cover these topics until he/she has covered their prerequisites.
    ii. **Link hiding**, which makes some links invisible or inaccessible to the user if the system decodes from the user-model that these links would lead the user to irrelevant information.

In spite of the efficiency of adaptive hypermedia systems in managing aspects of the navigation process, they still have the following drawbacks for SDL:

1. Adaptive hypermedia systems are mainly based on the designer's predictions of learners' needs and these may not match their real needs. It is hard to believe that any adaptive system will be able to predict precisely what learners want as even human experts may fail to do so. Although there are some user-driven techniques that give learners control over the adaptation process (Tsandilas and Schraefel, 2003), this control is still limited to what the designer allows, and it requires specific design steps to be taken prior to the learning session.
2. Adaptive hypermedia systems are often applied in small environments and they cannot be generalised to all Web content. Adaptive hypermedia

techniques can adapt the path of instruction to the learner's needs within specific instructional delivery systems but a single learning session on the Web may involve Web pages from several sites. These sites may employ different adaptation strategies or may not employ any. Such diversity may considerably disturb the coherence of the learner's cognitive model.

3. Adaptive hypermedia systems are based on user-modelling. User-models are often persistent or change slowly, and their construction is based on assumptions that do not always hold. In SDL, the learner's desires and goals may change, evolve or propagate as the learning process progresses (Zimmerman, 2002). Thus, an adaptive hypermedia system can make incorrect guesses about what the learner wants or it may not be able to capture any shift in the learner's goals (Tsandilas and Schraefel, 2003).

For all the above reasons, we argue that AH techniques do not fulfil the requirements of SDL from hypertext where learners have the freedom to follow any navigation paths. In our view, what is needed is a solution that goes further by enabling the learner to inject new hyperlinks and annotations in the visited Web pages in accordance with his/her knowledge structure without any limitations or preconditions. Such solution will be discussed in detail in chapter 4.

2.8.2 Web Browsers

Web browsers such as Microsoft Internet Explorer and Mozilla Firefox can be regarded as Web-based learning tools for popular use. The design of Web browsers plays a crucial rule in mitigating the navigation and cognitive problems that learners may encounter. Web browsers should provide facilities that enable visited pages to be easily revisited. Bookmarking and browsing history lists are examples of this facility. Bookmarks enable users to store references to previously visited pages so that they can be revisited. Dias et al. (1999) pointed out that the existence of bookmarks is important not so much in avoiding disorientation problems but rather in enabling "recovering" from an eventual possibility of disorientation. However, Kashihara et al. (2000) argued that bookmarking does not really facilitate knowledge construction as it does not identify how the learners make sense of the visited Web pages or how they have mentally linked pages during the navigation process. The browsing history also does not identify why learners have explored particular pages and which information might be important in knowledge construction. Therefore, Web browsers are not very suitable for knowledge construction in hyperspace as they do not allow learners to sufficiently rethink knowledge that they have constructed so far (Kashihara et al., 2000).

2.8.3 Visual Navigation Tools

Visual representations of Web spaces have long been used to help users to avoid or overcome disorientation in hypertext (Nielsen, 1990). Their graphical features can be used to organise the navigation paths and to facilitate the chore of revisiting Web pages. The advantage of such tools over traditional Web browsers is that they can represent many pages in a small space; this means that they are not limited to sequential page displays and they are scalable as they can show both recent and distant pages (Cockburn and Greenberg, 2000). The visual representation of Web space enables direct access to any of the previously visited pages. Thus, they can help users orient themselves in the information space and overcome the problems of context and minimalist representation. Danielson (2002) pointed out that a visual map depicting the pages and links between locations visited in a Website is all that is necessary to decrease the level of disorientation learners experience when they navigate a site. Park and Kim (2000) agreed, finding that the provision of a navigational graph assists learners in orienting themselves, finding relevant textual information, and decreasing cognitive load because the visual representation of Web contents leaves more space in the working memory for processing information. Cockburn and Greenberg (2000) classified these techniques into four categorises:

1. Hub-and-spoke dynamic trees which are generated in response to the user's navigational acts.
2. Spatial or concept map organisations that aim to exploit people's memory for the spatial location of objects.
3. Site maps that show a topology of the physical storage locations of pages.
4. Temporal organisation schemes that exploit the user's memory for the timing of their actions.

Despite the efficiency of these visual navigation techniques in reducing disorientation and cognitive overload, they do not support knowledge construction from the Web because they do not enable user-controlled structuring of information and navigation paths (Tsandilas and Schraefel, 2003). Only spatial and concept mapping techniques can enable users to structure pages and define relationships based on their own viewpoints. The next section discusses the use of concept mapping as a support for Web navigation.

2.8.4 Concept Map Based Tools for Constructivist Learning

Concept maps are effective tools of knowledge representation that allow for the transformation of complex conceptual statements into an understandable format. They facilitate both teaching and learning. Novak and Cañas (2006a) defined concept maps as graphical tools for organizing and representing knowledge. For instance, in such a graphic, concepts might be written into circles or boxes, related concepts connected by lines and these lines annotated with words to indicate the relationships between concepts. A concept map can therefore be considered, simply, as a means through which learners view and represent relations 'between things, ideas or people' (White and Gunston, 1992) and concept maps therefore are "graphical tools for organizing and representing knowledge" (Novak, and Cañas, 2006b). In addition, concept maps can be seen as a constructivist approach to SDL as they encourage learners to practice metacognitive skills. Mintzes et al. (1997) regarded concept maps as "metacognitive tools" that stimulate learners to think reflectively about what they know through the visual representation of concept meanings and relationships. The process of creating and modifying a concept map requires the learners to plan, monitor and make decisions about the different ways concepts are related to one another (McAleese, 1998; Brown, 1987). Concept mapping has also been described in the literature as a "student-directed strategy that does not rely on teacher involvement" (Chularut and DeBacker, 2004).

Constructivist tools for Web navigation are those environments that enable learners to construct their knowledge while navigating the Web. These tools often provide learners with a concept-mapping space as a support for gathering, representing, structuring and creating information (Zeiliger et al., 1997). They often utilize two separate spaces: hyperspace in which to perform the navigation process (e.g. a space for Web browsing), and a concept mapping space for knowledge construction support. Examples of concept map based tools that use a constructivist approach for Web navigation are reviewed in what follows. Subsequently, the drawbacks of these tools are discussed.

Interactive history (Kashihara et al., 2004), shown in Figure 2.2, is a tool implemented as plug-in to Internet Explorer 4.0 or higher to support navigation planning and reflection for knowledge construction in hyperspace. It provides an interactive window where learners can annotate their navigation history. They can graphically construct sequences of Web pages to be visited so that their learning goals can be achieved. The planning process is done by adding references to the visited pages on the drawing area and then annotating the links between these references to represent the navigation goals. It also enables learners to take notes about the pages visited and to annotate the links. The tool also generates a special visualization called "knowledge map" from the annotated exploration history. The knowledge map visualizes the relationships between pages in the navigation path, and thus allows the learners to reflect on what they have constructed during exploration.

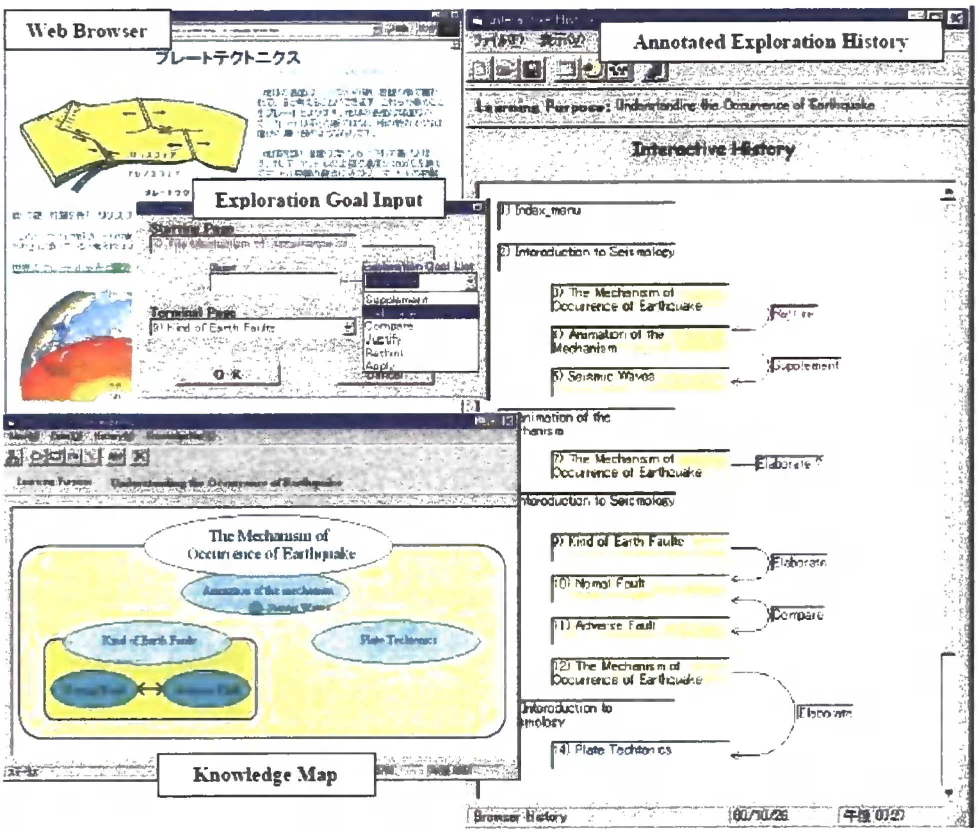

Figure 2.2: Interactive History (Kashihara et al., 2004)

Nestor Navigator (Khamidoullina et al., 2001; Esnault and Zeiliger, 1999; Zeiliger et al., 1999) is a Web navigator developed by CNRS-GATE laboratory and it is similar to Netscape or Internet Explorer. The Nestor main window, shown in Figure 2.3, is divided into two windows. The right window is a classical browser for navigating Web pages. The left window provides interactive and graphical support for knowledge construction. In this window learners can build concept maps to represent the semantic relationships between concepts explored in hyperspace. While the user is exploring the Web, a map is automatically drawn in the interactive window denoting the navigation path that has been followed. The learner can then edit the map, annotate nodes or build his own navigation paths. The drawn maps can help learners revisit Web pages in a specific order. NESTOR is designed to achieve the twin main goals of helping the trainee to become an active learner and making browsing easier for users who may have little experience of using the Internet.

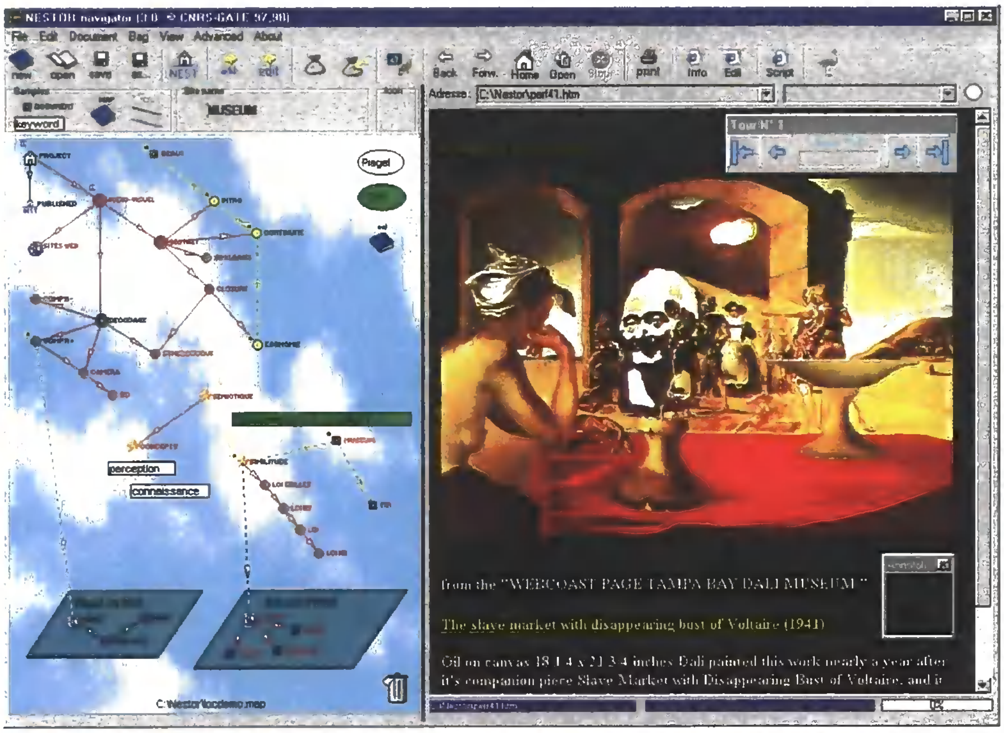

Figure 2.3: Nestor Navigator (Esnault L and Zeiliger, 1999)

NaVir (Djoudi, 1999) is a computer-aided system for virtual navigation of the Web. It is implemented in Java and can be used with Netscape, Internet Explorer or any other browser. It relies on a proxy server to pass information between the browser and the system. NaVir automatically generates navigation maps which allow the user to keep track of the path he has followed. Each page address (URL), the topic or title of the page, and the time spent connected to this page are kept in the nodes. The map is displayed upon request by the user at any stage of the browsing session. In addition, the system allows the map to be drawn from a list of identifiers of pre-selected pages. Also the user can follow the map's evolution by the creation or deletion of any link or by reorganization of the graph. The generated navigation maps can be shared among a group of users. The user has also the ability to save or print a map or to reopen the map constructed during a previous navigation session.

Despite the efficiency of concept-map based navigation tools in providing a constructivist approach to Web-based learning, the main pitfall of these tools is the separation between the navigation and planning processes, which are often performed in different spaces. This separation splits the attention of the learners and places an additional cognitive load on them as they are required to continually move between hyperspace, where they navigate the Web pages, and the concept mapping space, where they structure and visualize what they gained from the Web (Chandler and Sweller, 1992; Cockburn and Greenburg, 2000). In addition, when the map gets larger, the learner can be distracted and cognitively overloaded due to the effort required to trace a large number of nodes and the links among them. Jones (1987) made this point when he observed in his experimental work that the person's ability to use a hypertext map can quickly deteriorate as the number of units and connecting links increases. Although a concept-map based tool has been designed in this thesis as part of the tool presented in this thesis for knowledge construction, we have tried to resolve the above problem by reducing the dependence on the designed tool by using it only during the planning and monitoring stages of the learning process. This design decision will be discussed in detail in chapter 4.

2.9 Concluding Remarks

Learning is a process of knowledge construction not of knowledge recording or absorption (Harel and Papert, 1991), and it requires environments in which learners can be active designers and contributors rather than passive consumers (Fischer, 1998). Techniques such as adaptive hypermedia, bookmarking and visual navigation are effective in supporting navigation in hyperspace by mitigating the effect of disorientation and cognitive overload. However, they provide limited support for knowledge construction.

In contrast, navigation planning and concept mapping tools such as Nestor, Interactive History and NaVir provide potential support for knowledge construction by providing a separate space for information gathering, structuring and planning. However, their characteristic separation between the construction space and hyperspace imposes an additional cognitive burden on learners as it requires them to split their attention.

The vision of this thesis is to develop a tool, named the Knowledge Puzzle, that aims to fill the gap between navigation in knowledge space and navigation in hyperspace by adapting the information structure on the Web to the learner's information needs. The proposed tool leverages the theory of cognitive constructivism to study how the brain stores and retrieves information in order to identify the ways in which learners construct knowledge from hypertext. Learners are provided with a meta-cognitive space (e.g. a concept map tool) that enables them to represent and visualize the knowledge gained from the Web. Subsequently, the constructed knowledge map is converted to a hypermedia layer and then attached to the visited pages on the Web. The new hypermedia layer will highlight, annotate and interlink information components on the Web to reflect the knowledge map constructed by the learner. If the hypertext becomes restructured and annotated according to learners' needs and in a way that matches their cognitive aims, this can considerably enhance accessibility and thus reduce disorientation and cognitive overload. The Knowledge Puzzle tool along with the design decisions will be the theme of chapter 4.

Chapter 3
Semantic Web and Education

3.1 Introduction

Ontologies are the key concept in the development of the Semantic Web. This chapter highlights the importance of ontologies in different educational contexts. The definition, languages, classifications and building of ontologies are reviewed. This chapter also discusses semantic annotation and presents in some detail the different platforms, frameworks and tools used for semantic annotation. Some important guidelines and requirements that need to be considered when designing an annotation tool for e-learning are also highlighted. Finally, our views are presented and discussed.

3.2 The Semantic Web

The problem of the current implementation of the Web is that it is only understandable by humans. Machines cannot interpret information on the Web as people do. To explain this problem, imagine that a user wants to search the Web for the term 'apple', by which he/she means the fruit. The results that will be returned by the search engine will have no semantic relationship. The search may give Web pages for the 'Apple' computer company, or the fruit, or even for an online shop named 'Apple'. This ambiguity in the results could be removed and intelligent results could be obtained if resources on the Web were semantically annotated. This can be done by adding an extra layer of semantics to the current Web. The Semantic Web aims to improve the existing Web with a layer of machine interpretable metadata so that a computer program can understand what a Web page is about, and therefore draw conclusions. The Semantic Web as defined by its creator, Tim Berners-Lee (Berners-Lee et al., 2001), implies:

"... *an extension of the current Web in which information is given a well-defined meaning, better enabling computers and people to work in cooperation.*"

To add the layer of semantics to the existing Web, three challenges need to be successfully overcome (van Harmelen, 2004):

- A syntax for representing metadata,
- Vocabularies for expressing the metadata, and
- Metadata for lots of Web pages.

The Semantic Web includes the following technologies (Antoniou and van Harmelen, 2004):

- **Explicit Metadata**: the Semantic Web does not rely on text-based manipulation, but on machine-processable metadata.
- **Ontologies**: an ontology can be defined as an explicit and formal specification of a conceptualization (this topic is discussed in depth in the next section).
- **Logic and Inference**: where automated reasoners can infer conclusions from the given knowledge.
- **Agents**: are computer programs that work autonomously on behalf of a person. They receive tasks to accomplish, make certain choices and give answers.

As ontologies represent a core component in the Semantic Web, this chapter will make a thorough examination of ontologies and their applications in learning technologies. Thus, ontology types, design principles, ontology languages and the different approaches to building ontologies are discussed. In addition, some applications of ontologies will be reviewed. Finally, there will be a comprehensive discussion about the different semantic annotation techniques and methods that have been used in most semantic annotation tools.

3.3 Ontologies

Kalfoglou (2001) defined ontologies as follows:

"... an explicit representation of a shared understanding of the important concepts in some domain of interest. The role of an ontology is to support knowledge sharing and reuse within and among groups of agents (people, software programs, or both). In their computational form, ontologies are often comprised by definitions of terms organised in a hierarchy lattice along with a set of relationships that hold among these definitions. These constructs collectively impose a structure on the domain being represented and constrain the possible interpretations of terms".

From this definition, it can be seen that ontologies have many useful features in intelligent applications. The most important features of using ontologies can be summarised in the following:

- Ontologies provide vocabularies and concept hierarchies for referring to the terms used in specific subject areas. They provide logical statements that explain what the terms are and how they are related to each other (Chandrasekaran et al, 1999).

- Ontologies enable interoperability and shared understanding among different applications. The main purpose of ontologies is not only to work as vocabularies or taxonomies, but also to facilitate knowledge sharing and knowledge reuse among applications (Gruber, 1993; Guarino, 1995). The ontology that describes a specific domain can be shared and reused among different agents and applications as a formal specification and knowledge base for those agents and programs.

Ontologies can be classified into four different types (van Heijst et al., 1997) namely, application ontologies, domain ontologies, generic ontologies and representation ontologies. Application ontologies represent the information needed to model a particular application. Domain ontologies define concepts and vocabularies that are specific to a particular domain. Generic ontologies define concepts that are generic across different disciplines. Finally, representation ontologies provide a representational framework with a neutral view of the world.

There is also another classification of ontologies based on their generality (i.e. scope) and expressiveness (i.e. level of details) (Bruijn and Fensel, 2005). Regarding the level of generality there are three different types of ontologies: top-level ontologies (e.g. CYC[1], WordNet[2]) which are shared by many people in different domains, domain ontologies (e.g. UNSPSC[3], The United Nations Standard Products and Services Code for classifying products and services) which are shared between stakeholders in a particular domain and finally application ontologies (e.g. an ontology for an academic course) which are used for a particular application.

The other orthogonal classification of ontologies is based on their expressiveness. Ontologies can be distinguished by their different levels of expressiveness such as: thesaurus (e.g. WordNet), controlled vocabulary (e.g. Dublin Core[4]), informal/formal taxonomy (e.g. Yahoo directory[5]/UNSPSC), frames (e.g. RDFS), value restrictions (e.g. OWL data-type), limited logic constraints (e.g. OWL DL[6]) and general logic constraints (e.g. CyCL[7], OWL DL). However, the level of expressiveness can be seen as two distinct categories (Bruijn and Fensel, 2005):

- Light-weight ontologies, which include the concepts and the relationships between the concepts.
- Heavy-weight ontologies, which include axioms and constraints.

This thesis will focus on the use of domain and application ontologies with a light-weight level of expressiveness.

---

[1] http://www.opencyc.org/ [last accessed 15/11/2008]

[2] http://wordnet.princeton.edu/ [last accessed 15/11/2008]

[3] http://www. unspsc.org [last accessed 15/11/2008]

[4] http://dublincore.org/ [last accessed 15/11/2008]

[5] http://dir.yahoo.com/ [last accessed 15/11/2008]

[6] http://www.w3.org/TR/owl-guide/ [last accessed 15/11/2008]

[7] http://www.cyc.com/cycdoc/ref/cycl-syntax.html [last accessed 15/11/2008]

3.4 Ontology Languages

A set of formal languages is used to express the semantics of a resource on the Web so that humans, as well as machines, can understand it. These languages can be stacked on top of each other to form what Berners-Lee (2001) called "*The Semantic Web Language Layer Cake*". Figure 3.1 shows the layers of the Semantic Web namely,

- The Unicode and URI layer which forms the base for the following layers.
- XML and XML Schema layer which forms the syntactical basis for the Semantic Web languages.
- RDF and RDF Schema layer which represents the expressive language for the Semantic Web.
- OWL layer, which represents the ontology language for the Semantic Web.

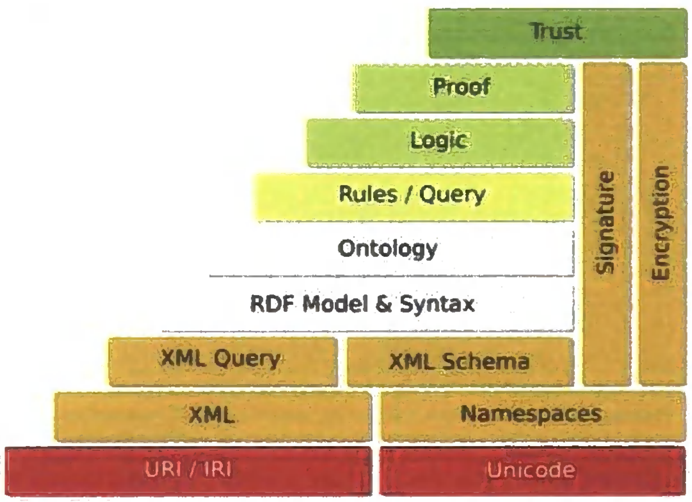

Figure 3.1: The Semantic Web Language Layer Cake (Berners-Lee, 2001)

An overview of each of the three languages used in the Semantic Web is presented in the following sub-sections.

3.4.1 XML/DTD/XML Schema

Although XML (eXtensible Markup Language) is not an ontology language, it is a core technology in the 'Layer Cake' and all the subsequent layers are built on top of it. XML is an application of SGML (ISO 8879). It is a structured language and was developed due to the shortcomings of HTML. XML is used to exchange data between Web applications. In order to support communication, Web applications need to agree on common vocabularies so that they can process exchangeable information regardless of the application design. Examples of XML common vocabularies are NewsML[8] (for news exchange), MathML[9] (for mathematics) and IEEE-LOM[10] (for learning objects).

3.4.2 RDF/RDFS

Resource Description Framework (RDF) is a language that provides a model for representing data about resources on the Web in the form object-attribute-value (O-A-V) triplets (Manola and Miller, 2004). Each RDF description can be also represented as a semantic graph or network whose parts are equivalent to RDF statements. RDF triples can be expressed in different ways: by using XML syntax (Figure 3.2).

```
<rdf:Description rdf:ID=" http://www.example.com/Ali">
      <Name>Ali</Name>
</rdf:Description>
```

Figure 3.2: RDF serialization using XML

or using N3/Notation (Figure 3.3).

```
<http://www.example.com/Ali> <Name> "Ali"
```

Figure 3.3: RDF in N3

or using binary predicate form, e.g. Property(object, value), Figure 3.4.

```
Name("http://www.example.com/Ali"," Ali")
```

Figure 3.4: RDF as a binary predicate

[8] http://www.newsml.org/pages/index.php [last accessed 15/11/2008].

[9] http://www.w3.org/Math/ [last accessed 15/11/2008].

[10] http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf [last accessed 15/11/2008].

or using a directed graph (Figure 3.5).

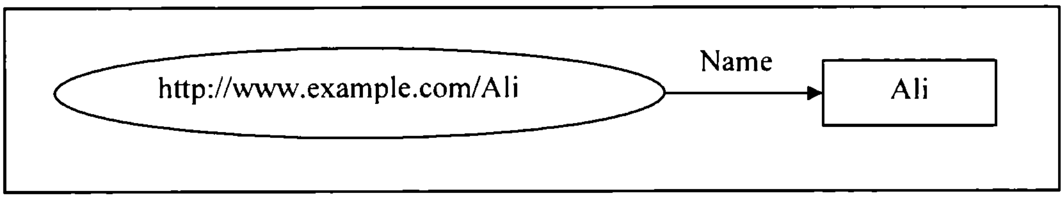

Figure 3.5: RDF as a graph

RDF is a domain independent language, in which no claim about a specific domain is made. This implies the need to define someone's own terminology using RDF Schema (RDFS). Also, RDF does not contain a vocabulary to author metadata, thus an RDF Schema is needed to define a predefined vocabulary to be used with metadata generation. RDF schema (RDFS) provides XML-based vocabulary to specify classes and their relations as well as to define properties and to associate them with classes (Brickley and Guha, 2004). The problem with RDF and RDFS is that they are quite simple compared to fully-fledged knowledge representation languages. For example, it is not possible in RDFS to specify cardinality constraints or to specify which classes are disjoint or equivalent. Due to these limitations, a more expressive language is needed (hence OWL).

3.4.3 OWL

OWL (Smith et al, 2004) is a semantic language that is a result of merging two other semantic languages; DAML (Hendler and McGuinness, 2000) and OIL (Fensel and Musen, 2001). OWL vocabulary includes a set of XML elements and attributes with well-defined meanings that can be used to represent domain concepts and their relations in ontology. In fact, OWL vocabulary is built on top of RDF(S) vocabulary, but it specifies further constraints and relationships among resources than are specified in RDF(S), such as cardinality, domain and range restrictions, inverse, disjunction, symmetry and transitive rules. Nowadays, OWL is the most commonly used language for representing ontologies. Another important feature in OWL is that it has a layered structure comprising three expressive sublanguages with the higher one built on top of the lower ones. These sublanguages are (Smith et al, 2004):

- **OWL Lite,** which supports building simple hierarchies and simple constraints.
- **OWL Description Logic (DL),** which provides maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). It includes all OWL language constructs, but it imposes certain restrictions on using them.
- **OWL Full,** is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees.

3.4.4 SPARQL

Unlike OWL and RDF(S), SPARQL is not intended for ontology representation, but for querying Web data; precisely it is a query language for RDF (W3C SPARQL, 2005). It can be used to:

- Extract information from RDF graphs.
- Extract RDF sub-graphs.
- Construct new RDF graphs based on information in the queried graphs.

Syntactically, SPARQL queries are of the form presented in Figure 3.6. Obviously, the syntax closely resembles that of database query languages such as SQL. The SELECT clause contains variables, beginning with "?" or "$". The WHERE clause contains a pattern. Prefixes are used as an abbreviation mechanism for URIs/namespaces and apply to the whole query.

```
SELECT     ?author
WHERE      {<http://www.library.org/books#Harry Potter>
           <http://www.library.org/elements/author> ?author}

PREFIX     library: <http://www.library.org/elements/>
SELECT     ?author
WHERE      {<http://www.library.org/books#Harry Potter>
           library:author ?author}

PREFIX     library: <http://www.library.org/elements/>
PREFIX     :<http://www.library.org/books>
SELECT     $author
WHERE      {:Harry Potter library:author  $author}
```

Figure 3.6: Examples of SPARQL queries

3.5 Building Ontologies

Ontologies are built either manually or semi-automatically (Gómez-Pérez and Manzano-Macho, 2004). While manual ontology building can be tedious and time-consuming, semi-automatic ontology building can substantially speed up the process of ontology generation.

The process of semi-automatic generation of ontologies is often referred to as an ontology learning process, which can be defined as:

"*the application of a set of methods and techniques used for building an ontology from scratch by enriching, or adapting, an existing ontology in a semi-automatic fashion using distributed and heterogeneous knowledge and information sources, allowing a reduction in the time and effort needed in the ontology development process*" (Gómez-Pérez and Manzano-Macho, 2004).

The process of ontology learning from text relies on a number of techniques that came from disciplines such as Natural Language Processing (NLP) and machine learning, and is applied to different types of unstructured, semi-structured, and fully structured data. These techniques can be summarized as follows (Gómez-Pérez and Manzano-Macho, 2004):

- **Linguistic techniques:** These include NLP techniques such as pattern-based extraction, semantic relatedness, etc. An example of a system using this technique is SOAT (WU and HSU, 2002).

- **Statistical techniques**: These techniques are based on different statistical measures (e.g. Term Frequency Inverse Document Frequency 'TFIDF') to assist the ontology designer to detect new concepts and the relationships between them. A sample system which uses this technique is WOLFIE (WOrd Learning From Interpreted Examples) (Thompson and Mooney, 1999).

- **Machine learning techniques**: Algorithms from the field of Machine Learning can help the ontology designer detect new concepts and their relationships, and then to place them in the correct position in the ontology.

OntoLearn (Navigli et al., 2003) is an example of a system that uses approaches based on Machine Learning.

Despite the efficiency of semi-automatic ontology building techniques, ontologies used in this thesis were built manually since semi-automatic ontology generation is out of the scope of this thesis.

3.6 Existing Ontologies on the Web

Existing ontologies can be found either in ontology libraries, specialized search engines or portals. There are many ontology libraries on the Web. For example, DARPA[11] (DAML Ontology library) provides many ontologies, written in DAML and ranging from medical research to business. Stanford University provides an ontology library called Protégé Ontology library[12] containing ontologies developed using Protégé editor. OntoSelect[13] is an ontology repository that harvests ontologies from the Web. It supports ontology search and enables users to explore ontologies according to the representation format (RDFS, OWL, DAML), the size of the ontology (number of classes and properties), and the human languages used (English, German, etc). In addition, ontologies can be found using specialized search engines such as Swoogle[14] and OntoSearch[15].

3.7 Ontologies in Education

Listing all possible categories of ontologies that are used in all educational disciplines can be very difficult if we consider that any ontology used to support learning can be called an educational ontology (Devedzic, 2006). However, several general categories of educational ontologies can be identified from previous literature reviews (Devedzic, 2006; Aroyo and Dicheva, 2004a; Aroyo and Dicheva, 2004b; Mizoguchi,

[11] http://www.daml.org/ontologies/ [last accessed 15/11/2008].

[12] http://protege.cim3.net/cgi-bin/wiki.pl?ProtegeOntologiesLibrary [last accessed 15/11/2008].

[13] http://olp.dfki.de/OntoSelect/index.php?mode=select [15/11/2008].

[14] http://swoogle.umbc.edu/ [last accessed 15/11/2008].

[15] http://www.ontosearch.org/ [last accessed 15/11/2008].

2001; Chen et al., 1998). What follows briefly introduces these categories and then specifies the types of ontologies used in this thesis.

- **Domain ontology**: This ontology describes the content of a learning material in terms of concepts of the subject domain and their relationships. In addition, they can be used to define the learning paths through the topics defined in the domain ontology. These learning paths are represented as a set of semantic and pedagogical relationships among the domain concepts reflecting an instructional approach to learning/teaching from the instructor's point of view.

- **Task ontology**: This ontology defines the semantic features of problem solving such as the problem types, structures, activities and steps. For instance, it may include concepts like problem, scenario, question, answer, explanation and so on.

- **Teaching strategy ontology**: This ontology is used to model the teaching experience by defining the instructions and principles underlying pedagogical actions and behaviours. For example, it may define the sequence of actions to be taken when the learner makes a mistake, or it may specify behaviours to encourage learners to explore alternative solutions.

- **Learner model ontology**: this ontology is used to build learner models. It may include concepts to represent the learner's personal data, preferences, performance, learning styles and any other details.

- **Interface ontology**: this ontology specifies the educational system's adaptive techniques and behaviours at the user-interface level. For example, it can specify how the interface of the learning material can be adapted to different display devices such as laptops and mobile phones.

- **Communication ontology**: Educational servers and pedagogical agents communicate over the networks by exchanging messages. This ontology defines the semantics of the message content languages (i.e. the vocabulary of the terms used in the messages).

- **Educational Web-service ontology**: this ontology is used to enable interoperation of educational Web services. This ontology should provide means for creating machine-readable descriptions of service such as how they run or coordinate.

A Semantic Web-based learning environment may use one or more ontologies from the above categories based on its educational and functional requirements. The approach adapted in this thesis is based on two types of ontologies: domain and learner model ontologies. Domain ontologies are required to represent the domain vocabularies and relationships between domain instances which will be used for the annotation process. A learner model ontology was developed to model the learner profile and browsing activities which will be used for the co-browsing approach.

3.8 Ontology Applications in Education

Ontologies have been used successfully in many educational Semantic Web applications. Although it is difficult to list all the educational applications that use ontologies, this section highlights some of these applications according to their area of research. For instance, in the area of learning objects, Jovanovic et al. (2006) proposed Tangram, an integrated learning environment for the domain of Intelligent Information Systems. It enables a semi-automatic annotation for learning objects' components through content-mining algorithms and heuristics. The system provides the learner with a personalised aggregated learning content, based essentially on learning object metadata (subject, hierarchical relations of the domain concepts, instructional paths, ordering relations) and the learner model (learning history, preferences, learning style).

In the area of metadata, Edutella is a peer-to-peer (P2P) network for exchanging information about learning objects (Brase and Painter, 2004). The learning objects are classified using domain specific ontologies and annotated using a subset of Dublin core and LOM metadata.

In the area of e-learning, LOCO-Analyst is a tool that aims to help instructors rethink the quality of the learning content and learning design of the courses they teach (Jovanovic et al., 2007). It provides instructors with feedback about the activities the learners performed and/or participated in during the learning process. It also informs instructors about the usage of the learning content they had prepared as well as the interactions among members of the online learning community.

In the area of educational Web portals, Woukeu et al. (2003) developed 'Ontoportal', an ontological hypertext framework for building educational Web portals based on simple domain ontologies. The ontological Web portal contains links to educational resources that are semantically interconnected.

3.9 Using Ontologies to Measure Semantic Relatedness

Semantic relatedness refers to human judgments of the degree to which a given pair of concepts is related. Measures of relatedness are automatic techniques that attempt to imitate human judgments of relatedness (Pedersen et al., 2007). Some opinions distinguish between semantic similarity and semantic relatedness because the relatedness measure considers different kinds of relations (subsumption (is-a), meronymy (part-of) or any other domain specific relations) while the similarity measure uses only subsumption relations (Pedersen et al., 2004). However, much of the literature uses these terms interchangeably, along with terms like semantic distance. In essence, semantic similarity, semantic distance, and semantic relatedness all mean, "How much does term A have to do with term B?".

Measures of similarity are widely used in search engines, Web mining, natural language processing and information retrieval. A number of similarity measures have been defined to compute similarity between concepts. Examples of such measures are the cosine similarity measure (Manning et al., 2008), Dice's coefficient (Rijsbergen, 1979) and Jaccard's index (Hamers et al., 1989). The cosine similarity measure is the most widely used measure which has been applied to content matching techniques such as document matching, document clustering, and ontology mapping. However, the former measures may not always determine the right matches when there is no direct overlap in the exact concepts that represent the semantics. This is because concepts are treated as self-contained units whereas the relationships between concepts are usually ignored when similarity is measured (Thiagarajan et al., 2008).

Ontologies are considered intuitive ways of organising concepts according to their semantic relatedness. These ontologies capture semantic relationships between concepts or vocabulary used in a particular domain and can potentially be used to discover inherent relationships between descriptions of entities. They can be used to group together closer related concepts and to space more widely apart the more distantly related ones. A lot of semantic relatedness measures have been proposed for use within an ontology. The majority of these measures have been based on two underlying approaches:

- **Distance based within an ontology**. The length of the shortest path between the two concepts measures the distance between them. The shorter the distance the more related the concepts are. Examples of measures that are based on the shortest path between concepts can be found in Rada et al. (1989) and Leacock and Chodorow (1998).

- **Information content based on a common parent between concepts**. Conceptual similarity between two concepts can be measured by the degree to which they share information. The more information they share then the more related they are. This shared information is contained in the most specific concept that subsumes these two concepts, the lowest common subsumer. Wu and Palmer (1994) and Resnik (1995) proposed sample measures based on the depth of the most specific common subsumer. However, this approach considers only taxonomic links between concepts while non-taxonomic links, whereas concepts may not share a common parent, are ignored.

This thesis uses a distance-based approach to measure semantic relatedness between users' browsing activities. Accordingly, users with related interests on the Web can be grouped in collaborative browsing sessions. The proposed collaborative browsing approach will be discussed in detail in chapter 5. For a detailed comparative study of ontology-based semantic relatedness measures the reader is referred to (Blanchard et al., 2005).

3.10 Knowledge Organization Systems

The use of semantic Web technologies in e-learning imposes additional requirements that need to be addressed by standards and specifications. Standardisation in the area of educational technologies is important to enable interoperability and reusability across different e-learning applications. However, interoperability among different applications requires an appropriate solution to the problem of disparate and heterogeneous metadata descriptions and schemas across domains (Stojanovic, 2001).

Currently, there are several types of standards in learning technologies such as Dublin Core, IEEE LOM (IEEE LOM, 2002), and ADL SCORM (ADL SCORM, 2004). However, current standards are not tailored for the Semantic Web. Thus, further efforts are needed in order to adapt them for use in Semantic Web applications (Devedzic, 2006). The W3C[16] (World Wide Web Consortium) made the first steps in this direction with the SKOS[17] (Simple Knowledge Organisation System) specification. The SKOS data model provides a standard, low-cost migration path for porting existing knowledge organization systems to the Semantic Web. It also provides a light-weight, intuitive language for developing and sharing the basic structure and content of concept schemes such as thesauri, classification schemes, subject heading lists, taxonomies, folksonomies, and other types of controlled vocabulary. SKOS is built upon RDF and RDFS, and its main objective is to enable easy publication of controlled structured vocabularies for the Semantic Web. It may be used on its own, or in combination with formal knowledge representation languages such as the Web Ontology language (OWL). The SKOS consists of three RDF vocabularies that are still under active development (Miles and Brickley, 2005):

- **SKOS Core**, a set of RDFS classes and RDF properties that can be used to express the content and structure of concept schemes (taxonomies, terminologies, etc) in a machine-understandable way. Table 3.1 shows some of the classes and properties defined in the SKOS Core vocabulary. The

[16] http://www.w3.org/ [last accessed 15/11/2008].

[17] http://www.w3.org/2004/02/skos/ [last accessed 15/11/2008]

namespace skos: is defined as xmlns:skos = "http://www.w3.org/2008/05/skos".

- **SKOS Mapping**, RDF vocabulary for describing mapping between different ontologies.
- **SKOS Extension**, containing extensions to the SKOS Core, it is useful for specialized applications.

In this thesis, The SKOS Core vocabulary was used to develop the vocabulary for the domain ontologies.

Term	**Explanation**
skos:Concept	An abstract idea or notion; a unit of thought
skos:ConceptScheme	A set of concepts, optionally including statements about semantic relationships between those concepts (e.g., a thesaurus).
skos:Collection	A meaningful collection of concepts.
skos:prefLabel	The preferred lexical label for a resource, in a given language.
skos:altLabel	An alternative lexical label for a resource.
skos:hiddenLabel	A lexical label for a resource that should be hidden when generating visual displays of the resource, but should still be accessible for text search operations.
skos:broader	A concept that is more general in meaning.
skos:narrower	A concept that is more specific in meaning.
skos:hasTopConcept	A top level concept in the concept scheme.
skos:inScheme	A concept scheme in which the concept is included.
skos:isSubjectOf	A resource for which the concept is a subject.
Skos:related	A concept with which there is an associative semantic relationship.

Table 3.1: Some terms from the SKOS Core vocabulary (Devedzic, 2006).

3.11 Semantic Annotation

3.11.1 Overview

Handschub and Staab (2003) defined annotation as "... *a set of instantiations attached to an HTML document*". Semantic annotation means assigning a Web resource with some machine processable meanings taken from Ontologies (Zhihong and Mingtian, 2003). Semantic annotation can also be defined as "... *a specific metadata generation and usage schema, aiming to enable new information access methods and to extend* *the existing ones*" (Kiryakov et al., 2003). In addition, Ding (2005) defined semantic annotation as "*...a process ... to label Web page content explicitly, formally, and unambiguously using ontologies*". Semantic annotation can also be named "semantic markup" (Devedzic, 2006), or ontology-based annotation (Handschuh and Staab, 2003).

Documents can be annotated either manually or semi-automatically. Manual annotation requires a user to annotate the document content using predefined ontologies. It is time consuming and error-prone. The use of human annotators usually produces errors due to factors such as unfamiliarity with the domain, limited training, personal motivation and complex schemas (Bayerl et al., 2003). Another problem with manual annotation is the huge amount of existing documents on the Web that need to be annotated before being usable on the Semantic Web. An example of manual annotation is the Onto-Mat-Annotizer tool (Handschuh et al., 2001).

To overcome the problems of the manual annotation, semi-automatic annotation of documents has been developed. Semi-automatic techniques are required because, as yet, there are no automatic techniques that can identify and classify all document content with complete accuracy. Therefore, all the existing annotation systems, whatever their range of automation, require human intervention at some point in the annotation process (Handschuh et al., 2002). An example of semi-automatic annotation techniques is SemTag (Dill et al., 2003).

There are several ways of annotating Web resources. One approach is to use annotation tools to mark up the downloaded Web pages manually. This approach is called internal annotation because annotations are contained inside the Web pages. However, Web pages stored in different Web servers normally do not have write access, and therefore annotations cannot be added internally. A more sophisticated approach is called external annotation in which annotations are stored separately in another document and loaded in the browser along with the Web page. The third approach is the collaborative annotation of Web resources through Wiki sites that let users insert and share their annotations and comments (Handschuh et al., 2002).

In addition, documents can be annotated on two levels:

- **Document-level annotation**: in which annotations are attached to the whole document.

- **Content-level annotation**: in which annotations are attached to certain parts of the document content (e.g. words, sentences, paragraphs). Such annotations come in the form of highlighted text, new elements inserted in the document and hyperlinks (Devedzic, 2006).

In this thesis, an external annotation approach is used to annotate Web pages on the content level. Content annotation can serve as a way to interpret Web pages being browsed by annotating any included domain terms with further explanations or definitions. Since Web pages cannot be altered without having write access, annotations will be stored separately in an external repository and will be attached to the Web page content upon the user's request.

3.11.2 Platform Classification

Reeve and Han (2005) classified annotation platforms based on the type of annotation method used into three types: pattern-based, machine learning and multi-strategy based.

Most pattern-discovery methods follow the method outlined by Brin (1998). In these platforms, an initial set of entities and rules is defined and then the content is scanned to find the matching patterns in which the entities exist. This process continues recursively until no more entities are discovered, or the user stops the process. Annotations can also be generated by using manual rules to find entities in the text.

Machine based annotation platforms use two methods: probability and induction. Probabilistic platforms use statistical models to predict the locations of entities within the text. Induction tools use either linguistic or structural analysis to perform wrapper induction.

Multi-strategy-based platforms combine both pattern-based and machine learning-based methods. There are no existing platforms that implement both strategies, although it has been used in some systems for ontology extraction such as On-To-Knowledge (Kietz and Volz, 2000).

3.11.3 Sample Annotation Frameworks

In a developing field such as the Semantic Web, it is difficult to complete a comprehensive survey of all current semantic annotation platforms due to rapid changes in this area. This section will attempt to summarize a set of well-known semantic annotation tools.

Ont-O-Mat (Handschuh et al., 2002) is the implementation of the S-CREAM, a framework that proposes both manual and semi-automatic annotation of Web pages. Ont-O-Mat is a user-friendly annotation tool for Web pages. It includes an ontology browser for the exploration of the ontology instances and a Web browser that displays the Web pages. Ont-O-Mat allows the annotator to mark-up relevant parts of the Web page manually by drag and drop interactions. It also provides semi-automatic annotation through the use of information extraction rules. Ont-O-Mat stores the annotations in the Web pages. It also provides crawlers that can search the Web for annotated Web pages to add to its internal knowledge base.

MnM (Vargas-Vera et al., 2002) is similar to Ont-O-Mat. It supports both automated and semi-automated annotation. MnM integrates a Web browser with an ontology editor. It also provides open APIs, such as OKBC[18], to link to ontology servers and for integrating information extraction tools. Unlike Ont-O-Mat, MnM can handle multiple ontologies at the same time. It stores annotations either locally in the Web pages or separately in a knowledge base.

The **KIM** platform (Popov et al., 2003) consists of a formal KIM ontology and a KIM knowledge base. The KIM ontology is an RDF(S) ontology that defines the entities and relations of interest. The KIM knowledge base stores the entity description information for annotation purposes. KIM employs an information extraction technique, which is based on GATE (General Architecture of Text Engineering) (Cunningham et al., 2002) to extract, index, and annotate ontology instances that are included in the Web page text. Annotations are stored inside the Web pages. KIM also provides a browser plug-in so that people can view the annotated information

[18] Open Knowledge Base Connectivity, http://www.ai.sri.com/~okbc/ [last accessed 20/11/2008]

graphically through different highlighted colours in regular Web browsers such as Microsoft's Internet Explorer.

Annotea (Koivunen, 2005; Kahan et al., 2001) is a text annotation tool that supports collaborative annotation of documents. The user can load the annotations attached to any document from a selected annotation server and see what his peer group thinks. Metadata in Annotea are represented in RDF. The types of documents that can be annotated in Annotea are limited to HTML and XML formats. The generated annotations can be stored either in the user's machine or in RDF servers.

3.11.4 On-Demand Annotation

In a comprehensive survey of semantic annotation tools, Uren et al. (2005) devised this category to list systems that are not strictly annotation tools. These tools produce annotation-like services on demand for users browsing un-annotated resources. The goal of such tools is to annotate resources that are either hard to annotate, such as external Web pages, documents which change rapidly, or those which might be annotated but with an unsuitable ontology. What follows reviews some of the existing on-demand annotation tools.

Magpie (Dzbor et al., 2004) is a tool operating from within a Web browser to allow users to associate meta-information with Web pages. It annotates Web pages by highlighting keywords related to the ontology of the user's choice. Then, appropriate Web services and functionalities can be associated to the highlighted keywords. The goals for Magpie were to:

1. Support information gathering by highlighting and annotating entities that are related to the domain ontologies
2. Enable users to adapt the annotation service based on their needs by allowing them to select domain ontologies to be used for annotation.

COHSE (Conceptual Open Hypermedia Service) (Bechhofer et al., 2008) is a framework supporting dynamic linking of arbitrary Web pages. It annotates Web pages by adding links to keywords that are related to predefined ontologies. The links are added either by a proxy server or by an augmented Mozilla browser. The design goals for COHSE were to separate Web links from the Web pages and to make these links conceptual (i.e. potentially generated from an ontology). COHSE focuses on hypertext authoring and how to make links ontology based while giving less attention to metadata processing and reasoning.

SemWeb (Sah et al., 2008) is a tool extending the Mozilla Web browser to provide adaptive annotation of Web pages being browsed. SemWeb employs a user-model ontology to capture browsing behaviours and then annotate the page content based of the deduced browsing interests. It investigates Semantic Web technologies and user-modelling approaches to better support the user's browsing using enriched semantic links.

The **Thresher** system (Hogue and Karger, 2005) allows users to teach their browser how to extract semantic Web content from HTML documents on the Web. Users can specify examples of semantic content by highlighting them in a Web browser and describing their meaning. The system then generates a general pattern, or wrapper, for the content and allows the user to bind RDF classes and predicates to the notes of these wrappers.

Since the focus in this thesis is to support learning from hypertext learning resources, an on-demand annotation service is used to enable learners in open communities to semantically annotate existing Web pages. This view emerges for several reasons:

1. Plenty of existing learning resources on the Web are in hypertext format, and most of them are not annotated. On demand annotation offers a "real-time" solution to annotate such resources that are normally difficult to annotate.

2. The annotation process, by definition, aims to associate extra information to documents. Thus, an idea adopted in this thesis is to use semantic annotation as a technology to support information gathering and instructional guidance, a development that can enhance learning through hypertext resources.

3. Unlike other annotation frameworks that are primarily designed for authors or specialists, on demand annotation allows any user, regardless of his/her experience, to annotate Web pages.

3.11.5 Semantic Annotation Tools for E-Learning

If general-purpose annotation tools are used to semantically annotate learning materials, human annotators are faced with specific requirements that the e-learning context brings to the annotation process (Devedzic, 2006). Azouaou et al. (2004) identified some guidelines for developing semantic annotation tools that commit to the requirements of learning applications. They first identified three main players in the annotation activity, which are: the annotator, the user of the annotation and the question of whether the annotation is, or is not, semantic. Then, based on the previous identification, they listed the requirements for e–learning annotation tools, which include:

1. **Usefulness of annotation for the teaching/learning context**: the semantic annotation should take into consideration the requirements of specific domains, the teaching/learning objectives, activities, and the users of the annotation (humans or software agents).

2. **Shareability**: an e-learning annotation framework should support communication of learners and teachers through annotation. Annotations should be both accessible and comprehensible to other interested users and applications.

3. **Usability of annotation**: manual annotation should not disturb teaching/learning activities, and the annotators should be put in their usual teaching/learning context while annotating.

Azouaou et al. (2004) have evaluated a number of annotation tools with respect to the above requirements. Among the evaluated tools, two were dedicated to annotating learning material, namely, MemoNote (Azouaou and Desmoulins, 2006a; Azouaou and Desmoulins, 2006b) and AnnForum (Azouaou et al., 2004). Finally, they have come up with the following results:

1. Many aspects of usefulness of annotation for the teaching/learning context are generally not supported by such tools, but still can be reached with them.

2. Manual semantic annotation facilities that these tools provide do disturb teaching/learning activities, thus they are not perfect from the perspective of usability.

3. Only some general-purpose annotation tools enable (but only to an extent) automatic and semi-automatic semantic annotation.

3.11.6 Annotation Goals in Education

Annotation in a learning context should have four goals (Azouaou and Desmoulins, 2005):

1. Classification of learning resources, which means organizing resources into a hierarchy based on the structure of the domain of learning.

2. Adding information, which means attaching resources with additional comments or complementary information.

3. Planning and scheduling of the learning activities and tasks.

4. Correlating, which means linking resources and/or activities based on the requirements of the learning domain and/or task.

This thesis will focus on the use of semantic annotation to classify and add information to existing Web resources for the purpose of supporting instructional learning and resource interpretation.

3.12 Concluding Remarks

From the previous overview of the different aspects of the process of semantic annotation, several points can be highlighted:

- Most previously mentioned tools rely on either human manual annotations or (semi)-automatic annotation that uses Information Extraction (IE) and Machine Learning (ML) techniques to extract valuable information from a Web resource.

- There are few semantic annotation tools dedicated to the e-learning domain. This might be attributed to the fact that the Semantic Web community has been more interested in building Semantic Web technologies to serve the needs of large industries/ organizations and/or research centres, rather than the needs of education.

- Most existing annotation tools have focused either on content providers (in the form of authoring tools) or on technically proficient end-users who know HTML and RDF (Hogue and Karger, 2005). This thesis explores the annotation process from the learner's perspective through an on-demand annotation service. It addresses the usefulness of annotation for the learning context and how it can support contextual interpretation, instructional learning and information gathering.

Therefore, the research question addressed here is how to enable the learner to associate semantics with any Web page that may not be annotated. The subsequent question is how to employ these semantics to support the interpretation of the page and provide guided learning. The vision of this thesis is to develop an on-demand annotation service to be used by learners in an educational context. The purpose of the proposed service is to support the interpretation of the existing hypertext learning resources on the Web. It uses content-level annotation to associate the Web page being browsed with additional layers of complementary information and instructional guidance. These layers are constructed on the fly from a set of educational ontologies and an external knowledge base. Such layers can increase learners' awareness and engagement in the learning process. The discussion of the proposed annotation service along with the design decisions will be the theme of chapter 5.

Chapter 4
Knowledge Puzzle, A tool for Knowledge Construction from Hypertext

4.1 Introduction

How to help learners construct knowledge and plan a navigation process on the Web are important issues in Web-based learning/education. This chapter demonstrates the Knowledge Puzzle, a tool for knowledge construction from the Web. It offers a constructivist approach to SDL by helping learners to:

1. Plan their navigation path with the least cognitive effort.
2. Personalize the information structure on the Web to accommodate the learner's self-constructed knowledge.
3. Publish the knowledge gained from the Web so that it can be reviewed easily.

The main contribution of the proposed tool to Web-based learning is the personalisation of information structure on the Web to accommodate the interlinked knowledge structure in the learner's mind. Self-directed learners will be able to adapt the path of instruction on the Web to their way of thinking, regardless of how the Web content is delivered. The way to achieve that is to provide the learner with a meta-cognitive tool that enables him to bring his knowledge to the surface and visualize what he has in mind. Once we get the learner's viewpoint externalised, it will be converted to a hypermedia layer that will be attached to and laid over the Web pages visited by the learner. The attached layer adapts the views of Web pages to the learner's information needs by associating information pieces that are not already linked in hyperspace and attaching the learner's notes to the page content.

This chapter starts by defining the requirements that should be fulfilled for effective knowledge construction from hypertext. It then considers the navigation process in hyperspace and explains the cognitive activities involved in it. The Knowledge Puzzle tool and its functionalities are demonstrated in the following sections. Afterwards, a thorough discussion is given to demonstrate how the requirements are met and how SDL is promoted through the different stages of the learning process. Finally, some implementation details are highlighted.

4.2 Requirements

This section outlines the requirements for the design of the proposed tool. These requirements are mapped, whenever possible, to the related discussion on SDL from hypertext in chapter 2.

R1: Reduce the cognitive load associated with Web navigation

To control and regulate the cognitive effort associated with Web navigation, the vision of this thesis is to stimulate learners to practice meta-cognitive skills (planning, monitoring, evaluating and revising) outlined in section 2.6. Mastering these skills leads to promote learning from hypertext to be a process of active construction, based on the theory of cognitive constructivism discussed in section 2.5.2. This can accordingly increase the learner's awareness of his/her cognitive activity and reduce the cognitive load associated with the Web navigation process.

R2: Adapting information on the Web to the learners' individual needs

The gap between knowledge in mind and information on the Web requires learners to make an excessive effort to monitor their learning progress. This thesis claims that the SDL from hypertext environment requires information space on the Web to be adapted to the progression of the knowledge construction process. This requirement can be divided into a set of sub-requirements as follows:

i. The framework should enable learners to annotate and add hyperlinks to the Web pages being browsed in order to match their individual information needs, without the need to have actual ownership of the Web pages.

ii. The framework should process any Web page without any prior design settings. The learners should be able to interlink and annotate the content of any Web page regardless of its structure.

iii. The adaptation provided by the framework should be entirely user-driven. It should not rely on any adaptive hypermedia technique such as link or content adaptation that might be provided by the content designer. This thesis claims that adaptive hypermedia is not appropriate for SDL from the Web for the reasons discussed in section 2.8.1.

iv. The framework should separate the annotations from the documents. Annotations and links should be layered over the Web page. This will enable learners to have different viewpoints of the same resources based on their different needs.

R3: Reduce dependence on external tools

Many research efforts have proposed using meta-cognitive tools to support knowledge construction (Kashihara and Hasegawa, 2004; Lee and Baylor, 2006; Khamidoullina et al., 2001). In spite of the significance of these tools to enhance cognitive activities, the excessive use of such tools while navigating the Web can disturb the learners' attention due to the additional effort required to manage the work on these tools, as was discussed in section 2.8.4. Therefore, it is necessary to reduce the dependence on these tools as much as possible while the navigation process is in progress.

4.3 Navigational Learning Process in Hyperspace

A prerequisite for designing an effective constructivist approach for SDL is to understand how learners construct knowledge from hypertext. In order to achieve that, it is essential to study how the human brain processes and retrieves information from the Web. Using the experience from previous research (Kashihara and Hasegawa, 2004; Mitsuhara et al., 2008; Tergan and Keller, 2005), this section explains the navigational learning process in hyperspace and highlights its relationship to the process of knowledge construction in the human mind. Figure 4.1 depicts how the process is carried out in both hyperspace and in the learner's mind. In hyperspace, learners generally start navigating with a specific learning goal. For example, they may start navigation for the goal of finding a definition for an unknown term or a description for a particular topic. They move between the various pages until they find the information that fulfils the learning goal. In Figure 4.1, information that fulfils the learner's needs has been found in three pages that are not directly linked in hyperspace (components A, B and C). Subsequently, the learner creates in his/her mind relationships between these distributed information pieces. Each relationship represents the goal of the navigation process from one piece to another. For example, the learner may decide that information in component B supplements what he/she has learned in component A, and that component C elaborates the description given in component B. However, these relationships only exist in the learner's mind since there are no direct hyperlinks between the information pieces in hyperspace.

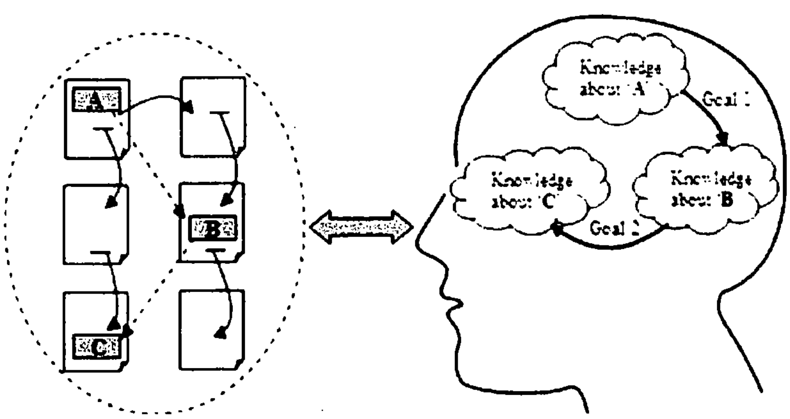

Figure 4.1: Knowledge construction process during Web navigational learning

One should note here that the learning goal arising from visiting a page is not always fulfilled in the immediately following page. Additionally, while searching for the fulfilment of the retained goal, it is possible for other navigation goals to arise. It is possible also that the learning goal may have several terminal pages with one starting page. For example, a solution of a particular problem could be found in the relationships between multiple pages. All such circumstances require the learner to make an excessive cognitive effort to retain the knowledge components and the relationships among them in his/her mind. If we can convert the links that learners mentally create between information components while navigating the Web into real informative hyperlinks and annotations in hyperspace, we can considerably reduce the cognitive overhead associated with the navigational learning process. The Knowledge Puzzle tool was developed to achieve such purpose. The tool and its functionalities are explained in the following sections.

4.4 The Knowledge Puzzle Tool

In order to help learners bring their thinking to the surface and visualize the knowledge that they gain from the Web, the Knowledge Puzzle tool shown in Figure 4.2 was developed. It is divided into two windows: a standard browsing window (to the left) and a planning space (to the right).

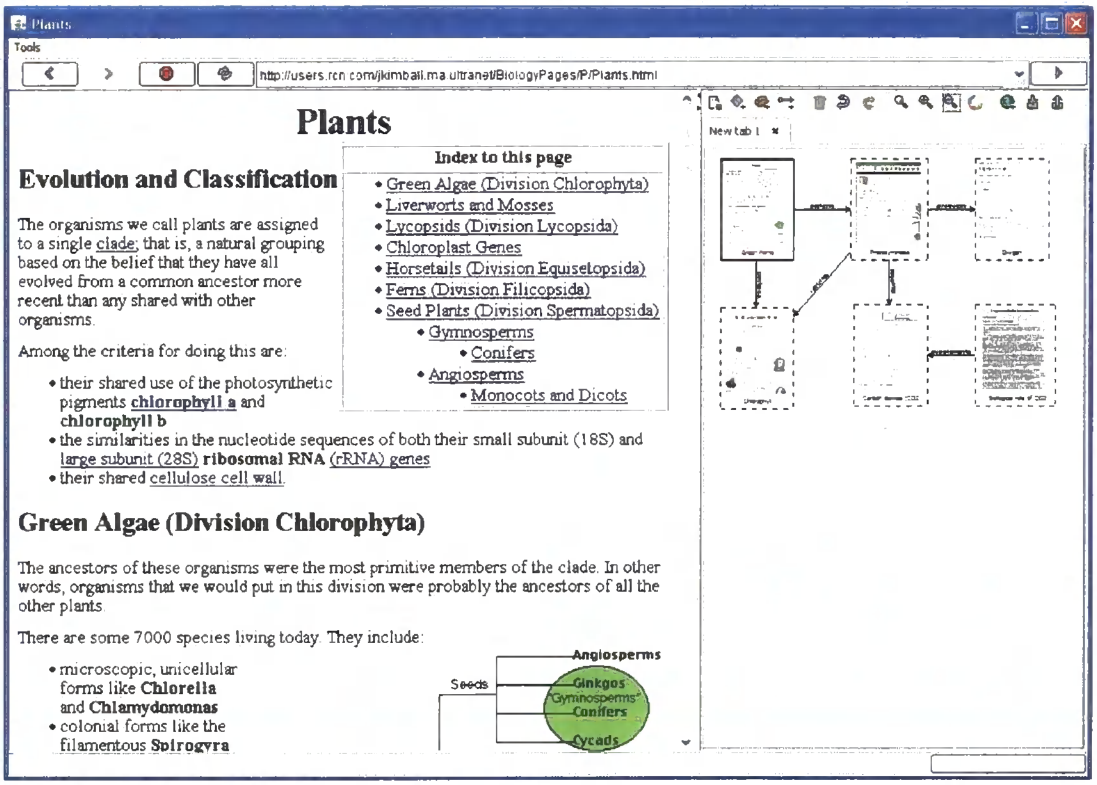

Figure 4.2: The knowledge Puzzle tool

The planning space allows learners to build a sequence of Web pages that fulfils specific learning goals. The tool emulates the navigational learning process, explained in the previous section: when the learner needs to set up a navigation goal in visiting a page, he/she can add a graph node representing the page where the goal arises. The graph node could denote either a whole Web page, or a selected portion of a page. Similarly, when the learner finds the information that fulfils the learning goal, he/she can add another node denoting the terminal page, or a particular portion of it, that fulfils the learning goal. The learner can then annotate each node with the main concept learned from the corresponding Webpage. Subsequently, the learner links the source and terminal nodes and annotates the connecting link to describe the relationship between them.

Each graph node is represented as a thumbnail for the source Web page. The use of thumbnails aids the human memory when previously visited pages need to be retrieved. In addition, each node maintains the URL of the corresponding page. Clicking on the node causes the corresponding page to open in the browsing area and the part of interest to be highlighted in a different colour. This direct linking between the components on the planning space and the components in hyperspace enables fast and easy access to information within the source pages on the Web.

The learners can look at the constructed graph whenever they wish during navigation and can also directly manipulate it. Each manipulation is done by means of mouse clicking/dragging parts of the knowledge map. There are four basic manipulations: adding, deleting, editing and changing the navigation goal links between the starting and terminal pages. Learners can also take a note about the contents learned from any Web page. This is possible through the entry form shown in Figure 4.3, which is attached to each graph node. Later on, these added notes can be embedded into the Web pages so that the learner can preview information on the Web along with his/her own notes.

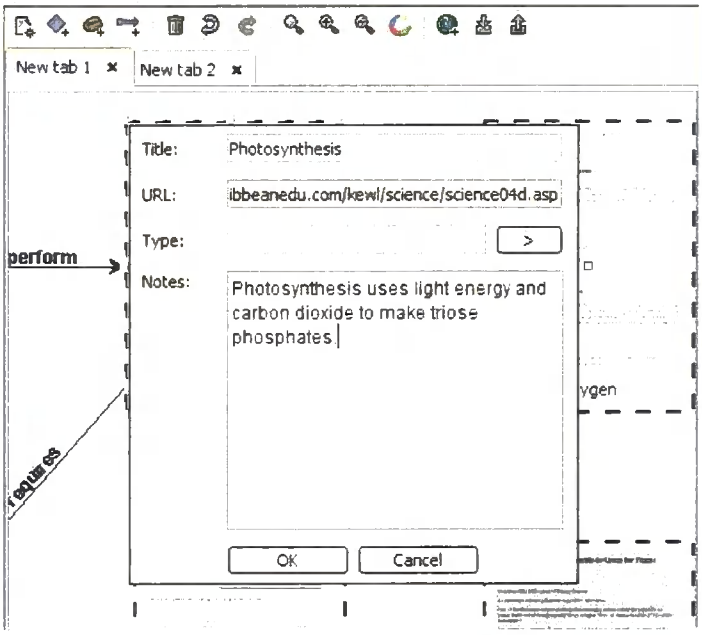

Figure 4.3: Note taking form

The main advantage of this tool over the traditional concept map and spatial tools is that it combines the benefits of both navigation path planning and concept mapping: it facilitates navigation path planning by offering the ability to connect information units on the Web and to directly access them by simple clicking on the corresponding graph nodes. This helps learners to consolidate the correct sequence of options and enables them to remake the path at any time. At the same time, it supports concept mapping by enabling learners to annotate the graph nodes with the key concepts gained while learning and to define the relationships among them. Accordingly, the constructed graph mirrors the knowledge map in the learner's mind and enables him to think reflectively about what he has gained from the Web exploration process.

Figure 4.4 shows an example of a knowledge graph constructed for the goal of understanding the photosynthesis process in plants. It shows the nodes denoting the Web pages visited and the primary exploration processes that have arisen. For example, the learner visited the pages labelled "Chlorophyll", "Carbon Dioxide (CO_2)" and "Oxygen" in order to explore the requirements and products of the photosynthesis process described in a previously visited Web page. The learner then visited the page "Biological role of CO_2" since it supplements the information found in the page "Carbon Dioxide". Each graph node is annotated by the learner with the main concept learned from the corresponding page. The learner also annotates the links between pages to describe the exploration goal from one page to another. For example, the link labelled "requires" denotes that the learner visited the terminal page for the purpose of understanding a requirement of the concept described in the previous page.

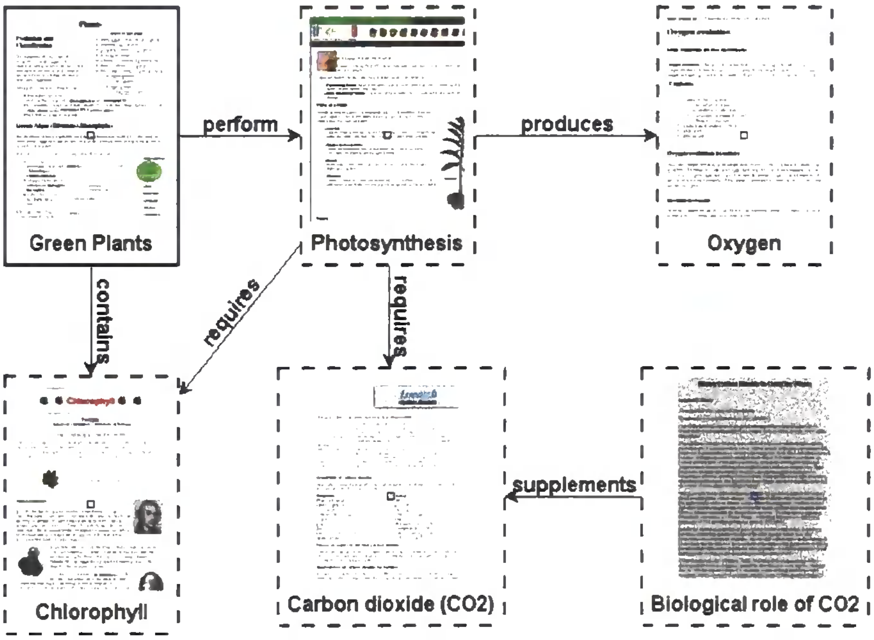

Figure 4.4: A sample knowledge graph

When the learner annotates a link connecting a source with a terminal node, he/she should describe both the relationship and its inverse. For example, if the link from "Photosynthesis" to "Oxygen" is labelled "produces", the inverse link should be labelled "is produced by" or something similar. While, for simplicity, the inverse links are not displayed in the knowledge graph, they will be used in the next stage when the knowledge graph is converted to a hypermedia layer.

4.5 Hypertext Layering

An emerging requirement for SDL from hypertext is to adapt the information structure on the Web to learners' individual needs. The Knowledge Puzzle tool automatically transforms the knowledge map, which was constructed by the learner on the planning space, to a layer of hyperlinks and annotations. The constructed layer is then attached over the visited Web pages. Thus, the learner will be able to see the knowledge components on the Web interlinked and structured exactly as he/she planned. The layering process is done as follows:

1. The links connecting any source node with terminal ones are converted to a menu of hyperlinks. The constructed menu is then embedded inside the Web page represented by the source node. The aim of this menu is to link information pieces in the source page with any related information pieces in terminal pages.

2. Similarly, menus are created in the terminal pages to include inverse hyperlinks that link information pieces in terminal pages with related information pieces in source pages.

Figure 4.5 shows an example for the conversion process. Figure 4.5.A shows two graph nodes from Figure 4.4 with the associated relationships. Figure 4.5.B shows the corresponding hypertext menus. Each menu contains hyperlinks representing the relationships associated with each node.

Since the pages on the Web are static and hence cannot be modified, the new structures will be embedded inside the local copies of Web pages after being loaded on the browser. Once any of the pages is loaded, the system automatically creates the corresponding menus based on the relationships defined on the planning space. The created menus are then embedded inside the loaded copy of the page by altering its HTML content.

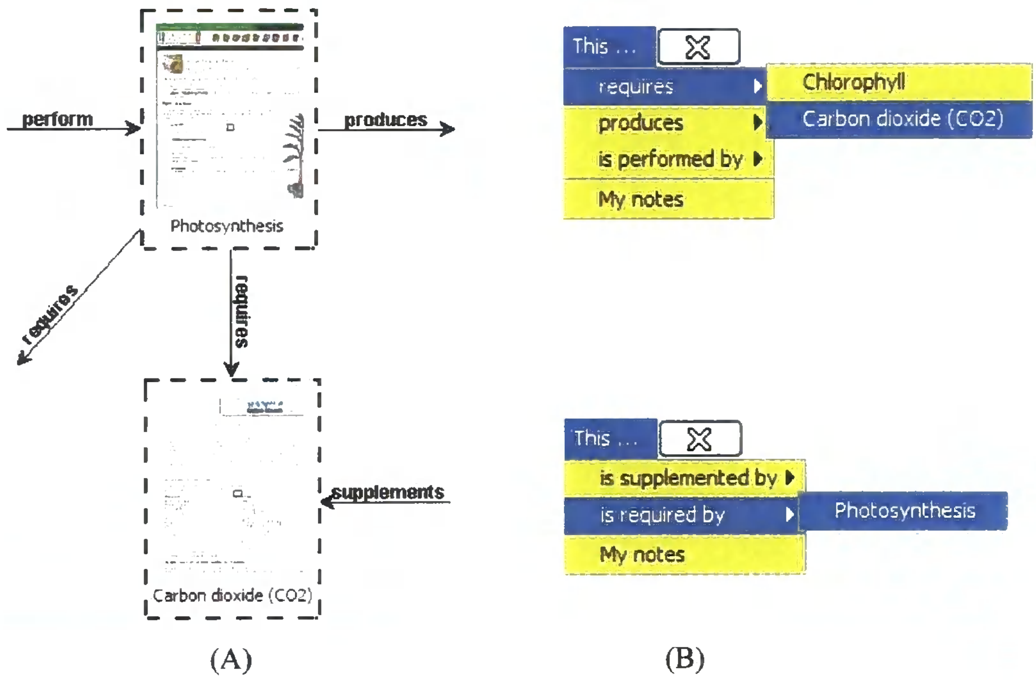

Figure 4.5: The conversion of the knowledge graph to hypertext components: (A) an excerpt of the knowledge graph, (B) The constructed hypertext menus.

Figure 4.6 shows how the Web pages, represented by the graph nodes shown in Figure 4.5, look after the constructed menus are attached. The piece of information that the learner is interested in is highlighted in red and a new hyperlink labelled "Click here" is attached to its end. Clicking on the link causes the menu to be displayed. The attached menus link Web pages and reveal how they are related from the learner's perspective. Thus, they enable seamless and direct movement between information pieces that make up the learner's knowledge. Note that the original pages on the Web do not include these changes and thus they are not directly linked in a way that matches the learner's viewpoint. Using the above hypertext layering over pages, there is no longer a need to own the pages in order to make the links from source to target pages. In addition, the labelling of the new associations enables the learners to recall why they navigated each Web page, how they processed its content and what was the goal fulfilled from each navigation process, thus providing support for learners to rebuild their cognitive models.

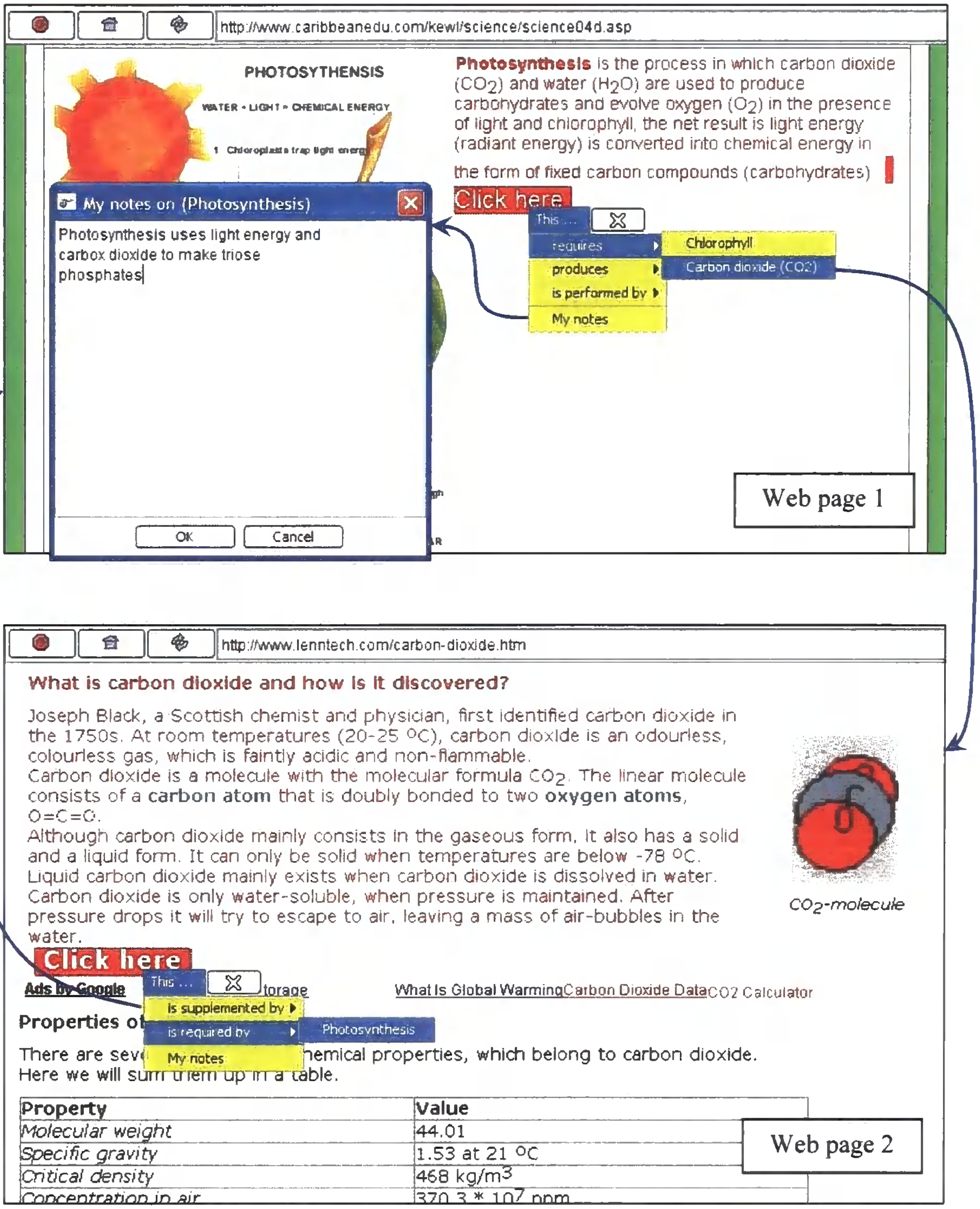

Figure 4.6: The Web pages after attaching the hypertext layer

Both menus show a link labelled "My notes". Clicking on this link will open a small window as shown in Figure 4.6. The window displays the notes and comments made by the learner about this information piece. This enables learners to preview their comments and notes side by side with the related Web content, something that cannot be achieved by traditional notebook and annotation tools that keep user-added notes separate from the Web content.

After altering the Web pages to reflect their information needs, learners can now hide the planning space and focus their attention on the Web navigation process, which becomes guided by the added hyperlinks and annotations. They need to recall the planning space only if they wish to add new relationships or modify the existing ones. Any changes made in the knowledge graph by the learner will be automatically applied in hyperspace. Therefore, the learner can manipulate the hypertext layer by adding, deleting and modifying components and links on the planning space.

4.6 Generation of Hypertext Knowledge

In addition to transforming the constructed knowledge to a hypertext layer, the tool generates an independent hypertext representing the whole constructed knowledge graph. The generated hypertext consists of a single page containing all the knowledge components interlinked as in the knowledge graph. Figure 4.7 shows an excerpt of the hypertext generated for the knowledge graph in Figure 4.4. Any referenced information pieces are extracted from the source pages and embedded in the generated hypertext (see elements A1, A2 and A3 in Figure 4.7). It also has a hierarchical structure where the terminal components become subsections of the source components. The relationships are revealed so that the reader can recall how and why he/she processed the information components (see elements B1 and B2 in Figure 4.7). Additionally, notes taken by the learner during the navigation process are attached to the corresponding components (see element C1 in Figure 4.7).

The generated hypertext promotes the knowledge review process since it gathers all the distributed information and user notes, and organises them in a single document based on the user-defined relationships. The learners can export the constructed knowledge to a local file so that it can be retrieved and reused later. Retrieving knowledge stored in such a file will not only regenerate the hypertext format, but will also reconstruct the complete knowledge graph on the planning space, a thing that allows learners to revise and modify the knowledge structure.

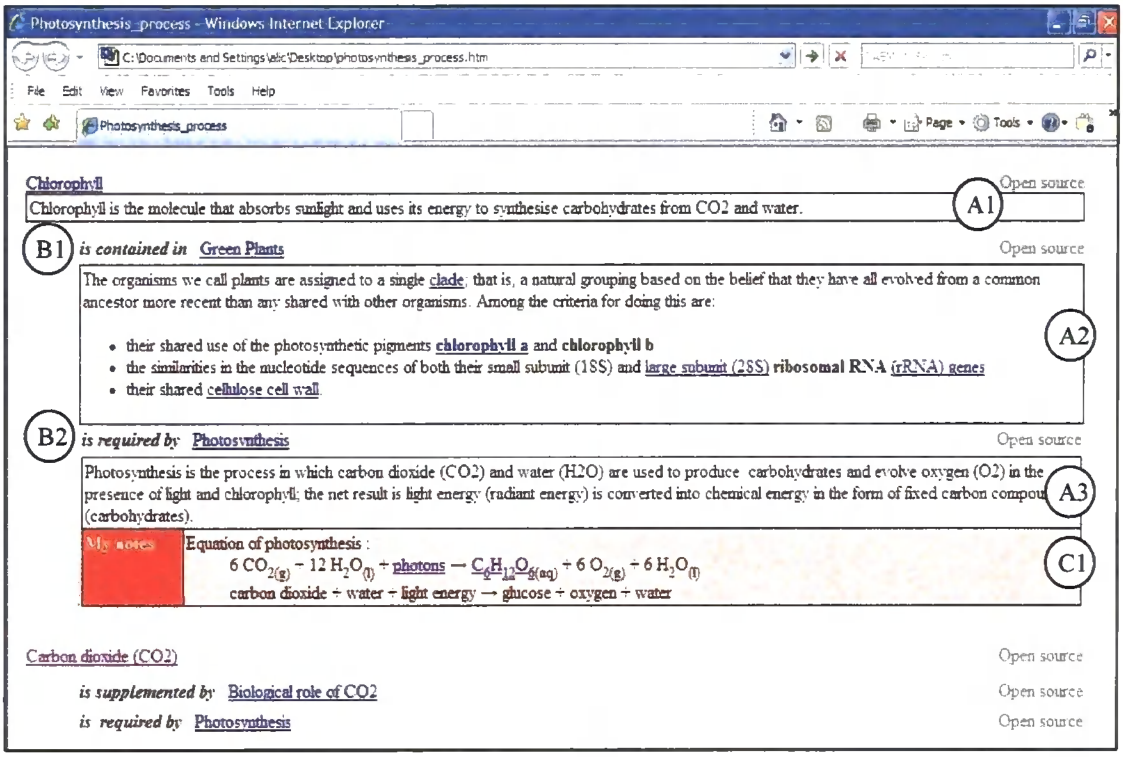

Figure 4.7: An excerpt of the hypertext generated for the graph in Figure 4.4

4.7 Discussion: the Use of Knowledge Puzzle Tool for SDL

After the functionality of the Knowledge Puzzle tool has been demonstrated in detail, the following section explains how the tool can be used to achieve an effective and scaffolded approach for SDL from hypertext. This section goes through the different stages of learning from hypertext and explains how the tool can resolve the problems associated with every stage. Furthermore, it discusses the educational benefits gained from using the tool in every stage and demonstrates how the requirements outlined in section 4.2 have been met. Figure 4.8 depicts an overview of the learning stages with hypertext and how the Knowledge Puzzle tool is utilized to facilitate learning at each stage.

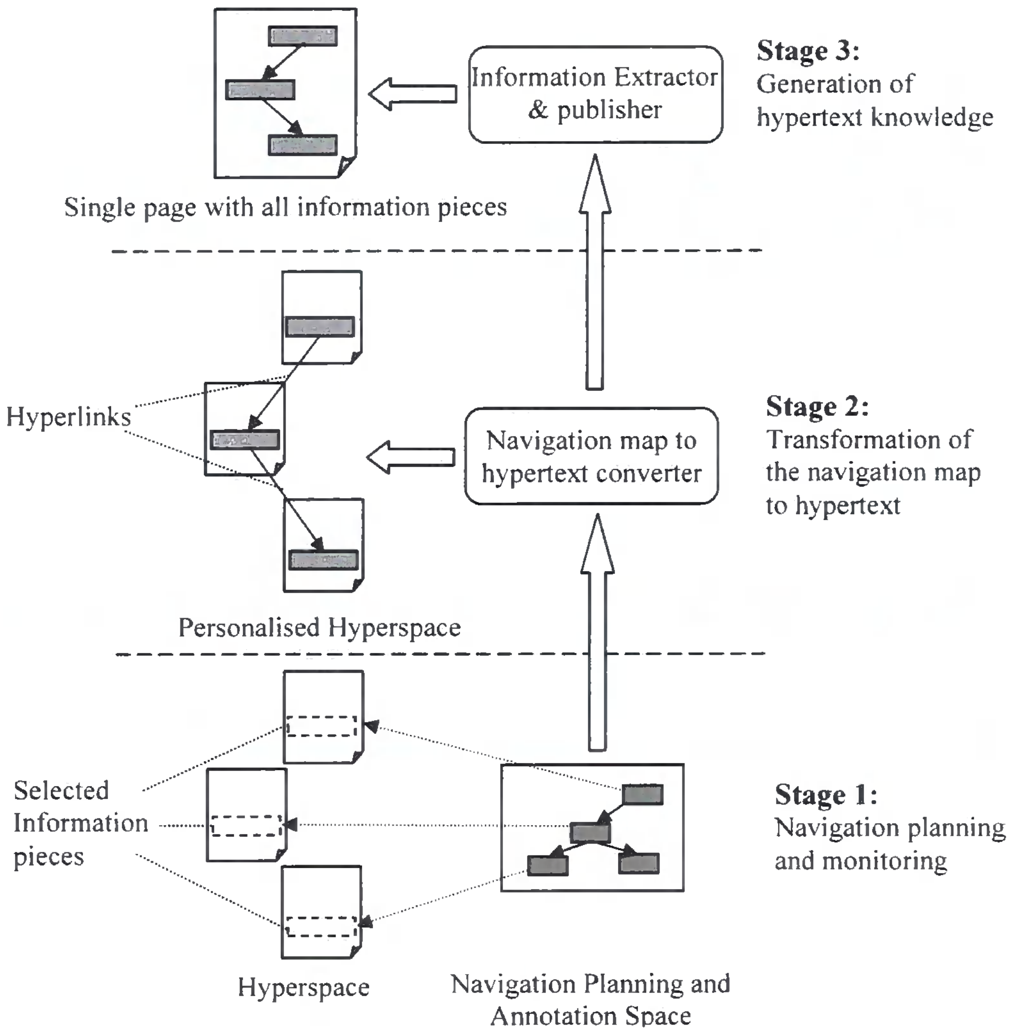

Figure 4.8: The use of Knowledge Puzzle tool for SDL from the Web

4.7.1 Stage 1: Navigation Planning and Monitoring

This stage involves learners when they start navigating the Web with the aim of exploring and gathering information that fulfils their needs. At this stage, learners need support to monitor and control the learning process. Thus, SDL at this stage is done using the two spaces provided by the Knowledge Puzzle tool. These are hyperspace for navigating the Web pages, and the space for planning the navigation path based on the learning goals. In the planning space, learners plan which page, or part of a page, to visit and the sequence of pages to be visited so that the learning goal can be achieved. Subsequently, information components on the Web can be previewed according to the planned path. The path planning and navigation are repeated during SDL.

From an educational perspective, helping learners build the optimum navigation path that fulfils their learning goals can release them from unnecessary browsing activities and protect them from getting lost in hyperspace. The separation between navigation path planning and the Web exploration process is crucial at this stage since it allows learners to become aware of monitoring their navigation process. Furthermore, learners can practice meta-cognitive skills by utilizing the benefits of concept mapping. Concept mapping promotes SDL by enabling learners to plan, monitor, evaluate and revise the information gained from the Web (Lee and Baylor, 2006). If the learners become able to externalise what they have in mind and put it in an understandable format, they will be released from the cognitive effort required to retain the acquired knowledge in mind (Kashihara and Hasegawa, 2003). This considerably fulfils requirement R1 in section 4.2.

4.7.2 Stage 2: Transformation of the Navigation Map to Hypertext

This stage involves learners when they start rethinking and reviewing the knowledge gained during the exploration process. Learners at this stage may need to revisit Web pages that they found useful during the Web exploration. They also need to recall why they visited these pages, which page contents are important, and how the contents of different pages are semantically related. In order to assist learners to achieve that, the contents of Web pages that make up the learner's knowledge will be annotated and interlinked on the Web in accordance with the graph that the learner constructed during the previous stage. The links connecting the sequence of Web pages on the planning space will be converted to real hyperlinks and annotations inside the Web pages as demonstrated in section 4.5. The ultimate goal is to adapt the Web structure to the learner's knowledge structure in order to achieve the desired paths.

From an educational point of view, converting the links that learners mentally create between information units while exploring the Web into real informative hyperlinks has many benefits:

- It helps learners orient themselves in hyperspace in such a way that facilitates rethinking the knowledge they gained without making excessive cognitive effort. Each learner will be able to navigate the Web from his own viewpoint by interlinking distant information pieces that may not be directly linked in hyperspace. This sort of user-driven adaptation of hypertext resources contributes towards the fulfilments of requirement R2 and all its sub-requirements explained in section 4.2.

- By moving from the planning space to hyperspace, learners are released from dependence on any planning or meta-cognitive tools, and thus they can focus their attention on the value of the information, which becomes explicitly interlinked and annotated according to their own needs. During stage 1, the planning space was necessary to promote self-study and meta-cognitive skills. However, when learners start the knowledge reflection and revision processes, the use of the planning space beside the normal browsing activities can place an additional burden on them. By transforming everything gathered or constructed on the planning space into hypertext components and attaching it to Web pages, the learners become no longer dependant on any external tools that may place additional cognitive loads on them, and this meets requirement R3 explained in section 4.2.

It should be noticed that what is offered here is different from link adaptation, the adaptive hypermedia technique. Link adaptation is based on manipulating links that already exist in Web pages in order to adapt the navigation path to the user's needs. However, it offers nothing if information pieces that make up the learner's knowledge are not already linked in hyperspace. The solution proposed here goes further by enabling the learners, rather than content authors, to add new hyperlinks and annotations to Web pages without the need for any underlying adaptation strategy.

4.7.3 Stage 3: Knowledge Generation in Hypertext Format

In order to help learners reflect on the knowledge gained, this stage enables learners to automatically transform the constructed knowledge, based on the planned navigation path, into a hypertext format as demonstrated in section 4.6. Such knowledge transformation aims to promote self-reflection on the constructed knowledge by bringing the whole structure and information into the learners' view, and thus enabling them to rapidly review their knowledge without the need to revisit the source pages.

Finally, it should be noticed that the above stages complement rather than substitute each other. Learners can move from one stage to another in accordance with the progress of the learning process. For example, they move from stage one to stage two when they think that they have finished the exploration and planning processes and now need to start knowledge revision and reflection. The transition to the third stage provides an opportunity to put the whole knowledge in a single space of hypertext in such a way that facilitates self-review. However, if learners need to revise the relationships between knowledge components in the two upper stages, they can go back to the planning space in stage 1 in order to alter these relationships between the corresponding graph nodes. Accordingly, the hypermedia layer attached over Web pages (in stage 1) as well as the published knowledge (in stage 2) will be adjusted to accommodate the changes on the planning space.

4.8 Implementation Highlights

This section aims to highlight some implementation details of the Knowledge Puzzle tool. Among many things, the focus will be on the implementation of the core functionality of the system, which is the transformation of the constructed knowledge map into a layer of hyperlinks, and then attaching it over Web pages.

4.8.1 Transformation of the Knowledge Graph to XML Format

The graph constructed in the planning tool is implicitly converted to an XML format in order to be machine processable. This also enables learners to save the graph information in an XML file, and then to rebuild the graph when the file is imported. Figure 4.9 shows an excerpt of XML code representing the main elements in the knowledge graph. The code describes the nodes and the relationships between them. It also describes the graphical attributes of each node which are required to rebuild the knowledge graph. The XML information of the graph is also used by the tool to construct the lists of hyperlinks which are then attached to the Web pages.

Each graph node is represented in the form of <node> element, which comprises the following sub-elements:

- <graphical_attributes>: refers to sub-elements identifying the node's graphical details such as position, width and height.
- <id>: refers to the node identifier. This is a code randomly generated when the node is first created.
- <title>: refers to the name associated to the node.
- <source_url>: refers to the URL of the source page which the node refers to.
- <selected_text>: if the node refers to a fragment of a Web page, the selected text is extracted from the page and stored using this element.
- <student_notes>: refers to the student's notes attached to this node.
- <start_node_index> and <end_node_index>: refers to the enclosing HTML nodes for the selected portion of the page. The use of these elements is detailed in the following section.
- <signature>: this element is used by the tool to identify misleading annotation. The use of this element is detailed in the following section.

In addition, for each link connecting two nodes the tool maintains data about it using the element <link>, which comprises the following sub-elements:

- <id>: refers to the link identifier. This is a code randomly generated when the link is first created.
- <name>: refers to the name of relationship between the two connected nodes.
- <inverse_name>: refers to the inverse name of relationship between the two connected nodes. For example, if the name of relationship is "requires", the inverse name is "is required by".
- <source_id> and <target_id>: refers to the id of the source and target nodes respectively.

```xml
<graph>
  <node>
    <graphical_attributes>
      <x>30</x>
      <y>300</y>
      <height>193.0</height>
      <width>139.0</width>
    </graphical_attributes>
    <id>n72lugwCQKf9WNHOSIuJ</id>
    <title>Chlorophyll</title>
    <source_url>
      http://www.chm.bris.ac.uk/motm/chlorophyll/chlorophyll_h.htm
    </source_url>
    <selected_text>Chlorophyll is the molecule that traps…
    </selected_text>
    <student_notes>student comments or self-notes</student_notes>
    <start_node_index>99</start_node_index>
    <end_node_index>452</end_node_index>
    <signature>46042562</signature>
  </node>
  <node>
    <graphical_attributes>
      <x>276</x>
      <y>18</y>
      <height>193.0</height>
      <width>145.0</width>
    </graphical_attributes>
    <id>nC7Ox1U8QULOj4VEKwvp</id>
    <title>Photosynthesis</title>
    <source_url>
      http://www.caribbeanedu.com/kewl/science/science04d.asp
    </source_url>
    <selected_text>Photosynthesis is the process in which carbon…
    </selected_text>
    <student_notes>student comments or self-notes</student_notes>
    <start_node_index>4131</start_node_index>
    <end_node_index>4146</end_node_index>
    <signature>84202580</signature>
  </node>
  <link>
    <id>ntHzznzfbI4wCL6hAdRc</id>
    <name>requires</name>
    <inverse_name>is required by</inverse_name>
    <source_id>nC7Ox1U8QULOj4VEKwvp</source_id>
    <target_id>n72lugwCQKf9WNHOSIuJ</target_id>
  </link>
………
</graph>
```

Figure 4.9: A sample XML code representing the knowledge graph components

4.8.2 Altering Web Pages by Accessing DOM Tree

Since the original page on the Web is static and hence cannot be modified, hypertext layering is applied by modifying the copy of the Web page that is cached locally after being loaded on the browser. The knowledge construction tool utilizes a Java based component Web browser, ICE browser 6[19]. The ICE browser implements portions of the W3C Document Object Model (DOM) Level 1 specification[20], a standard interface for programmatically accessing and modifying HTML documents. The DOM presents the document as a hierarchy of HTML elements. Each element has an index that enables direct access to it from within the document.

When a portion of a Web page is selected and added as a node on the planning space, the indexes at the start and end HTML nodes that enclose the selected portion are stored (refer to elements <start_node_index> and <end_node_index> in Figure 4.9). These indexes can identify the position of any HTML portion of the page source. Thus, the content of any portion can be manipulated by accessing its corresponding DOM elements. When a page is loaded on the browser and this page is referenced in the knowledge graph, the DOM tree of the page source will be accessed in order to highlight the parts of interest by modifying their colour and text attributes. In addition, the hypertext menus are constructed on the fly and attached to the page by creating the appropriate hyperlinks. All this is done without altering the layout of the pages and without imposing any significant time overhead. Thus, it will appear to the learner as if the original page is personalised to accommodate his/her own needs.

It should be noted that the modification of the original pages on the Web after they have been annotated might cause misleading annotations that point to wrong pieces of information. To minimize such an effect, each node is associated with a unique hash value (refer to element <signature> in Figure 4.9). This value is calculated from the HTML content of the annotated component. Every time a Web page is revisited, the hash values of its annotated components are recalculated and compared with the

[19] http://www.icesoft.com/products/icebrowser.html [last accessed 15/11/2008].

[20] http://www.w3.org/TR/REC-DOM-Level-1/ [last accessed 15/11/2008].

stored ones. In case of mismatch, the learners are warned of any possible misleading annotations and they are invited to re-annotate them.

Finally, it is worth mentioning that there are some existing hypermedia systems that aim to alter Web pages by directing HTTP requests to a proxy server and then altering documents when they come through the proxy (Tsandilas and Schraefel, 2003; Bechhofer et al., 2008; Djoudi, 1999). However, altering pages after being loaded on the browser has the advantage of not requiring any special proxy configurations.

4.9 Development Tools

The Knowledge Puzzle tool was developed in Java 1.5 with the support of the following libraries and tools:

- The ICEbrowser SDK[21]: a Java browser Software Development Kit that supports HTML/XML page rendering and DOM access.
- JGraph[22]: a Java open source graph drawing component.

4.10 Chapter Summary

This chapter demonstrates a tool to support knowledge construction during SDL from the Web. The development of the tool is driven by the requirements as outlined at the beginning of this chapter. The process starts with a planning space that supports meta-cognitive activities throughout the entire learning process. It encourages learners to construct a visual representation by externalizing knowledge that they learn from hypertext. The system then translates the visualized knowledge to a hypermedia layer and attaches it over a Web page. Therefore, the learner will be able to see the knowledge components on the Web interlinked and annotated exactly as he/she planned. The system then produces a hypertext version of the whole knowledge structure by extracting the information units from the source Web pages and organising them in a single Web page. Finally, the educational benefits and the uses of the tool throughout the stages of the learning process are revealed and justified.

[21] http://www.icesoft.com/ [last accessed 15/11/12008].

[22] http://www.jgraph.com/ [last accessed 15/11/2008].

Chapter 5
SWLinker, A Framework to Support Browsing of Hypertext Learning Resources

5.1 Introduction

Browsing hypertext resources in a self-directed way is often associated with problems that can distract learners and affect their self-motivation. These problems have been listed and discussed in section 1.3. This chapter presents a system called SWLinker (Semantic Web Linker) that aims to empower Web browsing in a way that promotes SDL from the Web. It is a Semantic Web-based system, offering three services to facilitate the learner's interaction with both information resources and human objects as follows:

1. **Semantic interpretation**: SWLinker offers the ability to bring semantic interpretation to Web browsing with little extra effort by learners. It acts as a knowledge aggregator that the learner can call on demand in order to annotate the content of the Web page being browsed with additional layers of complementary knowledge relevant to its content. The attached annotations supply learners with the information required to interpret the page content and resume the learning task. Thus, their need to suspend the learning task while they seek assistance or search the Web will be sufficiently reduced.

2. **Instructional guidance for in-depth learning**: In order to help Web users learn unfamiliar topics thoroughly and systematically, the system offers an on-demand annotation service that provides in-depth learning through domain concepts. The service attaches a guiding menu of links to each domain concept present in the text of the Web page. The attached menu expands to multiple levels to enable learners to find prerequisite concepts and sub-topic concepts from inside the Web page.

3. **Matching users with related browsing interests**: The system acts as a collaborative browsing assistant that helps learners to discover other users on the Web who have similar or related interests. The user-matching in the system follows an ontological approach to measure the semantic relatedness between the pages being browsed in order to discover users with semantically related browsing interests.

The following sections illustrate the design and architecture of SWLinker. The functionalities of the system will be demonstrated through detailed usage scenarios.

5.2 Design Principles

The overall goal of this project is to promote SDL from hypertext resources. This goal can be divided into a set of design principles that are essential for both learner and system administrator in order to make the system easy to use and extensible. These principles are:

- SWLinker should extend a standard Web browser so that learners are not required to learn new tools. It should also process and annotate any Web page regardless of its structure.

- SWLinker should use relevant methods to associate and view the various types of annotations. These methods should preserve the appearance of the Web pages after being annotated so that users will not get confused or distracted. The view of the annotations should also be customizable and should not produce messy or disorganised content.

- SWLinker should utilize a standard specification for domain ontologies. This makes the system easily interoperable and extendible by the ability to reuse existing ontologies that conform to the utilized specification.

- The knowledge base used by the system should be populated and fed automatically (or semi-automatically) in order to minimize the effort required by the system administrator.

5.3 On-Demand Semantic Annotation in SWLinker

Unlike the normal annotation techniques, SWLinker automatically associates semantic layers to Web pages rather than relying on manual annotations. These layers are composed using a set of educational ontologies and knowledge assets retrieved from an external knowledge base. SWLinker uses ontologies to associate meaning with the concepts found on a Web page and then, on the basis of the identified meaning, to offer the learner the appropriate functionalities.

5.3.1 Usage Scenario

To demonstrate the system in use, we present the following scenario about a student, Mohammed, who is navigating the Web to learn about the latest Web technologies. Using a Web browser with the SWLinker extension, Mohammed starts with a Web page presenting an article about Web technologies. While reading, he encounters some difficulties in interpreting the page content. He uses the annotation service of SWLinker to get an overview about interesting topics related to Web technologies. Figure 5.1 shows the student's browser window after it is annotated by SWLinker, at his request. Terms in the page that belong to the domain of Web technologies are highlighted and converted to hotspots. These terms are annotated according to an existing ontology that presents the terminologies used in various Web technologies such as markup languages, Internet protocols, Web services etc. Clicking on any of them causes a brief definition or description for that term to pop up in a message box inside the Web page. The attached descriptions are retrieved from an external knowledge base that has been populated with information pieces elaborating all domain concepts.

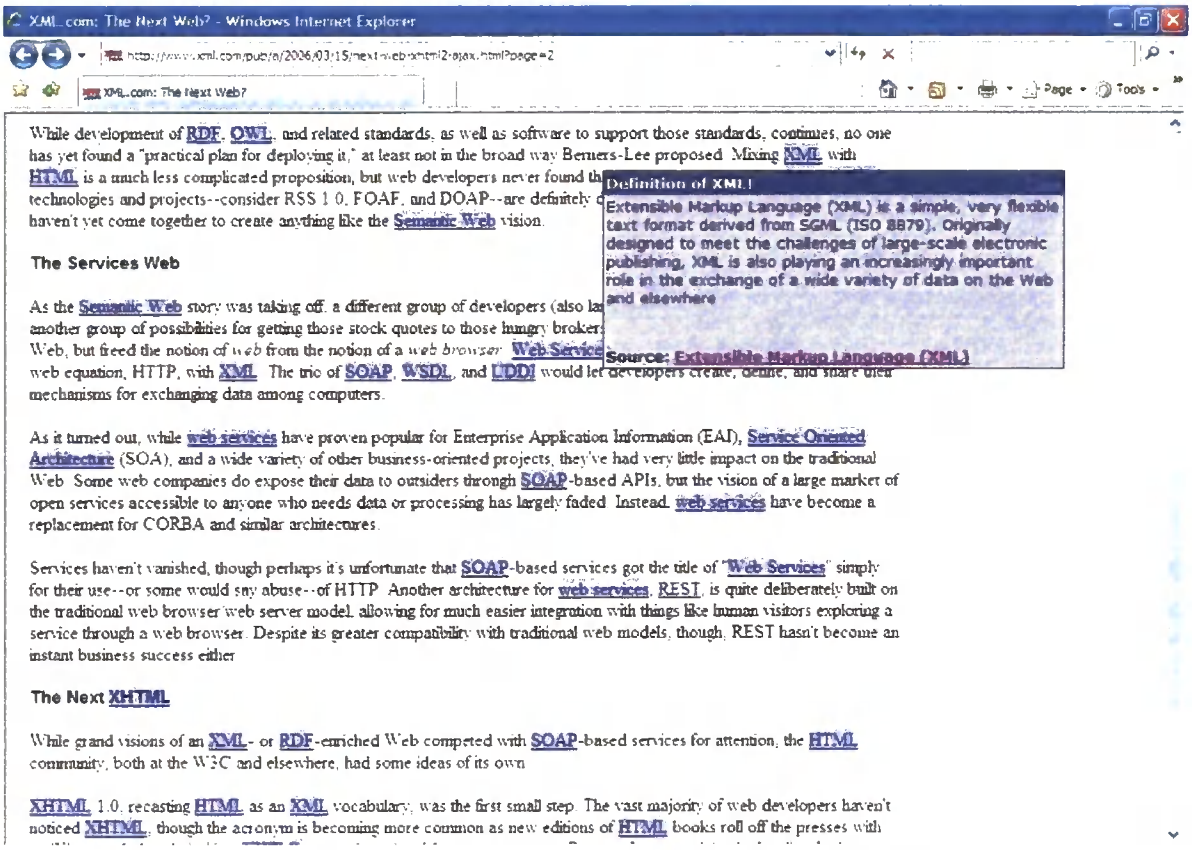

Figure 5.1: A Web page showing domain terms associated with explanatory information after being annotated by SWLinker.

After getting an overview about the unknown terms, Mohammed now wants to get a broader view by exploring related concepts and investigating the relationships between the different terms. He uses the second service of SWLinker to provide him with in-depth learning guidance. Figure 5.2 shows the terms contained in the page highlighted after invoking this service. By clicking on any term, a dedicated menu expands to multiple levels. Each menu shows the learning trajectory through that term and how it is instructionally related to other domain terms (e.g. sub topics, prerequisite topics and advanced topics). It also provides links to external learning resources on the Web to learn about the related terms. These menus are constructed according to the instructions path ontology, which defines the relationships between the domain terms from the author's perspective.

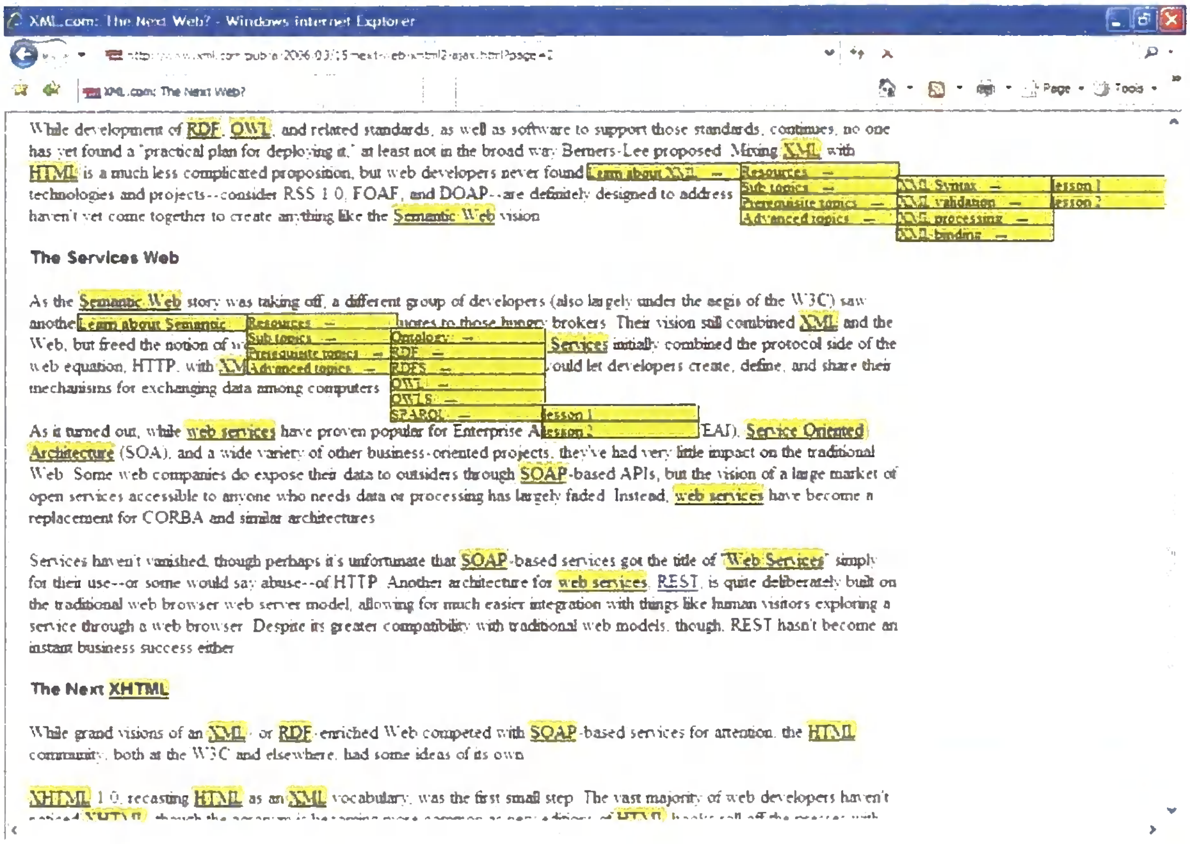

Figure 5.2: A Web page showing terms associated with guiding menus after being annotated by SWLinker.

5.3.2 Educational Benefits

By annotating Web pages while they are being browsed, it will be possible for learners to get the information they need without leaving the original site, and hence they can be kept more attentive and better engaged in the learning process than they would be if they had to go elsewhere to find that information. This thesis argues that the prevailing paradigm of pulling learners to knowledge resources is too restrictive for the needs of e-learning. Instead, the approach adopted here constitutes a shift towards the notion of pulling appropriate and relevant knowledge from where it resides, and making it available at the user's fingertips.

In addition, the learner becomes able to investigate the entire structure of learning about a particular term in a wider context by embedding menus that visualize the learning paths through the domain term. The goal of this guidance is to offer learners access not only to information about the included concepts, but also to knowledge resources dependant on instructional relationships among concepts. This menu acts as a knowledge portal that enables learners to explore the entire learning space of a particular concept and to study related subjects without leaving the Web page being browsed. Embedding these menus inside the page content reduces the "loss of context" feeling because they become part of the context. They also provide information hiding (menus are displayed only when requested by the learner). Having the learning path visualized as a multi-level menu enables to view only the level of instruction that the learner needs while other levels remain hidden and can be displayed on request.

From the previous scenario, it can be seen that the system supports an expert's view, as if a teacher had put all the different learning resources available to the learner together in the same Web page. It also shows that the appearance of the Web page is preserved after the new components are attached. This eliminates any confusion that might result if the modified copy did not look the same as the original.

5.4 Context-based Co-Browsing in SWLinker

Collaborative Web Browsing, or Co-Browsing, aims at extending currently available Web browsing capabilities in order to allow several users to share their browsing activities. Collaborative Browsing in e-learning can be a valuable ingredient for successful learning when thought of as a means to support collaboration among learners. However, this thesis argues that for co-browsing to be an efficient e-learning activity, it should go far beyond the simple mechanism of linking users browsing the same Web pages; it should include more intelligent solutions. SWLinker proposes what we term *Context-based Collaborative Browsing*, in which the collaboration is based on the context of the Web resource rather than the resource itself. For instance, it enables the matching and linking of users who have browsed Web pages that are physically different but which, instructionally, belong to the same or related topics. This will enable users to discover and share related browsing activities even if they are browsing difference resources or follow totally different navigation paths.

5.4.1 Usage Scenario

To demonstrate this service in use, we return to our student, Mohammed, who intends now to search for users on the Web with similar or related interests. After registering for the co-browsing service provided by SWLinker, he starts navigating the Web with the goal of learning about XML technologies. When he opens a tutorial about XML, a new window, shown in Figure 5.3, automatically pops up in front of him after a while, notifying him of other users who are registered with the service and who have looked at topics related to XML technologies. These users could be either online (currently browsing the Web) or offline. The window shows the users' contact details (e.g. email address, homepage). It also shows the list of pages that they have recently navigated. The lists include tutorials for XML, XQuery and XSLT, which are all related to the topic that Mohammed is looking for.

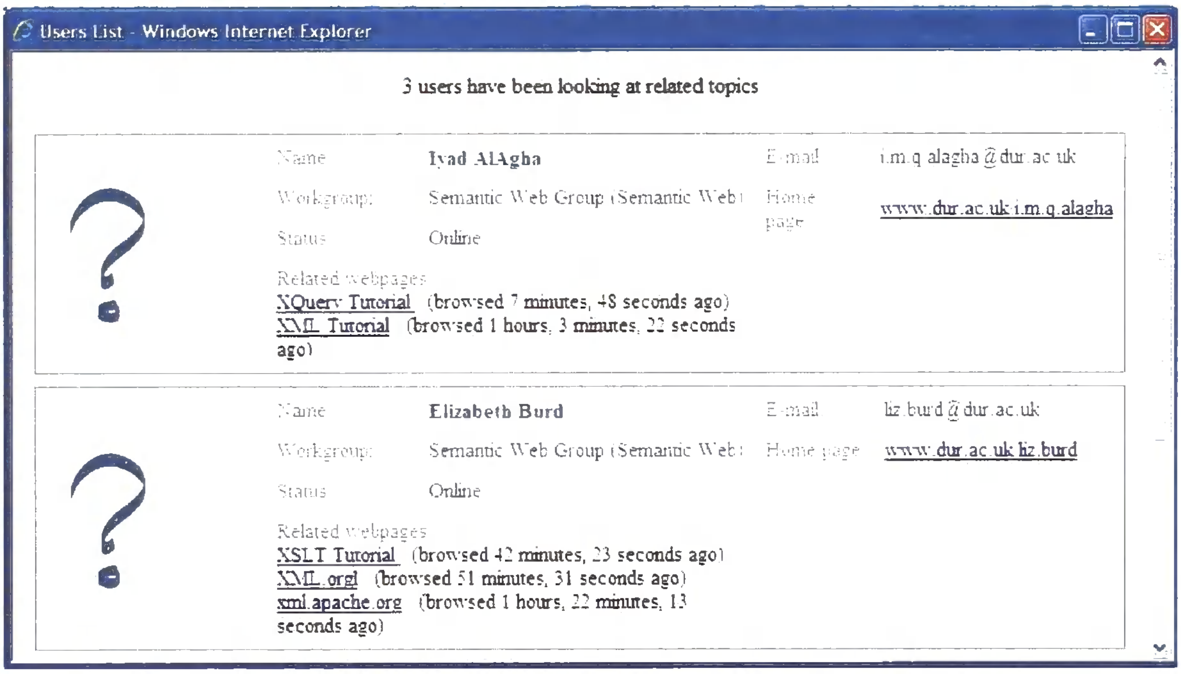

Figure 5.3: A Web page showing terms associated with guiding menus after being annotated by SWLinker.

5.4.2 Educational Benefits

From an educational perspective, the ultimate goal of the co-browsing service is to enable the system to predictably assemble effective groups of co-learners who have similar or related interests so that they become more productive by helping each other. It is important to note that the offered Co-Browsing service is not a screen sharing application. While users are informed of others with related browsing interests, they are not sharing the exact same experience and they do not see others' mouse cursors moving around etc. SWLinker instead focuses on sharing related browsing activities. It utilizes the context behind the browsing activity to become a unifying factor for synchronous or asynchronous collaboration across the Internet. Learners can be associated with others while they are involved in the navigational learning process. Additionally, the student is enabled not only to contact people with related interests, but also to explore Web pages on related interests browsed by them. This enables the student to share his/her browsing experience and discover new knowledge resources as he/she participates in multiple networks of co-learners. In the following sessions, the architecture of the system and the design approaches are revealed.

5.5 System Design

SWLinker has the service-oriented architecture, shown in Figure 5.4, in order to make its services accessible over the standard Internet protocols. The service provider, or the server side, comprises the modules that will handle users' requests. It consists of three modules: the Information Gathering (IG) module, the Instructions Design (ID) module and the Co-Browsing (CB) module. It also maintains and wraps all the repositories and ontologies used in the system. In the following sections, the components of the service provider and their functionalities are explained in detail.

The service requester, or the client side, is implemented as a plug-in to Microsoft's Internet Explorer, the standard Web browser that we selected. Extending one of the most commonly used Web browsers releases users from the burden of learning to use new tools. The plug-in is responsible for managing the interaction between the browser and the service provider as well as updating the browser output when responses are received. It also provides user-interface components that enable learners to interact with SWLinker services. It extends the right-click menu to add a new menu item for SWLinker's options. Clicking on any of the menu items causes the appropriate service on the service provider to be invoked and executed. It should be noticed here that the browser extension does not perform any content processing or reasoning tasks. The key point of reducing the complexity of the client's functionalities against those of the server side is to enable centralized updating and maintenance of system services and also to support easy implementation of SWLinker's plug-ins for other Web browsers.

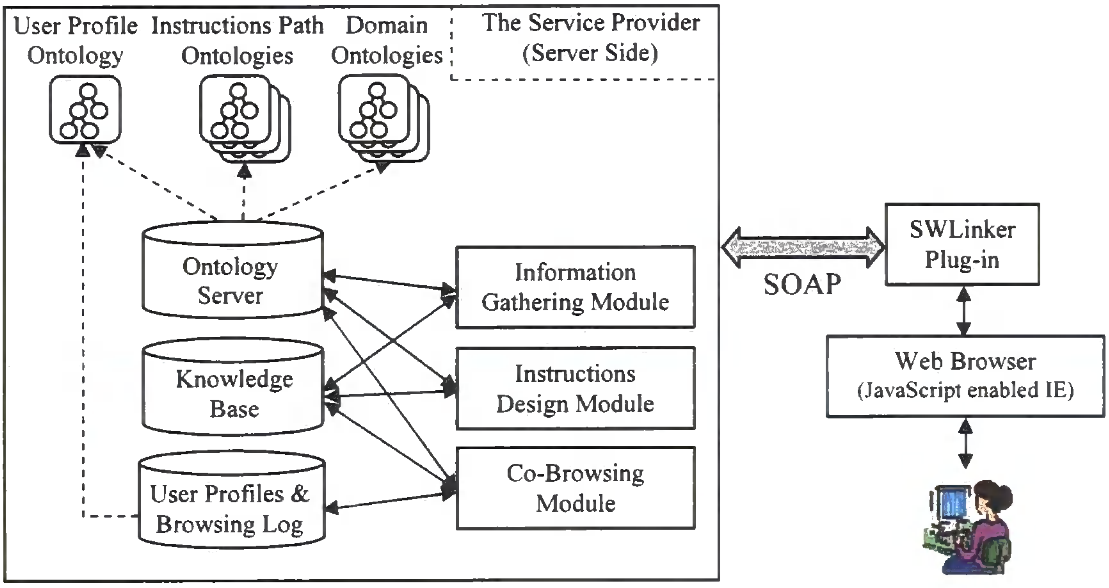

Figure 5.4: Overall architecture of SWLinker framework

5.6 Ontological Foundation

The Ontology Server stores the ontologies that represent the domains of knowledge covered by the system. These ontologies provide the formal descriptions and associations that are used to annotate the Web pages. The ontologies are represented in OWL-DL language (refer to section 3.4.3), and were built using Protégé[23], an open source ontology editor and knowledge base framework developed at Stanford University. Every single domain of knowledge should be represented by two ontologies: the domain ontology and the instructions path ontology.

[23] http://protege.stanford.edu [last accessed 20/11/2008].

5.6.1 Domain Ontology

The domain ontology defines the vocabulary used in any particular domain of knowledge and the relationships between the domain concepts. SWLinker accepts domain ontologies represented in OWL according to the SKOS Core specification (refer to section 3.10) in which domain instances are structured hieratically. The use of a standard specification such as SKOS makes the system easily extendible to process new domains of knowledge. Figure 5.5 shows a segment of the domain ontology that represents a sub domain of Web technologies. Note that the shown structure reflects the instructional view of the researcher.

Each domain concept (e-learning topic) is represented as an instance of skos:Concept class. The properties skos:prefLabel, skos:hiddenLabel, skos:altLabel are used to assign one or more terms referring to the same domain concept. These alternatives can enhance the search results by providing the ability to search for different alternatives representing the same domain concept. The hierarchal relations between concepts are represented by the properties skos:narrower and its inverse, skos:broader. skos:narrower is used to specify that a concept is narrower than another while skos:broader is used to specify the opposite. The class skos:ConeptScheme is used to represent the scheme of the domain, and this is linked with the top level concepts using the property skos:hasTopConcept.

When a request for annotation is received, the content of the Web page to be annotated is searched for all terms that match any of the ontology instances. In order to enhance the match results, it is preferable that every domain instance is referenced in the ontology by all its possible alternative names.

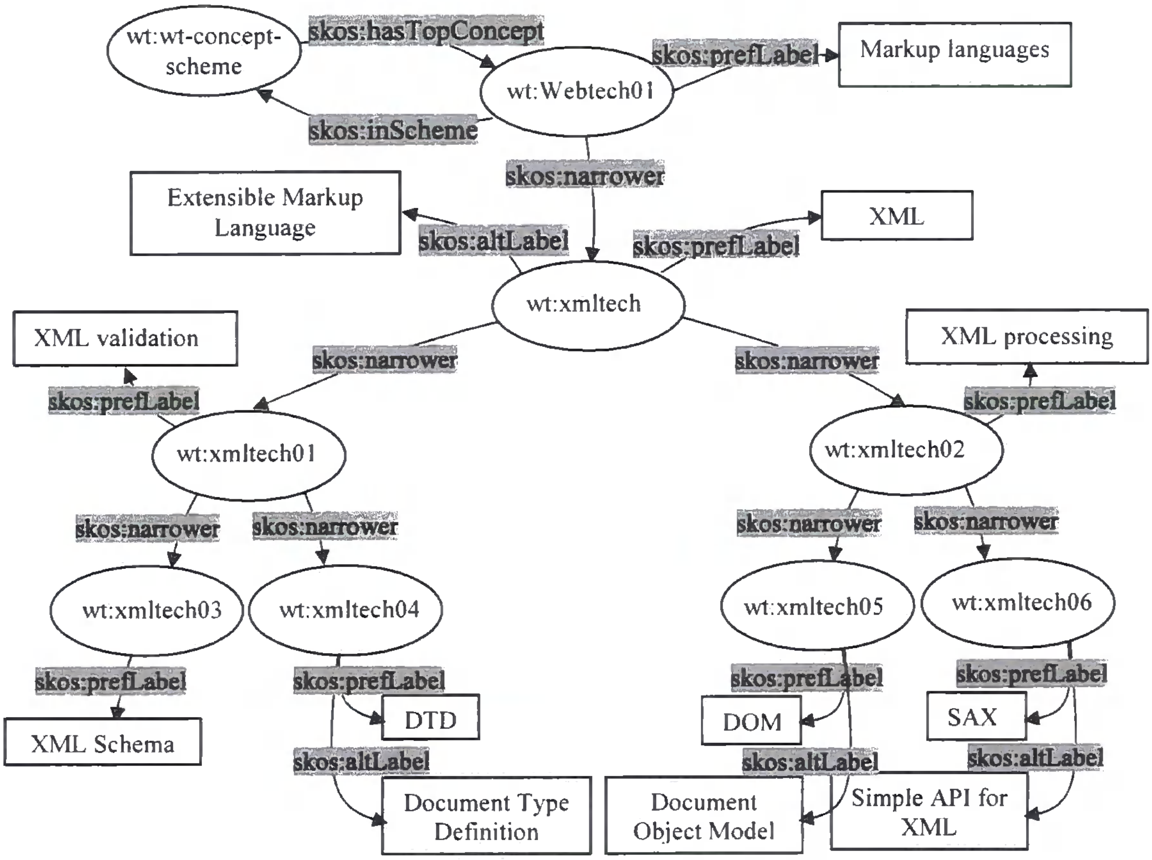

Figure 5.5: A segment of the domain ontology describing the concepts of the Web technologies domain

5.6.2 Instructions Path Ontology

The instructions path ontology defines learning trajectories through the topics defined in the domain ontology. It defines the prerequisite relationships between the domain concepts, thus reflecting a specific instructional approach to learning the domain. The defined learning trajectories are used during the annotation process to construct the guiding menus. These menus will be embedded in the Web page content to provide a detailed learning service for every domain term included.

The instructions path ontology introduces two properties that define prerequisite relationships between domain concepts: ip:isPrerequisiteFor and its inverse ip:requiresKnowledgeOf, where ip refers to the namespace of the instructions path ontology. The first one is used to assign a concept to its prerequisite concepts while the second is used to assign a concept to its subsequent concepts. In fact, the instructions path ontology extends the domain ontology by relating the instances of the domain ontology to each other. Although it is related to the domain ontology and can be included in it, this separation allows modification of the instructions path ontology according to the instructor's perspective while the domain ontology remains intact.

5.6.3 User Profile Ontology

SWLinker maintains a repository of user accounts and browsing logs to be used for the co-browsing service. The core of the co-browsing service is the shared browsing log that stores information about browsing activities for all users subscribed to the system. These browsing activities will be used by the system to identify the interests of each user and accordingly to match users who have similar or related interests.

Standard and ontology-based user profiles are interoperable, and they can be easily extended and combined with semantic metadata on the Web. Although there are plenty of user-modelling standards such as IMS LIP[24] and IEEE PAPI[25], these standards do not contain data about the user's browsing activities and logs which are essential for co-browsing in this project. Therefore, the user-profile ontology, shown in Figure 5.6, has been developed to represent relevant information about the users of SWLinker. It defines metadata elements for account details, contact information and group membership. These elements are initialized by the user who is required to create a new account in order to subscribe to the co-browsing service. The account and contact details are represented as instances of the classes up:AccountInfo and up:ContactInfo respectively. Each user can also be a member of one or more groups that are represented in the form of up:Group class. The group has a name and a competence which is one the concepts defined in the domain ontology.

The user-profile ontology also defines metadata for the user's browsing activities. The property up:hasStatus is used to by the system to describe the status of the user

[24] http://www.imsglobal.org/profiles/lipinfo01.html [last accessed 15/11/2008].

[25] http://edutool.com/papi [last accessed 15/11/2008].

(e.g. online or offline). Each browsing log is represented in the form of up:BrowsingLog class, which has the following set of properties:

- up:hasURL refers to the URL of the Web page being browsed.
- up:hasTime refers to the time at which the page is requested.
- dc:subject refers to a concept of the domain ontology that formally describes the subject matter of the browsed page. When the page is requested for the first time, the system extracts keywords from the page and attaches them to the browsing log in the user profile. Each browsing log instance can have one or more of this property based on the number of keywords extracted from the referenced page. These keywords will be used by the co-browsing algorithm to analyse user interests. The details of the co-browsing approach will be the theme of section 5.10.

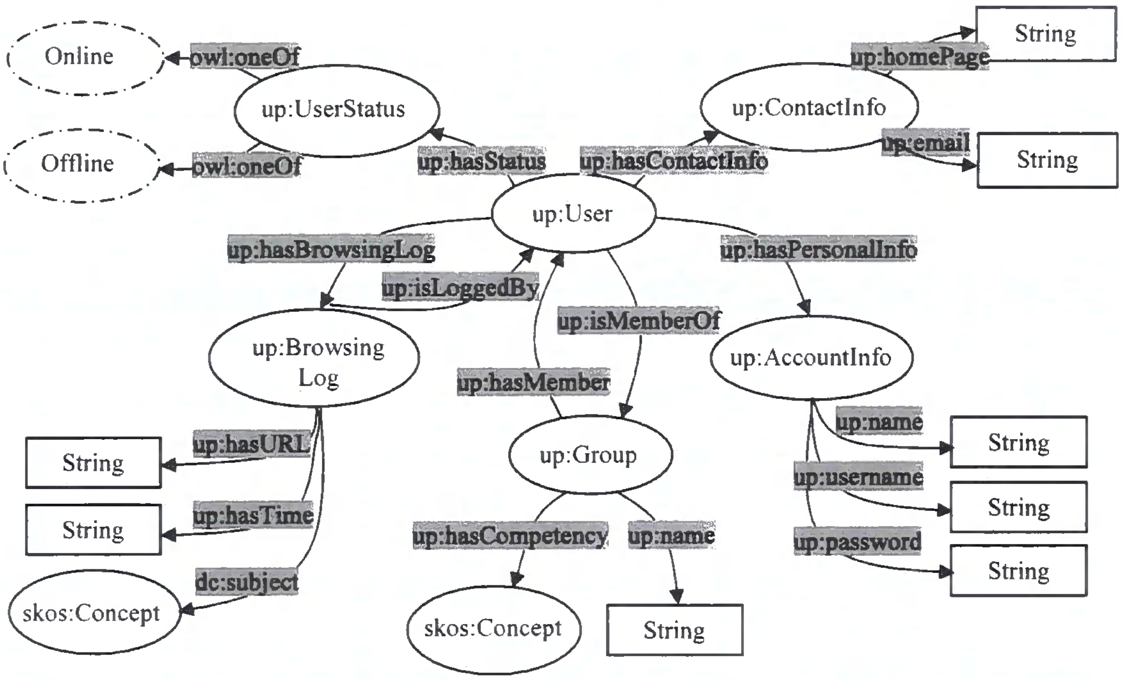

Figure 5.6: User profile Ontology

5.7 The Knowledge Base

The idea of enriching Web pages with a complementary knowledge is based on the availability of knowledge resources for all domain concepts. The knowledge base stores domain description information for annotation purposes. It is an RDF repository that contains two types of information component:

- **Definitions or brief descriptions of domain concepts**: These components are used by the system to construct the semantic layer to be attached to the page content. These resources should have high informative values with the least possible details in order to be sufficient for learners' needs yet also to be easily embedded inside Web pages.

- **Links to external learning resources elaborating the domain concepts**: These links are used in the guiding menus to offer links to explanatory and related learning resources.

A design principle for SWLinker is to make the system easily extendible to cover any domain of knowledge. A prerequisite to that is to minimize the effort required to populate the knowledge base. Therefore, a semi-automatic Web mining approach has been used to help the system administrator populate the knowledge base with highly informative descriptions for the domain concepts. The main purpose of the mining approach is to search the Web for the pages that contain obvious descriptions or definitions for domain concepts. Then, these descriptions are extracted from the containing pages and stored in the knowledge base, after being verified by the system administrator, in order to be used later in the annotation process. In what follows we describe the approach used to populate the knowledge base for any particular domain.

The domain ontology is traversed and every domain instance is submitted automatically to a search engine (e.g. Yahoo Web service), which then returns a set of pages. The top ranking pages are then mined for domain concept definitions or explanations. We applied the following definition identification patterns, defined using the experience from previous research (Liu et al., 2003; Harabagiu et al., 2000), to mine for definitions for the domain concepts.

- *{is | are} [adverb] {called | known as | defined as | explained as} {concept}*

- *{concept} {refer(s) to | satisfy(ies)}* ...
- *{concept} {is | are} {determiner}* ...
- *{concept} {is | are} [adverb] {being used to | used to | referred to | employed to | defined as | formalized as | described as | concerned with | called}* ...
- *{What} {is | are} [determiner] {concept}*
- *{concept} {- | :} {definition}*

The use of these patterns allows the filtering of search results in order to include only knowledge components that introduce the domain topics sufficiently well but with the least possible details. The page contents that match the above patterns are extracted from the source pages and presented to the system administrator to be verified before being stored as RDF triples in the knowledge base. Alternatively, the administrator can add the URL of the page in order to reference a Web resource that explains a particular domain concept.

Figure 5.7 shows how knowledge components are stored in the knowledge base as RDF triples. A set of Dublin Core metadata is used to represent the components in the knowledge base as follows:

- dc:subject identifies the concept(s) from the domain ontology that the component is subject of.
- dc:type refers to the type of the component. Any knowledge component can have one of two types: either "Definition" or "Link".
- dc:description is used to store either the definition body or a general description about the referenced URL.

```
<rdf:Description rdf:about="http://en.wikipedia.org/wiki/XML">
  <dc:subject rdf:resource="http://www.dur.ac.uk/i.m.q.alagha/Ontologies/domain.owl#xml" />
  <dc:type>Definiton</dc:type>
  <dc:description>The Extensible Markup Language (XML) is a general-purpose specification
  for creating custom markup languages .....</dc:description>
</rdf:Description>
<rdf:Description rdf:about="http://www.w3schools.com/xml/xml_whatis.asp">
  <dc:subject rdf:resource="http://www.dur.ac.uk/i.m.q.alagha/Ontologies/domain.owl#xml" />
  <dc:type>Link</dc:type>
  <dc:description> XML tutorial for beginners </dc:description>
</rdf:Description>
```

Figure 5.7: RDF triples representing sample components in SWLinker's knowledge base

5.8 The Information Gathering (IG) module

The IG module is responsible for dynamically generating the complementary knowledge layer and attaching it to the Web page content. On receiving a request from a client to annotate terms included in the Web page being browsed, the IG module downloads and parses the page content to extract terms that match any of the domain ontology concepts. A regular expression matcher is used to recognize the occurrence of concepts or instances in documents. The pattern for the matcher was created using the lexical labels of all concepts in the domain ontology. In order to improve the match rate, all possible labels for any concept, denoted by skos:prefLabel, skos:altLabel and skos:hidden, are included in the created pattern.

After identifying terms within the page that point to domain concepts, the IG module fetches a definition from the knowledge base for each discovered concept. The fetched definition is then attached to the Web page content. The local copy of the page is altered to attach the components to their corresponding terms exactly where they appear in the Web page. The attachment is performed by manipulating the HTML DOM and embedding JavaScript code into the local copy of the page source. This causes domain terms to be highlighted and enables the attached components to be revealed when the terms are mouse-clicked. However, the original page content remains untouched; only the interesting concepts and the corresponding text on the page are highlighted. The modified copy of the Web page is sent back to the client to be displayed on his or her browser on top of the original Web page.

5.9 Instructions Design (ID) Module

The Instructions Design module is responsible for constructing and attaching the guiding menus that offer in-depth learning about the domain terms contained in the page being browsed. On receiving a request from a user, this module uses the regular expression matcher to identify domain concepts included in the page. Then, it refers to the domain ontology as well as the instructional relationships defined in the instructions path ontology in order to construct a unique guiding menu for every discovered concept, linking it to its prerequisite, advanced and sub topics. Let x be the URI of the domain concept that points to the term to be annotated inside the page, the menu attached to the term will expand to multiple levels as the following:

- A sub menu for narrower concepts (sub-concepts or concepts that are more specific in meaning). These concepts are retrieved from the domain ontology by querying for concepts that are narrower in meaning than the concept to which the menu is attached. The following SPARQL query is used:

```
SELECT ?concept
WHERE  { x skos:narrower ?concept }
```

- A sub menu for prerequisite concepts. These concepts are retrieved from the Instructions Path ontology by querying for concepts that are prerequisites for the concept to which the menu is attached. The following SPARQL query is used:

```
SELECT ?concept
WHERE  { ?concept ip:isPrerequisiteFor x }
```

- A sub menu for advanced concepts. These concepts are retrieved from the instructions path ontology by querying for concepts that provide advanced knowledge for the concept to which the menu is attached. The following SPARQL query is used:

```
SELECT ?concept
WHERE  { ?concept ip:requiresKnowledgeOf x }
```

- For the main concept as well as each related concept included in the above levels, the system constructs a sub menu that includes links to Web resources explaining it. These links are retrieved from the knowledge base by querying for resources whose subject is every concept from the above sub-menus. The following SPARQL query is used:

```
SELECT ?res
WHERE  { ?res dc:subject x . ?res dc:type "Link"}
```

The result of the above procedure is a multi-level menu associated with every domain term contained in the page text. The menu gives sight of all the concepts that are semantically related to the annotated term through various relationships. It also offers links to relevant learning resources on the Web that explain these topics.

5.10 The Co-Browsing Module

The Co-Browsing module uses an ontology-based approach to match users who have similar browsing interests. The key to context-based co-browsing is to identify, from the users' shared browsing history, the Web pages whose contents are semantically related. Subsequently, the users who have navigated related pages are informed of each other's contact details because the system assumes that they have similar or related browsing interests.

Determining semantically-related Web pages is done by extracting keywords from the pages, and then determining the concepts from the domain ontology that best match these keywords. Accordingly, inferences can be made about what pages are related based on the semantic relatedness between the matched domain concepts. A distance based measure is used to estimate the semantic relatedness between concepts within an ontology. The shorter the path between any two concepts, the more semantically related they are likely to be. An ontology-based approach has been developed to help find mutual interests among users based on the Web pages browsed by them. The approach is explained as follows:

Users who intend to use this service should be first registered and authenticated by the service provider. The service provider maintains a shared browsing log that keeps track of all browsing sessions running at the same time.

On receiving a request for collaboration from a user, the system downloads the Web page currently being browsed and extracts keywords from its content. Following the common TF-IDF approach, text parts of the page content are considered and keywords are extracted and ordered by frequency. Only the most frequent keywords are taken, the rest are pruned.

Afterwards, the system identifies any potential concepts from the domain ontology that approximate the extracted keywords. To do so, both extracted keywords and lexical labels of domain concepts are stemmed and then compared. Stemming can improve the match rate as it finds the root form of the word by removing its suffix. An implementation of Porter's algorithm (Porter, 1980) is used for word stemming.

Domain concepts that match any of the page keywords are supposed to approximate the "subject matter" of the page with respect to the utilized domain of knowledge. If there is no matching, the Web page is considered to be out of the scope of the knowledge covered by the system, and hence the request for collaboration is ignored.

Subsequently, a new browsing log is added to the shared browsing log, denoting the user who is browsing the page, the URL of the page being browsed, the identified domain terms that approximate the subject of the page, and the time of browsing the page. Figure 5.8 shows a sample entry of the browsing log. Note that the metadata used to represent the browsing log are from the user-profile ontology. In addition, the domain concepts that describe the page content are stored in the browsing log. These concepts will be analysed later to identify users with related browsing activities.

```
<up:BrowsingLog
rdf:about="http://www.dur.ac.uk/i.m.q.alagha/Ontologies/BrowsingLog.owl#activity_OfZiz">
  <up:isLoggedBy
rdf:resource="http://www.dur.ac.uk/i.m.q.alagha/Ontologies/UserProfiles.owl#iyad_alagha"/>
  <up:hasURL>http://www.w3schools.com/xml/default.asp</up:hasURL>
  <dc:subject>
    <rdf:Bag>
      <rdf:li rdf:resource="http://www.dur.ac.uk/i.m.q.alagha/Ontologies/domain.owl#xml"/>
      <rdf:li rdf:resource="http://www.dur.ac.uk/i.m.q.alagha/Ontologies/domain.owl#html"/>
    </rdf:Bag>
  </dc:subject >
  <up:browsingTime>1205608885484</up:browsingTime>
</up:BrowsingLog>
```

Figure 5.8: RDF triples representing a sample browsing log.

The system then searches the shared browsing log for semantically-related Web pages. These pages are identified by measuring the semantic relatedness between the sets of domain concepts that describe the subject of the pages. The semantic relatedness between two Web pages *A* and *B* is measured using the following algorithm:

Let $\{x_1, x_2, ..., x_n\}$ and $\{y_1, y_2, ..., y_m\}$ be two sets of domain concepts denoting the pages *A* and *B* respectively. The semantic relatedness between *A* and *B* *is* calculated using the following equation:

$$R(A,B) = \frac{1}{n.m} \sum_{i=1}^{n} \sum_{j=1}^{m} w(x_i, y_j).$$

where $w(x_i, y_j)$ is the semantic relatedness between every two concepts x_i *and* y_j, and is calculated using the following distance measure:

$$w(x_i, y_j) = \begin{cases} 1 - \dfrac{d}{d_{\max}} & if\ d \le d_{\max} \\ 0 & if\ d > d_{\max} \end{cases}$$

d in the above measure is the shortest distance between the concepts x_i and y_j in the ontology, and is measured by the number of concepts on the shortest path from x_i to y_j. Concepts are considered unrelated if the distance between them exceeds a predefined number (d_{max}) of concepts. d_{max} can be adjusted based on the desired range of relatedness and the complexity of the ontology. This measure scores between 0 (for unrelated concepts) and 1 (for identical concepts).

The above distance measure is inspired from the distance measures in Rada et al (1989) and Leacock and Chodorow (1998). Slightly different from them, it allows the value of relatedness to change linearly with the distance within ontologies. This linearity is essential for the evaluation phase whereas the co-browsing algorithm will be assessed by expert judgment. Human experts will be supplied with a linear scale, in which the divisions are uniformly spaced, in order to rate the relatedness between sample pairs of Web pages. Therefore, the linear scores of relatedness obtained from the above measure will be easily comparable with the linear scores of relatedness obtained from experts. The evaluation of the co-browsing approach using experts will be discussed later in chapter 6.

Note that any two domain concepts can be connected through two different paths according to the type of connecting relations:

1. A path through the taxonomic relations (parent-child relations) from the domain ontology.
2. A path through the instructional relations from the instructions path ontology.

By default, the shorter of the two paths will indicate how much concepts are related to each other. However, the system can be configured to use a particular type of relations to measure the relatedness between concepts.

Using the above approach, the semantic relatedness value between the page being browsed by the user who invoked the co-browsing service and every page browsed by other registered users is calculated. Then, pages with the highest relatedness values are assumed to be semantically related. After identifying the semantically-related pages, the system refers to the browsing log to identify the users who have browsed these pages. Then, the system constructs a list of their contact details as well as the list of related pages browsed by each of them. The resulting list is constructed as a new HTML page and sent back to the requesting browser where it pops up in a new window, as shown in Figure 5.3. Thus, registered users who are browsing Web pages on the same or related topics will be automatically notified with each other's contact details and related browsing activities.

From the client's view, users (registered for the co-browsing service) can browse Web pages normally while synchronization messages are sent periodically in the background to the service provider to update the user status. Once the user opens any page on the browser, a request for collaboration will be sent to the service provider, which will start executing the above procedure to identify other users with related browsing activities.

5.11 System Deployment

Configuring SWLinker to cover a particular domain of knowledge requires the following steps:

1- **Developing Ontologies:** Each domain of knowledge is represented by two ontologies: the domain ontology and the instructions path ontology. The domain ontology should be compliant with the standard SKOS specification. It is recommended that the domain ontology includes all the alternatives that refer to every domain concept since this will increase the match rate and thus the number of annotations within the Web page.

2- **Populating the knowledge base:** After developing the domain ontology, the system administrator activates the Web mining approach explained in section 5.7 to feed the knowledge base with definitions and explanatory Web resources for the ontology terms. Figure 5.9 shows the user interface that enables the administrator to send search queries to the underlying mining approach, which automatically searches the Web and returns a list of candidate results to the administrator to choose from. The right side of the interface depicts the content of the domain ontology in a tree-like format. The right side shows the search options, whereas the administrator can choose either to search for all domain terms, or for particular terms from the ontology tree.

3- **Manipulating the knowledge base:** The administrator is allowed through another user interface, shown in Figure 5.10, to browse and manipulate the knowledge base. When a category is selected by clicking on its link on the ontology tree, information stored in the knowledge base and related to the selected category is displayed on the left. This information is what the system uses to annotate the occurrences of the term in the Web page content. There are three basic manipulation methods: adding, deleting and editing.

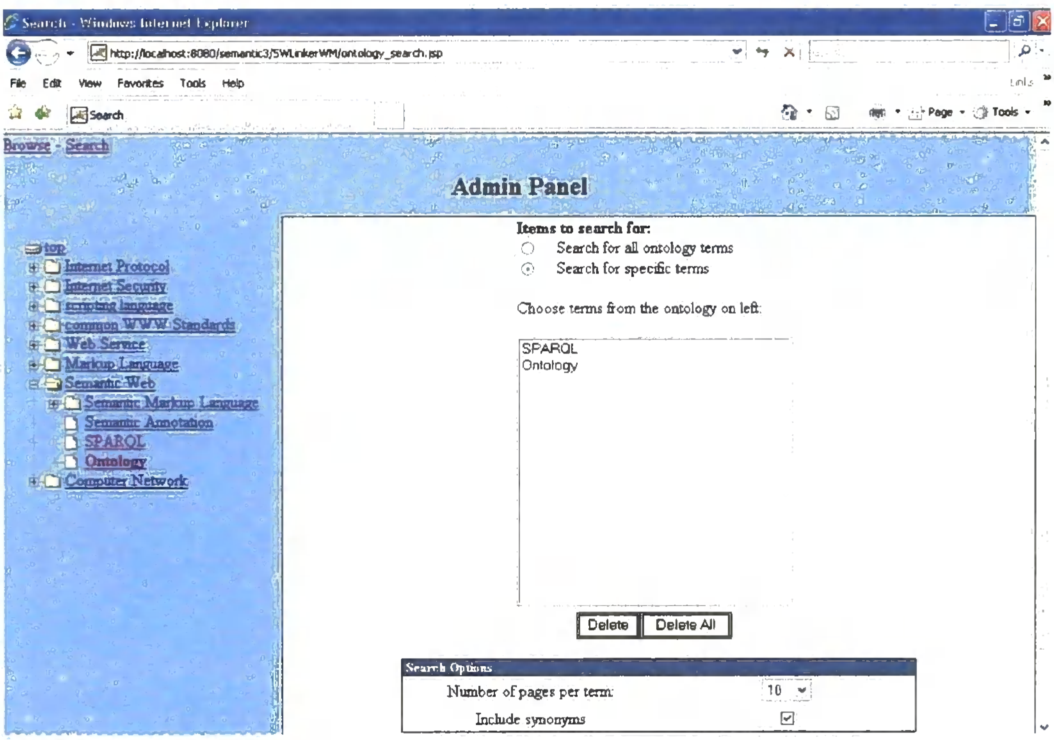

Figure 5.9: The user interface of the Web mining approach used by SWLinker

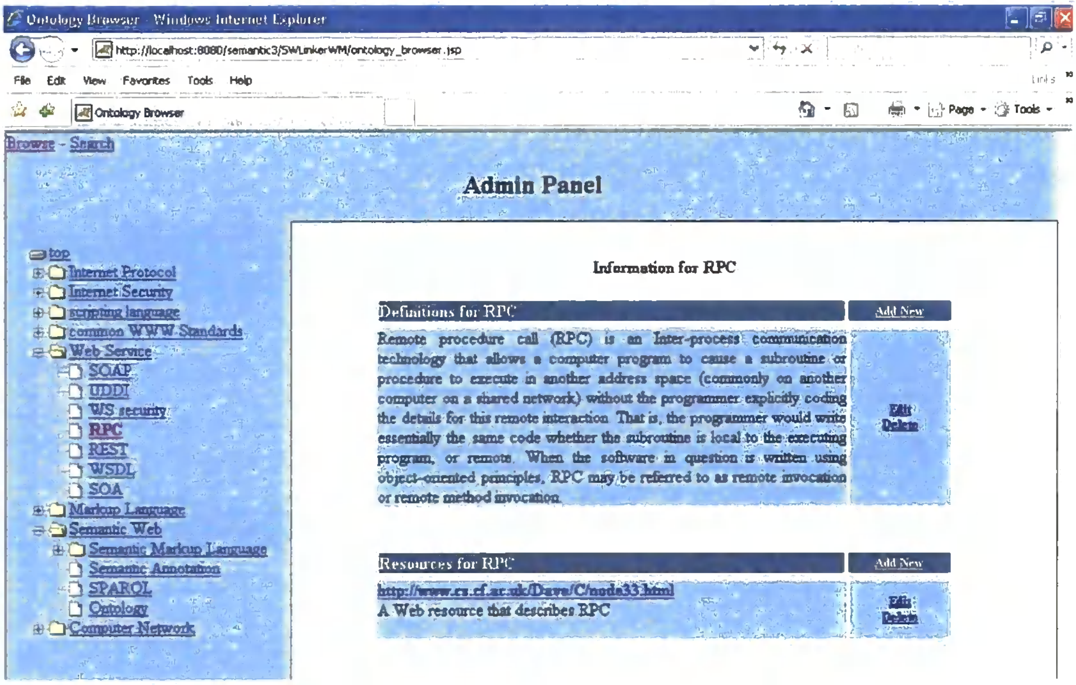

Figure 5.10: Browsing and manipulating the knowledge base contents used by SWLinker

5.12 Related Work

SWLinker provides two principal services, namely, the on-demand annotation of Web pages and the collaborative browsing. Therefore, each of these services is discussed separately when the system is compared with existing research efforts.

Many existing systems have supported the on-demand annotation of Web pages by utilizing concepts and relations captured from ontologies. Some of the inspirations for SWLinker were the Magpie (Dzbor et al., 2004), CoHSE (Bechhofer et al., 2008) and KIM (Popov et al., 2003) systems, which were discussed in section 3.11.4. These systems are similar to ours in adopting an ontology-based approach and extending standard Web browsers. However, SWLinker is distinguished by its different design goal. It aims at promoting SDL from hypertext resources by interpreting Web pages, allowing the discovery of instructionally-related topics and matching users who have related browsing activities. While the above solutions utilize ontologies for the purpose of ontology driven linking (as in COHSE) or named entity recognition (as in Magpie and KIM), SWLinker utilizes the instructional and hierarchal relationships defined between concepts to offer instructional and in-depth assistance through the constructed guiding menus. This thesis claims that integration of such instructional assistance with Web browsing can significantly promote SDL on the Web. This claim is supported by the literature surveys (Narciss et al., 2007; Hadwin et al., 2005), which stressed the necessity to support SDL on the Web by using instructional interventions. While such interventions could be technically delivered to learners through tutoring or learning management systems, SWLinker offers more generic instructional interventions that can be delivered through a standard Web browser and embedded inside any Web page related to the domains of knowledge covered by the system.

In addition, while other solutions do not tackle development and configuration issues such as ontology construction, standardization and knowledge base population, SWLinker pays more attention to these issues. It utilizes a standard specification for domain development (e.g. SKOS) that makes the system interoperable and easily extensible to cover additional domains of knowledge. Instead of using static databases, it employs a data mining approach to semi-automatically feed the knowledge base according to the utilised domain ontologies. While existing solutions support a single ontology at a time, SWLinker supports the annotation of pages according to multiple domains of knowledge, as long as each domain is represented by the appropriate ontologies in the ontology server that should conform to the SKOS specification. This allows, for example, to annotate terms in a Web page that belong to different domain ontologies.

With respect to co-browsing, there are a number of systems that support collaboration through augmented Web browsing. The CWB (Collaborative Web Browsing) (Esenther, 2002) is one of several systems that allow users to synchronously browse the same Web page. CWB is distinguished by being extremely lightweight as it does not rely on a plug-in, binary executable or Java applet. The I2I system (Budzik et al., 2002) allows users to leave calling cards on Web pages which state that the user would be happy to talk to other users on a topic related to the Web page contents. Magpie (Domingue et al., 2004) provides an interesting co-browsing service as it is based on a shared ontology-based semantic log that captures knowledge about the interesting entities found by a community of users browsing the Web. Pro-active trigger services can be fired whenever interesting patterns of communal behavior are detected. These patterns indicate overlaps indicating users are browsing related Web resources that cover the same research areas, have the same author or are owned by the same organization. All the above systems are based on linking users who are browsing the same Web pages, as in CWB and I2I, or Web pages that share the same entities as in Magpie. However, SWLinker is distinguished because it matches users by measuring the "instructional relatedness" between the "subject matters" of the Web pages browsed by them. Thus, it can provide more appropriate results for the field of learning where the instructional associations play the major role in finding users with related or complementary learning needs.

In addition, collaborative learning techniques apply different approaches to match and link users with similar interests. Similar user-interests can be identified by comparing the users' activities, preferences, profiles and/or browsing histories. The limitation of these approaches is that they consider the "similarity" rather than the "semantic relatedness" of different interests. It is important here to highlight the difference between the terms 'semantic similarity' and 'semantic relatedness'. Semantic relatedness is a more general concept than semantic similarity. Semantic similarity typically shows a synonymic relationship. For example, XML is the same as extensible markup language. However, other relationships are possible too: XQuery and XML have a part-whole relationship but they are not the same. Concepts that are not considered semantically similar can very well be semantically related. Accordingly, topics of interest for different users could be semantically related in spite of being apparently different. SWLinker utilizes domain ontologies to discover the semantic relatedness between users' browsing activities.

Ontologies have been widely used in the field of information retrieval to measure semantic relatedness between terms. Therefore, this thesis does not claim that the proposed ontology based algorithm for user-matching makes any novel contribution to the field of information retrieval, which is beyond the intention of this thesis. However, the contribution lies in the integration between collaborative browsing and ontology based semantic relatedness measure. To our knowledge, there is no approach that utilizes a shared ontology in collaborative browsing in order to match users based on the semantic relatedness between pages browsed by them.

5.13 Development Tools

The service provider (server side) was developed entirely in Java 1.6 with the support of the following libraries:

- Jena API[26] for ontology manipulation and interface.
- Apache axis 1.4[27]: to implement the Java Web Service through which the system offers its services over the Internet.
- Jericho HTML Parser[28]: required to manipulate HTML documents and attach the annotations and other components.
- Yahoo Search Web services SDK[29]: required to communicate with Yahoo Search service.
- DHTML tip messages[30], multi-level menu[31] and tree menu[32]: JavaScript components used for annotations, guiding menus and ontology browsing respectively.

The service requester (client side) was developed in Visual C++.NET. It was implemented as a Browser Helper Object (BHO). A Browser Helper Object is a DLL (Dynamic Link Library) that attaches itself to every new instance of Internet Explorer. It is used to intercept the Internet Explorer events, invoke the Web service in the background upon the user's request, and display the results.

5.14 Chapter Summary

This chapter presents SWLinker, a distributed system that leverages ontological engineering to enable users to access complementary and in-depth knowledge resources through a standard Web browser. The proposed approach supports real-time

---

26 http://jena.sourceforge.net/ [last accessed 20/11/2008]

27 http://ws.apache.org/axis/ [last accessed 20/11/2008]

28 http://jerichohtml.sourceforge.net/doc/index.html [last accessed 20/11/2008]

29 http://sourceforge.net/projects/jyahoosearchsdk/ [last accessed 20/11/2008]

30 http://www.bosrup.com/web/overlib/ [last accessed 20/11/2008]

31 http://www.dynamicdrive.com/dynamicindex1/blmmenu/index.htm [last accessed 20/11/2008]

32 http://www.softcomplex.com/products/tigra_tree_menu/ [last accessed 20/11/2008]

interpretation of any Web pages existing on the Internet by attaching semantic layers of knowledge chunks. It also enables learners to discover domain terms in a wider context by embedding portals that offer a grand vision of all instructionally-related concepts and sub topics. Learners are still able to query knowledge bases and associate knowledge with what they are currently browsing but, crucially, without leaving the original site and without interrupting the ongoing learning process. This seamless integration between the learning process and knowledge resources provides more situational and flexible interaction with the knowledge gained. Being implemented as a plug-in to a familiar Web browser, it featured a friendly way for ordinary users to perceive the benefits of the Semantic Web and to learn how to interact with semantically enriched knowledge.

Furthermore, SWLinker is capable of intelligently linking learners engaged in shared browsing activities. The user-matching in the system utilises an ontology-based approach to measure the semantic relatedness between the pages being browsed in order to discover users with semantically related interests.

SWLinker is built on top of multiple ontologies. The extensibility and interoperability of the system are promoted by using a semi-automatic Web mining approach to feed the knowledge base with highly informative resources and by adopting a standard specification for ontology development (e.g. SKOS core). SWLinker is believed to be a step towards converting normal Web browsing into a valuable and interactive e-learning activity, at both individual and collaborative levels.

Chapter 6
Evaluation, Analysis and Discussion

6.1 Introduction

To evaluate the Knowledge Puzzle and SWLinker tools presented in this thesis, various evaluation metrics may be considered such as the usefulness of the tools in promoting learning, the extent to which learners' needs are met compared to learning without using them, usability and ease of use. Both tools share the same ultimate goal, which is promoting SDL from hypertext resources. Thus, it is essential first to understand how a SDL process can be evaluated. Researchers have distinguished between two groups of methods and instruments used to assess processes involved in SDL (Winne and Perry, 2000):

1. Methods that measure SDL as an aptitude, describing relatively stable qualities or attributes of the student, and enabling prediction of future behaviour (cognition and motivation). These methods are mainly concerned with psychological aspects involved in the learning process. They include self-reporting questionnaires, structured interviews and teacher judgments.

2. Methods that measure SDL as an activity characterised as more complex measures that collect information on the states and processes the student undertakes while he/she is self-studying. These methods include trace methodologies and observation measures.

The second method was adopted in this thesis. A collection of qualitative and quantitative methods were used to observe and analyse the execution of the SDL task by participating students while they used the technologies presented in this thesis, and the outcomes of their use. These measures seem to be more appropriate for the experimental work than the methods suggested in the first category; they allow for a more systematic and technical assessment approach and they facilitate the explicit identification of the strengths and weaknesses of the work presented in this thesis.

The evaluation strategy adopted in this thesis was based on conducting controlled experiments through which participants used the developed tools to carry out SDL tasks. Subsequently, the tool utility and learning outcomes were assessed using the following methods.

- **Analysis of users' behaviours and activities while using the tools.** Observation and implicit measures of user behaviours can be used to analyse the utility of the tools, the user satisfaction and the task success. This approach has a long history in information retrieval from the use of hypermedia systems where relevance feedback from user behaviour is used to indicate a user's needs, interests and preferences (Kelly and Teevan, 2003; Oard and Kim, 2001).

- **Comparative analysis.** The value added by the tools can be exposed by running the learning task with, and without, the tools, and then making a comparative analysis between the two outcomes. An alternative strategy was to provide users with several optional services, including the services offered by the tools to be assessed along with other services, and then determine how users utilize these services and what particular services they favour. Both strategies were adopted to evaluate the tools presented in this thesis.

- **User questionnaires.** Questionnaires were devised and circulated to explore the experiences and views of participants and to investigate to what extent the tools helped them carry out the given tasks. They were also used to enquire about the perceived advantages/disadvantages of the tools and to gather participants' suggestions and comments.

This chapter starts with the evaluation of the Knowledge Puzzle tool, followed by the evaluation of the SWLinker system. For each part, the experimental settings and evaluation metrics are presented, followed by a thorough discussion of the data analysis and the results obtained. The chapter ends with the evaluation summary and concluding remarks.

6.2 Evaluation of the Knowledge Puzzle Tool

6.2.1 The Learning Task

A two-week-long study was performed to formatively evaluate the Knowledge Puzzle tool. Participants were first-year undergraduate students at the Durham University Department of Computer Science who were undertaking the "Computer Systems" module. A total number of 50 students participated in the study.

Students used the Knowledge Puzzle tool to complete an assignment in which they were asked to explore and study a collection of Web resources with the aim of writing a report that described the security threat 'virus'. The objective of the assignment was to gain knowledge of computer viruses as they relate to computer systems in general and to networks in particular. The students were told that the report should include the following sections:

- A general description of a virus, including how a virus is spread.
- A description of an instance of a virus.
- How the network is used and/or affected as a consequence of viruses.
- How to protect computer systems and users from viruses.

The selection of first-year undergraduates was thought to be suitable for the assigned task because such students had the basic Internet usage skills but they were unlikely to have advanced knowledge about Internet security threats.

In addition, since the focus in this thesis is on learning from hypertext resources, the selection of computer viruses as a learning topic had the advantage of being of a broad topic that could be learned effectively from Web pages. This is because such a broad topic is more likely to be discussed in Web pages, which usually provide general overviews of topics for non-researchers or users. Thus they are better than research papers, which are often more focused.

In addition, designing the task to have a number of components, as listed above, was essential for evaluating the tool. This composite nature of the task required students to navigate through multiple Web pages until they obtained the information that fulfilled the assignment requirements. The idea was to stimulate students to execute several navigation processes and to follow different navigation paths in a self-directed way using the Knowledge Puzzle tool in order to complete the assignment. Subsequently, their navigation behaviour was observed and analysed in order to assess the efficiency of the tool in helping them to resolve any problems they encountered.

6.2.2 The Learning Material

The learning material used in the task was a corpus of selected HTML resources related to the topic of computer viruses[33]. These resources were either existing Web pages collected from various Web sites or learning material converted to HTML format from other document formats (e.g. pdf, doc). The goal of specifying the learning material to be used was to unify the test conditions and release learners from searching the Web for relevant resources since the search time and effort were not part of the assessment metrics.

It was essential to ensure that the information contained in the given material was valid and reliable since any invalid information could distract learners. Therefore, the Web pages were inspected by the course instructor prior to the experiment in order to ensure that they contained valid information. In addition, during the process of selecting the learning materials, it was necessary to ascertain that the task solution was not entirely explained in a single resource. This was necessary since one of the main hypotheses that this thesis aimed to investigate was that the Knowledge Puzzle tool can efficiently help learners interlink separate segments of information that make up the task solution. Therefore, if the entire solution could be found in a single resource, it would be difficult to judge the success of the tool. Table 6.1 outlines about

---

[33] Learning material can be accessed through:
http://www.dur.ac.uk/i.m.q.alagha/cs/virus_resources.htm [last accessed 22/01/2009].

the corpus of the learning resources provided to the students and indicates their complexity.

	Learning Material
Number of Pages	42
Number of Links per Page	3.2

Table 6.1: Corpus of resources used in the experiment

6.2.3 Evaluation Measures

Before the experiment is explained, the measures used to evaluate the tool are introduced. Both quantitative and qualitative measures were used for the assessment process. The following section introduces these measures and explains the rationales behind using them.

6.2.3.1 Measures of Navigation Behaviours

The purpose of the experiment was to analyse the utility of the Knowledge Puzzle tool in order to ascertain whether or not it improved learning compared to learning without it. Students were expected to revisit pages frequently for various reasons, such as the need to recall previously-read information, to make notes about visited pages, to compare the contents of different pages and to create links with other information pieces. Thus, the adopted approach was to enable a comparative analysis to be conducted between the impacts of using traditional navigation mechanisms that exist in traditional browsing tools and the impacts of using the new mechanisms provided by the Knowledge Puzzle tool. The hypothesis was that the tool's special revisit and navigation aids have an advantage over traditional aids since they reduce disruption to the students' browsing activities and raise students' attention levels. The following two cases were defined:

- *Explore-Without*: denotes that the student explored and revisited the learning material using traditional navigation aids such as backtracking (e.g. back and forward buttons) and page bookmarks.

- *Explore-With*: denotes that the student explored and revisited the learning material using the special aids offered by the Knowledge Puzzle tool. These were navigating through the concept mapping tool or through the hypertext layer which the tool added over the visited pages.

The utility of the system was analysed by counting the number of pages visited, the number of revisits per page and the number of pages revisited during the primary exploration processes. Comparing the averages of these counts under both *Explore-Without* and *Explore-With* conditions, the utility of the tool can be evaluated. Further discussion about these measures and how they were used for assessment is presented with the discussion of the results in section 6.2.6.

6.2.3.2 Measures of Learning Outcomes

Outcomes of learning tasks are often assessed by conventional assessment methods such as examining students or marking-up their written essays. These methods help determine to what extent the task has been understood and completed by students. However, the outcomes of the task accomplished through the Knowledge Puzzle tool were assessed by inspecting the knowledge maps constructed by the students. This decision was based on the presumption that examining students or marking-up their essays without considering the learning strategy they adopted and the navigation paths they executed would fail to reveal the precise value of the Knowledge Puzzle tool. For example, examinations would be unlikely to answer questions such as the extent to which the students experienced cognitive and navigational difficulties while doing the task, and how well they used the tool to plan and manage their learning.

As explained in chapter 4, the constructed navigation map should mirror the knowledge structure in the mind of the student, and thus indicate whether or not the knowledge obtained fulfilled the task requirements. Our view on the value of analysing the constructed maps to assess the learning outcomes is supported by many research efforts that suggested the use of concept maps as a method of assessing learning. For instance, Okebukola (1992) ascertained that students who were successful in solving problems were also successful in concept mapping. Mason (1992) and Shavelson et al. (1994) used the scores obtained from concept maps to test and evaluate learning and they achieved successful results. In the light of these studies, the ability to represent self-constructed knowledge by using concept-mapping can be accepted as a valid evaluation criterion. The analysis method and results will be discussed in section 6.2.7.

6.2.4 Tool Preparation

The Knowledge Puzzle tool had to be prepared to capture the data required for the measures discussed in the previous section. The tool was configured to provide several navigation aids that facilitated revisiting pages. In addition to the special aids provided by the tool (e.g. the navigation path planning and hypermedia layering), the tool was designed to include navigation aids that are provided my most traditional Web browsers such as backtracking, a homepage button and page bookmarks. These were included in the tool in order to conduct the comparative analysis between the use of new navigation aids and the traditional ones.

In order to track the student's behaviour while using the tool, it was configured to log the participant's activities. The logging was done automatically in the background without intruding on the user's work. The log records were stored in client machines rather than in a central server. This allowed for each individual's activity to be captured on his or her personal copy of the tool rather than obtaining pooled general statistics typically captured by log server. The logged activities included all the actions and browsing processes performed by each student while using the tool to complete the assignment.

Prior to the experiment, the tool was demonstrated to the participating students and they were shown how to use the different options in the tool. They were also asked to download and use the tool, and they were referred to a detailed user guide.

6.2.5 Task Procedure

The experiment was conducted in the Durham University laboratories. Students were instructed to use the Knowledge Puzzle tool to explore the Web pages with which they were provided in order to complete the assigned task. They were instructed to use the planning space of the Knowledge Puzzle tool to plan the navigation path by adding references to the collected resources and linking them with appropriate relationships. As this was a SDL task, the students had the freedom to use any of the navigation aids provided by the tool to revisit pages in order to review the information they had obtained. In addition, participants were free to make decisions concerning the navigation paths they followed or the pages they needed to use. They were not obliged to explore all the provided pages but only those that they found useful.

After they had formulated a solution for the task, they were instructed to export the constructed map to a single HTML page and submit it to us. Finally, they were asked to fill in a user questionnaire.

During the task, the researcher and the demonstrators were available to offer guidance to the participating students and to collect general observations about the progress of the task.

6.2.6 Assessment of Users' Behaviors

6.2.6.1 Data Collection and Preparation

38 of the 50 students submitted the required files at the end of the task. The file submitted by each participant was a Web page that contained the details of the knowledge map he/she had constructed. This file was imported to the Knowledge Puzzle tool so that the map could be rebuilt in order to analyse the utility of the tool and to assess the learning outcomes. The submitted file also contained the complete browsing log, which included all the navigation actions made by the participant. These actions were implicitly translated into a meaningful format so that they could be easily processed. It had been decided that seven distinct actions would be captured while the tool was in use. These actions covered the use of both the traditional navigation aids (e.g. backtracking and bookmarks) and the use of the special aids provided by the tool (the concept map and hypertext layering). Table 6.2 explains these actions. The first column presents the actions to be captured. The reference codes used to denote these actions in the log file are shown in the second column. The third column lists the navigation aid denoted by each action.

Navigation Action	Reference Code	Navigation Aid Used
The back button is used to revisit a page	BACK_BUTTON	Backtracking
The forward button is used to revisit a page	FORWARD_BUTTON	Backtracking
A page bookmark is used to revisit a page	BOOKMARK	Page Bookmarks
A node on the constructed map is used to revisit a page.	PAGE_NODE	Concept Map (section 4.4)
A node on the constructed map is used to revisit a specific passage inside a page.	PASSAGE_NODE	Concept Map (section 4.4)
A hyperlink added by the tool is used to revisit a page	PAGE_KP_LINK	Hypertext Layer (section 4.5)
A hyperlink added by the tool is used to revisit a specific passage inside a page	PASSAGE_KP_LINK	Hypertext Layer (section 4.5)

Table 6.2: Navigation actions logged by the Knowledge Puzzle tool

Each log entry consists of three elements: the target URL, the action code and the time when the action was triggered. Figure 6.1 shows an excerpt of the log file.

```
url:http://all.net/books/virus/part2.html , action:PAGE_NODE , time: 10/12/2008-09:19:21
url:http://www.dur.ac.uk/i.m.q.alagha/cs/resources/html_book_01/what_is_a_virus.htm ,
action: PASSAGE_KP_LINK , time: 10/12/2008-09:25:32
url:http://www.securityfocus.com/infocus/1188, action:PASSAGE_NODE , time:
10/12/2008-09:30:51
```

Figure 6.1: An excerpt of the log file

6.2.6.2 Results and Discussion

The user logs stored in the collected files were inspected to analyse the utility of the tool. Table 6.3 shows the overall utility. During the experiment, each student explored an average of 16 Web pages and made a total of 48 revisits to them. Thus, the average number of revisits per page was 3.

Total number of revisits (a)	48
Number of unique pages visited (b)	16
Revisit per page (a/b)	3

Table 6.3: Analysis of page revisits

Table 6.4 shows how the navigation aids offered by the Knowledge Puzzle tool contributed to the overall utility. Looking at the knowledge maps constructed by the students, an average of 7.5 pages had been referenced by nodes on the constructed map. These are the pages actually utilized by students to obtain information about computer viruses among the total number of visited pages. The results show that these pages were revisited more than 35 times in total, either through the concept map or through the attached hypertext layer, with an average of 4.7 revisits per page.

Number of starting and terminal nodes	7.5 (47 %)
Number of revisits	35.5 (75.5%)
Revisit per page	4.7

Table 6.4: Utility of the Knowledge Puzzle tool

According to these results, only 47% of the visited pages were utilized by the students and these were the most visited pages. In other words, more that 75% of the page revisit actions (35.5 of 48) were made only on pages referenced from the knowledge map while about 12 revisits were made to other pages using other navigation aids (i.e. backtracking and bookmarks). This indicates that the Knowledge Puzzle tool could direct learners' attention only to the pages that included information of interest. A higher number of revisits to the Web pages actually utilised indicate that learners have formed an accurate model of knowledge and they are unlikely to be lost.

Figure 6.2 depicts how the total number of revisits was divided over the two cases *Explore_Without* and *Explore_With*. About 36 revisits were made by using only the navigation aids of the Knowledge Puzzle tool. These were executed either by double clicking of the nodes on the concept map, or through the hyperlinks added by the tool to the Web pages. On the other hand, only 12 revisiting process were made using the traditional navigation aids (backtracking and bookmarks). These results indicate that students gave preference to the tool's navigation methods over traditional ones.

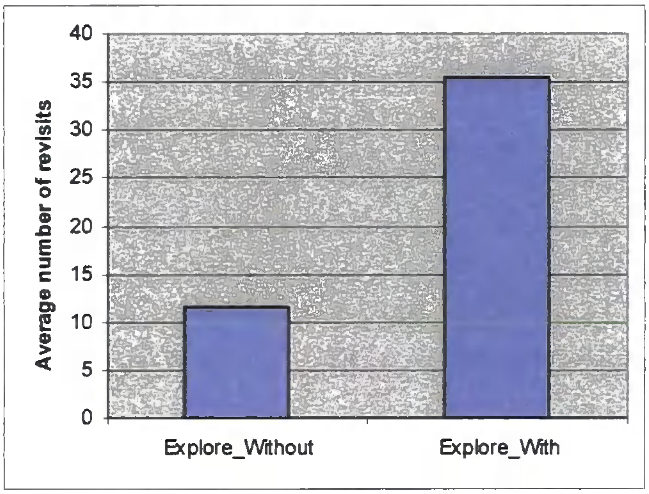

Figure 6.2: Analysis of the revisit actions under *Explore_Without* and *Explore_With* conditions

Figure 6.3 extends the previous data to show how much each navigation method was utilized. It indicates that most revisiting processes were made using the constructed map on the planning space, followed in popularity by the layer of hyperlinks created by the tool and attached over the visited pages.

Comparing the use of the knowledge map against the use of the hypertext layer, the results show that the students tended to prefer using the map nodes to revisit pages rather than using the attached hyperlinks. This result did not accord with our expectations because this thesis claimed that the transformation of the constructed map into a set of hyperlinks and annotations inside pages would reduce the dependence on the concept mapping tool. However, the results showed that students seemed to be highly dependant on the constructed map over all other navigation methods.

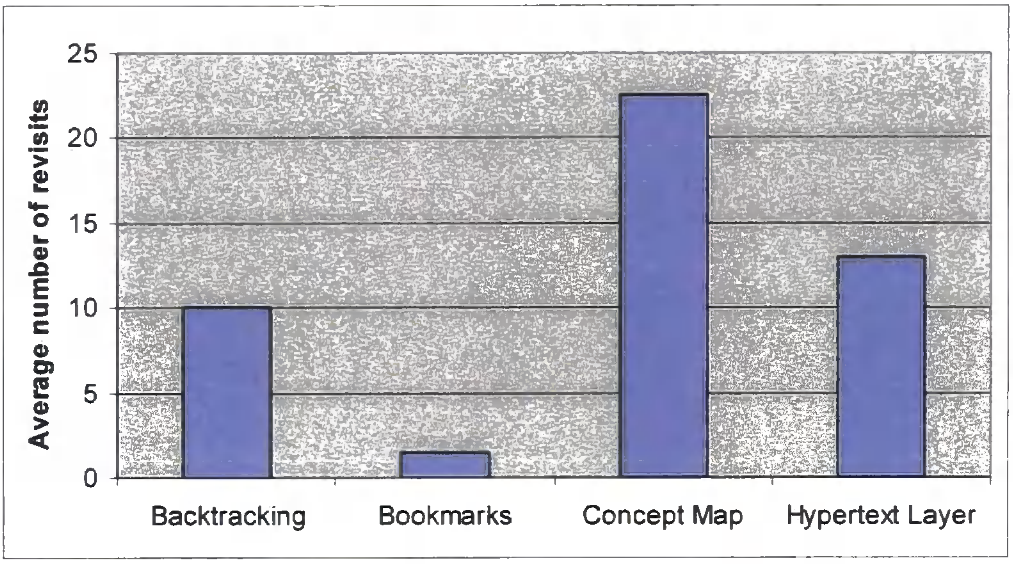

Figure 6.3: Analysis of the revisit actions performed through various navigation aids

In order to investigate the reasons for such behaviour, the actions performed by each student and the complexity of his/her constructed map were inspected individually. Table 6.5 shows information related to a sample of 5 students with various map complexity. The second column refers to the number of nodes included in the map constructed by each student, and this indicates the complexity of the map. The third and fourth columns show how many times the pages were revisited through the map and the attached hypertext layer respectively.

	Number of nodes in the constructed map	Revisits using the concept map	Revisits using the hypertext layer
Student 1	12	17	23
Student 2	12	19	27
Student 3	10	14	17
Student 4	7	15	8
Student 5	5	13	9

Table 6.5: Activities of a sample of 5 students, exposing the relationship between the map complexity and the utilization of navigation aids.

It can be seen from Table 6.5 that, as the complexity of the constructed map increased, the use of the concept map to revisit pages decreased whilst the use of the attached hyperlinks inside pages increased. This inverse relationship between the complexity of the map and the utilization of newly attached hyperlinks was expected due to the increasing effort required to manage and trace the constructed graph when it becomes large. However, these results were not obvious from the overall analysis of user logs because the average map complexity was low (the average number of starting and terminal nodes = 7.5). This overall result unveiled a possible limitation in the design of the experiment, namely that the size of the learning task was not large enough to assess the utility of the attached layer of hyperlinks. However, the inspection of individual usage of the tool resolved this limitation, to a certain extent, by showing that there was an increasing tendency to use the attached hyperlinks as the size and complexity of map increased.

6.2.7 Assessment of Learning Outcomes

The learning outcomes of using the Knowledge Puzzle tool were assessed by evaluating the knowledge maps constructed by students. The following aspects were used to determine enhancement of the learning outcomes:

1. **Validity and appropriateness of gathered information:** the Web resources referenced in the constructed map should include information that sufficiently fulfils the requirements of the task (learning about computer viruses).

2. **Validity and coherence of the map structure:** the overall structure of the map reflects the level of comprehension achieved by the student. This can be determined from the validity of the links connecting the graph nodes, the meaningfulness of the labels used and their concordance with the concepts of the proposition. The labelling of nodes and relationships should harmonize with information resources referenced from the map.

In general, students demonstrated a reasonable ability to reference valid information for the assigned task. A remarkable finding was that 8% of the overall referenced pages were not included in the corpus of pages provided to them by the assignment materials. Although this percentage is low, it shows that some students successfully used the tool to explore further Web pages in a self-directed manner. The analysis of the structure of the maps helped to identify several deficiencies in both the tool's functionality and the students' performance. In general, students were able to link nodes in a way that reflected a reasonable level of comprehension as most of the links were defined and labelled according to the task requirements. For example, relationships such as "method of spread", "how to protect", "effect on networks" and "instance of" were used frequently in the constructed maps. The main problem was in naming the graph nodes. In many cases, the names of the nodes did not reflect the topics learned from the referenced information on the Web. This was because students tended to rely on the names automatically assigned to the nodes when they were first created. Each node was automatically assigned the title of the source Web page to which that node referred. However, very often the broad title of page did not match the exact topic of the referenced segment. This limitation can be overcome by modifying the tool to prompt students to name the nodes once they are added, instead of relying on default naming.

Another limitation in the Knowledge Puzzle tool was revealed by the students' labelling of the links between map nodes. They tended to use two types of labels:

- Labels that denote hierarchal relationships between general topics and sub topics (e.g. "part of", "instance of", "has part").
- Labels that denote navigation goals between pages. These labels describe pedagogical and/or instructional relationships between source and terminal nodes (e.g. "explains", "defines", "justifies", "provides an example").

Although these sorts of labels revealed the students' ability to build multiple views of knowledge (hierarchal and instructional views), some students were confused because they had to decide what and when they need to use each type. In our view, this limitation can be resolved by enabling learners to build separate layers of knowledge, each of which uses a different type of labelling, each of which can be displayed or hidden upon the student's choice. Further discussion of this solution will be presented in chapter 7 when future work is discussed.

A clear limitation of the above assessment approach was that it did not have a standard reference to which results could be compared. However, in a SDL task such as the one presented here, it was difficult to construct a standard reference because of the different valid solutions that may be produced from the various navigation paths followed by students. However, the overall structure can give clear indications for the outcomes of using the tool, regardless of the path followed in learning.

Another limitation that influenced the quality of the learning outcomes was that some students did not take the experiment seriously in spite of it being part of the module assignment. The fact that the assignment was formative, and hence did not count towards the overall mark, made some students unenthusiastic about completing the task to the best of their ability. This was also clear from the collected data: of the 50 students who started the experiment, only 38 submitted the required data and only 44 handed in the assignment questionnaire. In addition, although the students received instruction on how to use the Knowledge Puzzle tool during the main lecture and had plenty of time prior to the experimental work to familiarise themselves with it, the majority did not do so. This meant that they had to spend some of the laboratory time allotted to the experiment to learning how to use the tool before they were able to start the experiment itself.

6.2.8 User Questionnaire

6.2.8.1 Questionnaire Design

The questionnaire was divided into two parts: The first part consisted of eight multiple choice questions, each with a four-point scale (Strongly Agree, Agree, Disagree or Strongly Disagree). The second part consisted of two questions which required written answers and which aimed to explore participants' perceptions of the tool and their suggestions regarding its design and use. A copy of the questionnaire is attached in appendix A. In what follows, these questions are explained and then the results are discussed. Each question is explained by providing the Question statement(s), the actual text used in the questionnaire, and then giving the rationale for asking the question.

Question statement -The Knowledge Puzzle tool helped me link separate pieces of information available on the Web.

Rationale for question – This thesis has claimed that the difficulty of interlinking segments of information on the Web hampers the knowledge construction process. This question aimed to reveal whether or not the tool could reduce this problem and thus enhance learning by enabling learners to build an interlinked structure for knowledge gained from the Web.

Question statement - The Knowledge Puzzle tool helped me navigate effectively.

Rationale for question – This question was used to investigate two hypothesises: 1) that the tool facilitated Web navigation by enabling students to plan and follow the sequence of pages that achieves their learning goals and 2) that the new features offered by the tool did not hamper the navigation process.

Question statement - The Knowledge Puzzle tool helped me directly access information I needed inside the Web pages.
Rationale for question – Enhancing accessibility contributes towards the overall satisfaction of Web-based learning. The Knowledge Puzzle tool automatically highlights and displays segments of interest within Web pages when they are visited through the graph nodes or through the attached hyperlinks. This question was asked to check if this option was operational when tested with different pages and to what extent it fulfilled the desired level of accessibility.

Question statement - The new type of links attached to the Web pages by the Knowledge Puzzle tool helped me navigate easily through the Web pages.
Rationale for question – The aim of this question was to assess to what extent the new hyperlinks and annotations attached by the tool to Web pages were more successful in facilitating Web navigation than the other navigation aids provided by the tool.

Question statement - The Knowledge Puzzle tool effectively reduced the amount I need to remember.
Rationale for question - This question aimed to identify to what extent the tool reduced the cognitive efforts associated with the navigation process. However, it was decided to avoid terms like "cognitive load" or "cognitive overhead" because participants might not know their meaning. Cognitive load is primarily associated with the human memory and overload occurs when short-term memory is pushed to its limit (Conklin, 1987). Therefore, the question asked about the reduction in the amount that participants needed to remember, since this is virtually synonymous with the reduction in cognitive effort and would be more easily understood by participants.

Question statement - Grouping and structuring information components in a single page helped me quickly review the knowledge gained from the Web.

Rationale for question – The aim of this question was to assess the service of generating a hypertext representation of the self-constructed knowledge, explained in section 4.6, by investigating to what extent it helped participants rapidly review the knowledge obtained from the Web.

Question statement - The Knowledge Puzzle tool is easy to use.
Rationale for question – This question aimed to evaluate the usability and learnability of the tool.

Question statement - The user-interface of the Knowledge Puzzle tool is friendly.
Rationale for question – This question aimed to assess the friendliness of the user-interface.

Question statement - What single aspect of the Knowledge Puzzle tool most supported your learning?
Rationale for question – Since the Knowledge Puzzle tool offers several facilities for knowledge construction (e.g. path planning, hypertext layering and generation of hypertext knowledge), this question aimed to identify what single aspect learners found to be most helpful for learning. It was hoped that the answers would provide a focus for future work.

Question statement - Do you have any suggestions to improve the Knowledge Puzzle tool?
Rationale for question – Suggestions from students can help open new directions for further enhancements as well as helping to identify the tool's weaknesses from the learners' point of views.

6.2.8.2 Results and Discussion

44 students handed in the questionnaire at the end of the experiment. Table 6.6 shows the results of the multiple-choice questions.

Question Statement	Strongly Agree	Agree	Disagree	Strongly Disagree
The Knowledge Puzzle tool helped me link separate pieces of information available on the Web	11	23	6	4
The Knowledge Puzzle tool helped me navigate effectively	7	18	12	7
The Knowledge Puzzle tool helped me directly access information I needed inside the Web pages	16	19	6	3
The new type of links attached to the Web pages by the Knowledge Puzzle tool helped me navigate through the Web pages	15	19	6	4
The Knowledge Puzzle tool effectively reduced the amount I need to remember	17	15	7	5
Grouping and structuring information components in a single page helped me quickly review the knowledge gained from the Web	18	20	4	2
The Knowledge Puzzle tool is easy to use	9	16	11	8
The user-interface of the Knowledge Puzzle tool is friendly	14	20	5	5

Table 6.6: Questionnaire results – The Knowledge Puzzle tool (Multiple choice questions)

Focusing on the tool's main goal, which is its ability to link separate information sources available on the Web, the majority of the students (77%) responded with either "Agree" or "Strongly Agree". Moreover, most of the students (80%) very positively rated the fact that the tool helped them directly access information of interest inside Web pages. These results demonstrate the overall satisfaction with the system's goal. On the other hand, only 57% of the students acknowledged that the tool helped them to navigate effectively while 43% either disagreed or strongly disagreed with that proposition. At first sight, this result indicates a much lower overall satisfaction with the tool's support for Web navigation. However, observation of the usage of the tool and analysing the students' suggestions for improving the tool has helped to identify a number of reasons for such a low satisfaction rating. These are as follows:

- Students were more familiar with using traditional Web browser tools than using our tool and they were required to make an additional effort to learn the new functionalities. This additional effort, even though small, can affect the students' motivation and satisfaction. This conclusion was confirmed by correlating it to the suggestions proposed by students to improve the tool. For example, many of them suggested improving the tool's usability by adding tabbed browsing, enabling keyboard shortcuts and improving rendering quality. All such suggestions explicitly reveal the influence of traditional Web browsers.

- Many of the students reported some poor rendering when browsing some Web pages. This shortcoming was attributed to the development of the tool in Java. As explained in section 4.8.2, the tool utilizes a Java component called ICEBrowser to enable the browsing of Web pages. Despite the advantage of ICEBrowser in rendering Web content over traditional Java Swing components, it still causes some rendering defects, especially with pages that include rich components such as Ajax. These defects can distort the appearance of the page, obstruct content annotation or sometimes cause the tool to crash. To our knowledge, the only way to resolve this limitation is to migrate to another programming language with better HTML rendering capabilities than that which Java currently offers.

However, this discussion suggests that the navigation difficulties encountered by students originated mainly from their lack of familiarity with the new tool or from implementation issues, and thus did not arise from the core contributions of the tool.

Regarding the layer of hyperlinks and annotations added by the tool over Web pages, most of students' opinions (77%) were positive, i.e., either "Strongly Agree" or "Agree", indicating that the new components facilitated navigation through the Web pages. Although the usage of the attached hyperlinks was limited for reasons discussed in section 6.2.6.2, this result proves that students realized quickly the benefit of adapting and linking pages as a way to enhance learning from hypertext resources.

Another encouraging result was the students' response to the question about whether the tool reduced the amount they needed to remember. 73% of the opinions were positive, indicating that the tool had effectively reduced the cognitive load. In addition, the great majority of the students (86%) positively evaluated the generated hypertext representation of knowledge by agreeing with the hypothesis that it facilitates quick review of knowledge gained from the Web.

Regarding the ease of use, 57% of participants agreed, or strongly agreed, that the tool was easy to use, while 43% had a negative view. This result was again expected due to the navigation difficulties discussed above. Finally, about 77% of participants positively evaluated the user-interface.

When asked about the single aspect that most supported learning, the answers varied greatly. Twelve students considered the linking of Web pages with hyperlinks that describe effectively the way they are related as the most beneficial aspect of the tool. Nine students reported that gathering all information sources, students' notes as well as the relationships, in a single HTML file helped them most because it was easy to review the knowledge gained. Eight students reported that the ability to highlight and access important bits of information inside Web pages was the most beneficial aspect. They also liked the ability to attach their own notes to the Web page contents. Finally, four students generally liked the visual representation of knowledge using concept mapping. Despite the variety of the students' answers, they all highlight the students' interest in the support that the tool offers for Web-based learning. Some of them offered the opinion that the concept behind the tool was innovative, very interesting and useful. One of them declared that he/she intends to use the tool in the future, especially for citations.

The students provided many valuable suggestions for future improvement of the tool. For instance, some students reported that the user-interface should be a little more intuitive. They suggested enhancements such as tabbed browsing and the ability to choose colours for greater accessibility. Some students suggested the ability to reference not only textual information inside pages but also other components such as images and videos. Other students proposed the need for more freedom in concept mapping through the ability to group nodes under particular concepts. Another student suggested the development of the tool as a plug-in to Firefox or Internet Explorer because this is the best way to benefit from the advantages of the tool as well as the usability of traditional Web browsers. Finally, many of them stressed the need to make the tool more reliable by improving HTML rendering capabilities.

6.3 Evaluation of the SWLinker Framework

6.3.1 Evaluation Overview

The SWLinker framework offers two different services to promote learning from hypertext resources, namely the annotation of Web pages and the co-browsing service. Each of these services was evaluated separately as follows:

- The annotation service was evaluated by conducting a controlled experiment in which participants were involved in a simple learning task. The aim was to assess to what extent the semantic annotations can better support learning and also to identify the strengths and weaknesses of the SWLinker's approach. The adopted evaluation approach was to compare the outcomes of learning with the use of the service with learning which did not use it. In addition, a user questionnaire was circulated at the end of the task in order to assess the users' satisfaction and to obtain suggestions on further improvements or amendments.

- The co-browsing service was evaluated by validating the underlying ontology based approach for matching users. We stress that a valid and precise identification of users with related interests is mainly based on the accuracy of the underlying algorithm in measuring the semantic relatedness between pages browsed by different users. Thus, if we can assess the extent to which the algorithm can effectively estimate how related Web pages are, this will accordingly lead us to an assessment of the validity of the co-browsing service in finding users with related interests. The evaluation was carried out by means of expert judgement: A set of pairs of Web pages was created, and then the similarity between the pages in each pair was rated by human experts.

Then, the rates obtained from the expert were compared with those computed by the proposed algorithm.

6.3.2 Educational Setting

Sample domain[34] and instructions path[35] ontologies were built for the evaluation process. The system was designed to provide information about various Web technologies. The vocabulary defined in the domain ontology comprised more than 170 terms denoting Internet protocols, Internet security, Markup Languages, Semantic Web technologies, Web Services, Computer Networks, common WWW standards and scripting languages. All possible synonyms for these terms were also included in the ontology in order to increase the potential match rate for the annotation process. Hierarchal relationships (e.g. "broader", "narrower", "related") as well as instructional relationships (e.g. "isPrerequisiteFor", "requiresKnowledgeOf") were also defined between concepts.

The knowledge base was populated with information sources that sufficiently covered all domain ontology concepts. The Web-based interface, explained in section 5.11, was used to semi-automatically populate the knowledge base with relevant information from the Web. The concepts, the relationships between them and the knowledge base contents were acquired from multiple sources such as Wikipedia[36], Webopedia[37], Dmoz[38] and Yahoo[39] directory, in so far as these sources provided formal controlled vocabularies for Computer Science related topics. Several Websites

---

[34] The domain ontology can be accessed through:
http://www.dur.ac.uk/i.m.q.alagha/Ontologies/domain.owl [last accessed 22/01/2009].

[35] The instructions path ontology can be accessed through:
http://www.dur.ac.uk/i.m.q.alagha/Ontologies/IP.owl [last accessed 22/01/2009].

[36] http://www.wikipedia.org/ [last accessed 22/01/2009].

[37] http://www.webopedia.com [last accessed 22/01/2009].

[38] http://www.dmoz.org [last accessed 22/01/2009].

[39] http://dir.yahoo.com/ [last accessed 22/01/2009].

such as W3Schools[40], W3C[41], assorted books and online course curricula were also scanned in order to obtain an overview of the named topics and to help to define instructional relationships between them.

6.3.3 Evaluation of the SWLinker's Annotation Service

6.3.3.1 Experiment Procedure

A set of 16 participants was selected for the experiment. All participants were students from different degree disciplines, excluding computer science. The assumption made was that the participants should not be acquainted with the domain of Web technologies so that the outcomes of the experiment would not be compromised by their pre-knowledge. However, participants should be reasonably skilled in using the Internet and in search techniques and should use them regularly. The participants were interviewed individually and their interactions on the computer were screen-recorded throughout the task.

The first part of the experiment aimed mainly to assess the value gained from the first service proposed by SWLinker, namely the annotation of domain terms in Web pages with relevant definitions. The objective was to investigate whether or not the added annotations supported the interpretation of Web content and promoted learning in terms of reducing the overall effort and time required to accomplish a Web-based learning task.

Each participant was given a Web page containing a short article from the Web[42], in which a collection of terms from the utilized domain ontology was embedded. Many of these terms, however, were not explained or defined within the given page. The purpose was to put the participant in a position where he/she needed further information in order to interpret the content and thus pursue the task. Participants were then asked to read and compile the article. Since participants were inexperienced

---

[40] http://www.w3schools.com [last accessed 22/01/2009].

[41] http://www.w3.org/[last accessed 22/01/2009].

[42] http://www.xml.com/pub/a/2006/03/15/next-web-xhtml2-ajax.html?page=2 [last accessed 22/1/2009].

in the field of Web technologies, it was expected that they would encounter unfamiliar terms within the context. Accordingly, they should seek definitions for these terms using a search engine of their choice.

After participants finished compiling the article, they were instructed to use the SWLinker's plug-in to Internet Explorer to define the terms included in the Web page, and then to compare information attached by the system to that obtained from the search engine.

The following part of the experiment aimed to assess the value of the instructional guiding menus that provide the learning paths for terms included in the Web page. The objective was to investigate whether or not the provided menus can help users to learn unfamiliar terms in-depth and systematically without imposing significant additional cognitive load compared to learning without them.

At this stage, the assumption made was that the participants had already obtained, from the previous part, a grand overview of unfamiliar terms contained in the given article, and thus they now needed to learn these terms in a wider context. Participants were instructed to explore the Web to identify, as much as possible, the knowledge prerequisites for each discovered term (i.e. the topics they need to learn before they become able to learn about the discovered term). To simplify the discovery of prerequisites, participants were offered a list of Web sites that provided assorted tutorials and controlled vocabularies of Web terminologies such as W3schools, Webopedia and dmoz.

Participants were then asked to invoke the SWLinker's annotation service in order to attach instructional menus to domain terms. They were directed to explore the menus in order to identify prerequisite and other instructionally-related topics.

The total time duration of this task was one hour. This duration was based on observations made during a pilot test of three participants, which showed that one hour was sufficient to compile the given article. To better fit the task within the given time, the test was stopped when the number of unfamiliar terms explored by the participant exceeded 10 such terms. The analysis of results obtained from the pilot study suggested that a larger set of terms would not make a significant difference to the results.

6.3.3.2 Questionnaire Design

The questionnaire was divided into two parts: The first part consisted of six multiple-choice questions each with a five-point scale (Very Good, Good, Average, Poor and Very Poor). The second part consisted of two written questions that aimed to explore participants' perceptions of the experiment and their suggestions regarding the tool. A copy of the questionnaire is attached in appendix B. In what follows, the questions and the rationale for them are explained.

Question statement - SWLinker provided relevant information about unknown terms in the given task.
Rationale for question – By assessing the relevance of the annotations we aimed to implicitly examine two aspects: i) the ability of the SWLinker approach to provide relevant and useful interpretation of Web content; and ii) the ability of the underlying Web mining approach to provide valid explanations for domain terms from the Web, as these explanations are the source of the annotations.

Question statement - The menus attached by SWLinker provided relevant guidance to learn about unknown terms in the given task.
Rationale for question – This question aimed to assess the extent to which the instructional associations provided through the attached menus were useful for learners in accomplishing the task requirements.

Question statement - Information and guiding menus attached by SWLinker matched my needs.
Rationale for question – This question aimed to evaluate the overall users' satisfaction with the annotation service.

Question statement - SWLinker effectively supported the learning task compared to the task without it.
Rationale for question – The aim of this question was to assess the hypothesis that the system better supports learning compared to learning without it.

Question statement - Information and guiding menus attached by SWLinker did not disturb my browsing activities.

Rationale for question – The aim of this question was to ensure that the appearance and readability of the hypertext resource were not hampered by the newly-attached components, and thus that users did not get confused or obstructed while learning with SWLinker assistance.

Question statement - SWLinker is easy to use.
Rationale for question – This question aimed to evaluate the usability of the system.

Question statement - What aspect(s) of SWLinker did you like?
Rationale for question – This question required a written answer and it aimed to get feedback about users' preferences that would help to determine how best to develop the tool to fulfil users' needs.

Question statement - Do you have any suggestions to improve SWLinker?
Rationale for question – The written answers to this question aimed to obtain further suggestions and perceptions from users that might lead to improving SWLinker.

6.3.3.3 Results and Discussion

Table 6.7 shows the results of answers to the multiple-choice questions. In what follows, the results of the questionnaire are discussed in parallel with observations made during the task execution so that the correlation between the responses to the questionnaire and the usage of the SWLinker can be highlighted.

When asked if the system provided relevant information for unknown terms included in the task, 81% of participant responses were positive, i.e., either "Very Good" or "Good". These results revealed that the explanatory information attached to the page content was sufficient for participants to proceed with the learning task without the need to resort to external resources. These results can also be used as an indicator of the efficiency of the underlying Web mining approach that supplied information used for the annotation process.

Question Statement	Very Good	Good	Average	Poor	Very Poor
SWLinker provided relevant information about unknown terms in the given task.	6	7	2	1	0
The menus attached by SWLinker provided relevant guidance to learn about unknown terms in the given task.	5	7	1	2	1
Information and guiding menus attached by SWLinker matched my needs.	4	6	3	3	0
SWLinker effectively supported the learning task compared to the task without it.	9	5	1	1	0
Information and guiding menus attached by SWLinker did not disturb browsing activities.	7	5	2	1	1
SWLinker is easy to use.	10	5	1	0	0

Table 6.7: Questionnaire results – SWLinker (Multiple choice questions)

The users' opinions were also confirmed by the analysis of the users' interactions, which showed a remarkable drop in the effort and time required to interpret the document content. While compiling the document, participants identified an average of eight unknown terms and took between 20 and 30 minutes to find relevant definitions for these terms using a search engine. Using the SWLinker's annotations, participants quickly gained a brief overview of each topic without going through the ordeal of browsing through a large number of pages returned by the search engine. This provided further evidence of the effectiveness of the annotation of terms to ease learning by bringing the desired interpretation to participants without them having to search for it. We stress here that we are neither criticizing search engines nor claiming to have provided a substitute for them. In fact, search engines are indispensable for Web users, and it is undeniable that the explanations offered by SWLinker were originally obtained from search engines. However, we can look to SWLinker as a mediator that offers a cost-effective way of gathering topic-specific information from search engines and bringing it into the Web browsing experience.

However, three participants declared that the definitions attached to some terms were not clear enough. This was because the difficulty level of some definitions was too high for novice users, and thus they need to be replaced with simpler definitions. From another perspective, it should be said that the purpose of attaching supplementary information to the page content is not necessarily to explain these terms in detail. Learners, instead, can study domain-specific terms thoroughly by referring to detailed learning resources. The purpose of the annotations is to help sustain learner engagement by giving him/her the basics to interpret the document without being repeatedly required to interrupt his/her work to seek details from external resources. For example, annotations can be helpful in a case where the unknown terms embedded in the document are not crucial to the core interests of the learner, and thus he/she needs just a general explanation of these terms in order to be able to continue reading the document.

A remarkable finding was that some of the attached definitions to the page content were exactly the same as the definitions obtained by participants when they used the search engine. This was because the attached definitions were originally extracted, by the Web mining approach, from the top ranked pages obtained from the search engine. These top ranked pages were, very often, the same sources that participants referred to when they self-dependently explored the Web. This again proves the reliability of the added annotations and their consonance with the users' expectations.

When asked about the effectiveness of the guiding menus attached by the system and whether they provide relevant guidance, the majority of students (75%) rated them very positively, i.e "Very Good" or "Good". In general, most participants shared the same opinion that the guiding menus provided a friendly way for ordinary users to learn topics in detail. However, two participants helped to reveal a drawback by raising questions about the order of the prerequisite topics provided through the attached menus. In fact, when the guiding menu shows the prerequisites for a particular topic, it does not list them in a specific order. This could mislead the learner if there are instructional dependencies between the listed prerequisites. This drawback was created in the development of the instruction path ontology because the instructional associations between terms were defined without considering cases where a term is associated with multiple prerequisite. This drawback can be resolved by using the RDF container <rdf:Seq> to list terms in order when the order is important for learning.

When asked whether the annotation and guiding menus provided by the system matched their needs, 62% of the students responded with "Very Good" or "Good", 19% responded with "Average" while 19% used the "Poor" rating. In addition, the great majority of students (88%) acknowledged that the system effectively supported the learning task compared to learning without it.

The observation of the users' browsing experience explicitly revealed difficulties encountered by participants while they tried to identify related or prerequisite topics from the Web using a search engine. Very often, participants failed to find material to provide an adequate answer for this task, except when they referred to Web resources that explicitly listed the intended prerequisites or related topics. From a certain perspective, this poor performance was due to the participants being greatly influenced by the amount of time they were required to commit, and this revealed a shortcoming in the conducted experiment. In fact, identifying related and prerequisite topics is typically an analytical process, the assessment of which requires a long-term experiment. However, this shortcoming can be resolved, to some extent, if we bear in mind that our aim was not to measure precisely the task completion time but generally to prove the time-effectiveness of using SWLinker's guiding menus compared to learning without them. This aim was considerably fulfilled and approved by participants who acknowledged the effectiveness of the attached menus in presenting the instructional relationships between topics in a short time.

75% of participants said that the layer of annotations attached by SWLinker did not disturb their browsing activities. This result demonstrates the efficiency of the visualization cues used to embed annotations within the page content without disrupting its readability or appearance.

Regarding the usability of the system, almost all participants responded positively, i.e., either "Very Good" or "Good". This result was expected due to the implementation of the client-side of the tool as a plug-in to the Internet Explorer, thus enabling users to invoke the system services by simple clicks without disrupting their normal browsing activities.

When asked about the aspect of the service that they most liked, the majority of participants replied that they liked the idea of attaching further explanations and instructions to the Web page in an intuitive manner. They stressed on the advantage of the system's efficiency to save time and effort compared to browsing without it. 5 of them particularly specified that they most liked the way in which multi-level menus could be attached to terms within the page, offering assorted paths to learn about terms in detail.

Finally, the suggestions proposed by participants to improve the system focused on ease of use. Some suggested enabling better content personalisation. For example, 4 participants suggested the need to control the colouring of the annotations and the attached menus. Three others suggested that the service would be more intuitive if the plug-in was implemented as a browser toolbar instead of as an extension to the context menu (sub-menu of the right-click menu). However, in our view, the browser tray is often over-used and often over-crowded with several toolbars.

6.3.4 Evaluation of the Co-Browsing Service in SWLinker

6.3.4.1 Overview

Evaluation of semantic relatedness measures remains an open question as there is no standard approach to use. However, previous literature reviews (Budanitsky, 1999; Blanchard et al., 2005) have identified three different approaches summarized as the following:

- **Mathematical analysis**: This approach consists of a theoretical examination of the mathematical properties of a measure, such as whether it is actually a metric, whether it has singularities, etc. Such analysis may certainly support the comparison of several measures but perhaps not so much the evaluation of individual measures.

- **Comparison with human judgment**: This approach arguably yields the most generic assessment of the validity of a measure; however, its major drawback lies in the difficulty of obtaining such judgements.

- **Application specific evaluation**: This approach is based on comparing similarity scores obtained by the measure to those obtained by using well known taxonomic hierarchies (or ontologies) such WordNet[43] and MeSH[44]. WordNet is a controlled vocabulary offering a taxonomic hierarchy of natural language terms. MeSH (Medical Subject Heading) is also a controlled vocabulary offering a hierarchical categorization of medical terms. However, to our knowledge, there is no standard vocabulary for Web technology terms that we can use for evaluation because Web terminology is rapidly evolving. In addition, lexical databases, such as WordNet, cannot be used to validate the similarity measure in our case. This is because such lexicons can only identify lexical similarity between words; i.e. synonymy, but it cannot identify the semantic relatedness between scientific terms, which often requires a domain expert who can identify how these terms are conceptually related.

In the light of the above discussion, the comparison with human judgment seems to be the most appropriate approach for validating the measure presented in this thesis. The assessment was carried out by computationally calculating the semantic relatedness between a set of Web page pairs using the mathematical approach presented in section 5.10. Subsequently, the scores were compared with those obtained from human experts on the same set of pairs. In what follows, the experiment that we conducted is presented, and then the results are revealed and discussed.

6.3.4.2 Experiment

A set 60 Web pages was selected from our knowledge base. These pages describe various domain concepts that have different degrees of relatedness. The selected pages were then set in 30 pairs (see appendix C for the full list of pairs).

Six domain experts were asked to work individually to rate the semantic relatedness between pages in every pair. The selected experts were two lecturers, two research associates and two postgraduate students in the Department of Computer Science at Durham University and all had a strong background in teaching or using most of the

[43] http://wordnet.princeton.edu/ [last accessed 22/01/2009].

[44] http://www.nlm.nih.gov/mesh/ [last accessed 22/01/2009].

Web technologies covered by the system. Having lecturers and researchers to rate the semantic relatedness between pages aimed to benefit from their instructional experience and pre-knowledge of how topics are conceptually related.

The experts were asked to rate relatedness on a five-level judgment scale from 1 to 5, where 1 indicates minimum relatedness (e.g. unrelated pages) while 5 indicates maximum relatedness (e.g. pages that explain the identical topic). Figure 6.4 depicts a sample showing how pairs of pages are presented to experts, whereas each pair is denoted by the URLs of the pages, followed by the five-point scale.

P01.	http://xml.coverpages.org/uddi.html http://webservices.xml.com/pub/a/ws/2001/04/04/webservices/index.html O 1 O 2 O 3 O 4 O 5

Figure 6.4: A sample pair of Web pages to be rated by experts for semantic relatedness

The scores obtained from the experts were then normalized to values between 0 and 1 in order to be comparable with our measure's normalized results. Afterwards, the score of relatedness between pages in every pair was computed using the SWLinker's measure. First, keywords were extracted from the pages using TF-IDF algorithm. Then, the ontology-based measure in section 5.10 was used to calculate the semantic relatedness value between the sets of keywords.

6.3.4.3 Results and Discussion

Table 6.8 shows the overall results. The first column lists the pair IDs. The second column shows the average scores obtained from the experts, while the third column lists the scores obtained from SWLinker's measure.

ID	Average Expert Rate	SWLinker
P01	0.75	0.79
P02	0.63	0.53
P03	0.63	0.72
P04	0.71	0.75
P05	0	0
P06	0.54	0.62
P07	0.54	0.38
P08	0.67	0.6
P09	0.13	0.29
P10	0.38	0.72
P11	0.46	0.22
P12	0.58	0.51
P13	0.71	0.65
P14	0.54	0.7
P15	0.58	0.54
P16	0.33	0.35
P17	0.71	0.8
P18	0.71	0.75
P19	0.71	0.71
P20	0.67	0.71
P21	0.21	0.15
P22	0.71	0.68
P23	0.75	0.78
P24	0.67	0.71
P25	0.38	0.23
P26	0.13	0.19
P27	0.42	0.24
P28	0.83	0.73
P29	0.667	0.77
P30	0.583	0.38

Table 6.8: Results of expert judgement and SWLinker's measure

SWLinker's measure can be evaluated based on the relationship between the two sets of scores. This relationship can be assessed for its strength as well as its significance as the following:

1. **The strength of the relationship**: is indicated by the correlation coefficient r , which is a statistical measure that can show whether and how strongly pairs of variables are related. The larger the correlation coefficient, the stronger the relationship is. In a series of n measurements of X and Y written

as x_i and y_i where i = 1, 2, ..., n, then the correlation of X and Y, r_{xy}, is written:

$$r_{xy} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

However, the strength of the relationship is actually measured by a related statistical measure called the coefficient of determination r^2, which is the square of the correlation coefficient. Based on results in Table 6.8, the correlation r between the average scores obtained from the experts and the scores obtained from SWLinker's measure is 0.862. In order to determine whether this value represents a strong correlation or not, we refer to the coefficient of determination r^2 which is 74.3%. This means that 74.3% of the variability in the SWLinker's scores can be explained by the fact that they are related to the average scores obtained from experts. In the field of semantic similarity, a coefficient of determination greater than 64% is generally described as strong, whereas a coefficient of determination less than 25% is generally described as weak (Liu et al., 2007; Couto et al., 2007). Thus, a value of 74.3% shows a strong correlation of the SWLinker's rates with the experts' rates.

2. **The significance of the relationship**: determines the probability that this correlation could have occurred by chance. Note that the relationship can be strong and yet not significant. Conversely, a relationship can be weak but significant. The key factor is the size of the sample. For small samples, it is easy to produce a strong correlation by chance, while it is not easy to get a strong correlation by chance for large samples. To determine whether the correlation of 0.862 represents a significant value, we refer to the statistical table 6.9 (Robson, 1983) which lists the smallest values of correlation coefficient significantly different from zero at a probability level of 0.05 for different values of N, where N is the number of pairs of scores. Since 30 pairs of scores were used in our experiment (N = 30), the value for the correlation coefficient should be as high as 0.34 in order to be significant. Thus, a

correlation of 0.862 with N=30 shows a significance relationship that is unlikely to occur by chance.

N	Correlation Coefficient	N	Correlation Coefficient
5	0.67	16	0.44
6	0.63	17	0.43
7	0.60	18	0.42
8	0.58	19	0.41
9	0.55	20	0.40
10	0.53	25	0.37
11	0.51	26	0.36
12	0.50	27	0.36
13	0.48	28	0.35
14	0.47	29	0.34
15	0.46	30	0.34

Table 6.9: Significance of correlation coefficients (Robson, 1983)

Although the above discussion has shown that the correlation between the average expert scores and our measure's scores is both strong and significant, one should note the accuracy of the SWLinker's measure is proportional to the instructional view represented in the ontology. The closer the ontology representation is to the experts' view and the less ambiguous is the relatedness between pages, the more the results of our measure approximate the expert judgements.

In addition, the comparison with expert judgement has demonstrated the efficiency of the matching approach to identify instructionally-related Web pages. For example, despite the fact that some Web pages had almost wholly different keywords (refer to P06, P12 and P22 as examples), SWLinker's measure identified a strong relatedness between them, and this approximated the average scores obtained from the experts on the same pairs. These pages would not have been considered to be highly related if examined using similarity measures that are based on direct content or keyword matching approaches. Thus, our approach provides more appropriate results for the field of learning where the instructional relationships take priority over any other metrics in order to determine similarity between documents.

Although the comparison with expert judgment helped evaluate the SWLinker's underlying measure for semantic relatedness, its main drawback is the lack of an application-specific approach for assessment that can explicitly show the learning benefits and outcomes of applying the measure within a realistic learning task.

6.4 Summary and Concluding Remarks

A combination of quantitative and qualitative methods was used to evaluate the Knowledge Puzzle tool. Firstly, the students' navigation behaviours were recorded and analysed. The aim was to quantitatively compare the use of the navigation aids uniquely offered by the tool against those offered by most traditional browsers. The results showed an obvious tendency to use the new techniques over traditional ones. In addition, the results revealed the ability of the tool to draw the user's attention to the primary exploration processes. This was deduced from the increased revisits per page ratio for pages especially referenced from the constructed map. The limitation of this method was the limited size of the learning task, which was not sufficient to assess the value of hyperlinks and annotations attached by the tool over Web pages. This value can only be exposed when the complexity of the map becomes unmanageable and untraceable. However, the inspection of individual navigation behaviours considerably resolved this limitation by helping to identify an explicit relationship between the size of the constructed map and the use of the attached hyperlinks.

The learning outcomes were assessed by analyzing the knowledge maps submitted by the students. These maps provided evidence of how successful the students were in using the tool to reference information and to link nodes with expressive relationships. In addition, the common problems in the constructed maps helped us to identify some shortcomings of the tool and to inspire future improvements.

Data collected from the Knowledge Puzzle questionnaire was used to assess the students' satisfaction with the tool. Students confirmed the ability of the tool to link separate bits of information on the Web and reduce the cognitive effort required to retain collected information. On the other hand, the students' rating of the usability of the tool and the ease of navigation was low. This was, to some extent, expected due to their lack of familiarity with the tool and to implementation shortcomings.

In order to evaluate the research ideas included in the SWLinker, an evaluation framework was implemented and evaluated during a user trial. The user study looked into the effectiveness of the system in supporting the user's task, to what extent learning was facilitated compared to learning without the system, how the annotation of Web pages influenced the browsing activities, the extent of overall user satisfaction and, finally, how easy it was to learn to use the system. Data was collected by means of a user questionnaire and recording the users' interactions with the system. Results indicated that the system showed potential to have a high level of effectiveness and a better overall user satisfaction. The users' response as well as the observation of the task execution showed that the contribution of bringing up supplementary information and attaching it to the page content using interactive visualization cues resulted in a substantial drop in the time and effort required to carry out the task.

The co-browsing service in SWLinker utilizes a semantic relatedness measure in order to match users with related browsing experiences. The quality of the matching results is conditional on the validity of the underlying measure. Thus, the evaluation of the co-browsing service was based on validating the semantic relatedness measure, and this indicated the efficiency of the SWLinker's co-browsing approach. The assessment was carried out by computationally calculating the semantic relatedness between a set of Web page pairs. Subsequently, the scores were compared with those obtained from human experts on the same sets of pairs. The results showed that the relatedness values obtained from our measure were highly correlated with the expert judgement.

Finally, the students' comments and suggestions on the two systems helped us to understand more about the systems' strengths and weaknesses. In general, they rated positively the support that the systems offered in terms of breaking the barriers of the 'static' nature of hypertext by making the learning process more interactive, more responsive and more active. Their comments also suggested possibilities for improving the systems either by making them more reliable and intuitive, or by developing more personalised learning services.

Chapter 7
Conclusion and Future Work

7.1 Introduction

This chapter reviews the research presented in this thesis and summarises its achievements based on the criteria for success defined in chapter 1. It also discusses its general research contribution to the field of Web-based learning. Finally, suggestions are made for the direction of future work.

7.2 Thesis Summary

This thesis focused on presenting technologies to improve SDL, particularly from hypertext resources. The ultimate goal was to change the way in which traditional hypertext learning resources are seen as static resources that lack any great degree of interactivity and interpretation.

Aiming at narrowing the gap between what exists on the Web and what resides in the learner's mind, the Knowledge Puzzle tool has been proposed. The tool benefits from the theory of cognitive constructivism in order to stimulate learners to visualize the navigational behaviour and semantic processing of Web content using a meta-cognitive tool. Once this phase is complete, the constructed map is converted to a layer of expressive hyperlinks and annotations over Web pages, causing the knowledge structure on the Web to more accurately represent the knowledge structure in the learner's mind. Finally, a hypertext version of the whole constructed knowledge is produced to enable fast and easy reviewing.

Generally speaking, the use of the Knowledge Puzzle tool incorporates each of the intellectual skills listed in the Bloom's taxonomy, starting from gathering information and planning the navigation paths (knowledge recall, comprehension), to deciding what and how Web content should be interlinked and annotated (application, analysis) and "fitting" the information structure on Web to the structure of the learner's knowledge (synthesis), ending with publishing the whole knowledge as a single entity file in order to facilitate self-assessment and reviewing (evaluate). This considerably addresses the issue raised at the beginning of this thesis in section 1.2, which is namely how to enable learners to use static hypertext resources, which provide only passive access to information, to progress to higher cognitive activities such as applying, analysing and evaluating the knowledge they glean.

The adaptation strategy offered by the Knowledge Puzzle tool differs from various other adaptive hypermedia techniques by being entirely user driven, and being without any restrictions, limitations or preconditions. The contribution of the tool resides in the proposed constructivist approach for SDL that enables learners to operate actively by manipulating, interlinking and annotating static hypertext learning resources on the Web. This contribution achieved the criteria for success defined for this research:

"***Support knowledge construction from the Web: This criterion means that the proposed tool for knowledge construction should enable learners to effectively manage structure and rethink the knowledge they gain from the Web with the least effort***".

The evaluation of the Knowledge Puzzle tool in the classroom using a set of quantitative and qualitative measures successfully revealed the superiority of the proposed approach over conventional navigation cues in enhancing navigational learning. The experiment that was conducted also showed a significant drop in instances of "lost in hyperspace" due to the ability of the tool to draw the learner's attentions to the primary exploration processes. In addition, participants greatly appreciated the ability of the tool to link Web resources and to plan navigation paths that significantly reduced the cognitive effort required for knowledge retention and processing. These results show the accomplishment of the second criteria for success:

"***Reduce the impact of problems associated with navigation in hyperspace: since the focus is on learning from hypertext resources, this criterion implies that the proposed approach should mitigate, as much as possible, the effects of***

disorientation and cognitive overload, the two main problems that learners experience when trying to navigate within hypertext systems".

In order to resolve the learning difficulties discussed in section 1.3, SWLinker was proposed as an effective approach to bringing semantics into the Web browsing experience at both individual and collaborative levels. At the individual level, SWLinker acts as a complementary knowledge source, which a learner can call on demand to gain instantaneous access to the background knowledge relevant to a Web resource. By annotating Web pages while they are being browsed, learners can get the information they need to interpret and comprehend the content without leaving the original site and going elsewhere to find that information. Thus, they will be more attentive, better engaged in the learning process and more motivated to complete the learning goal. This achieved the third criteria for success:

"Enhance and sustain learner engagement during self-directed learning: the on-demand annotation of Web pages should keep learners better engaged and more highly motivated by reducing the need to suspend the learning task in order to seek help".

SWLinker differs from older similar systems for annotating Web resources in that it provides detailed learning guidance through the menus attached by the tool over domain terms included in the page. These menus act as internal portals that allow users to explore the entire learning path through each term without additional effort and without interrupting their learning process to seek assistance. The observation of the system usage during the experiment and the responses to the user questionnaire demonstrated the effectiveness of the attached menus in enabling users to gain more complete knowledge. This fulfilled the fourth criteria for success:

"Support in-depth learning while browsing learning resources on the Web: this criterion will assess if the annotation service helps learners gain more complete and in-depth knowledge while they browse the Web".

SWLinker is also distinguished by the fact that it takes into account and addresses the evolution and maintenance of knowledge. The adoption of a standard specification for ontology development and the use of semi-automatic Web mining to feed the knowledge base make the maintenance and evolution of system less troublesome.

At the collaborative level, SWLinker aims to break the conventional view of Web browsing as an individual activity by leveraging the notion of collaborative browsing. The novelty of this approach lies in the fact that it utilises not only the similarity between users' navigational behaviours but also the semantic relatedness between them, and this leads to more intelligent matching results. The ability to bring users with mutual interests together relies of the effectiveness of underlying ontology-based approach to calculate the degree of relatedness between Web pages. This effectiveness has been experimentally demonstrated by comparing expert judgment on sample pairs of pages with the scores obtained from the ontology-based approach. Such comparison fulfilled the fifth criteria for success:

"Improve collaboration by bringing together users with related interests: this criterion will assess if the user matching algorithm adopted in the co-browsing service can effectively match and interlink users based on their semantically related browsing activities".

Our conclusion is that SWLinker can effectively showcase the numerous benefits of the Semantic Web technologies in people's everyday activities.

7.3 Future Work

Even though the technologies presented in this thesis have considerably achieved their intended goals, there are many potential extensions that can enhance SDL from hypertext resources. These extensions are as follows.

7.3.1 Improved Knowledge Representation

The experimental use of the Knowledge Puzzle tool demonstrated the success of this approach to adapting the Web information structure with the self-constructed knowledge. However, the desired adaptation depends on how well the students can represent and visualize their knowledge in terms of a concept map. The evaluation phase of the Knowledge Puzzle tool revealed that the current support it offers for knowledge representation does not meet all the learners' expectations. While the tool enables basic annotation and linking of pages and segments of information with objective relationships, other operations are still required to make the constructed map more expressive and reflective. For instance, learners very often need to group a number of graph nodes to denote a shared relationship. They may also need to create nodes that represent abstract concepts rather than pages, and then to make relationships between concepts, or between concepts and pages. Another potential enhancement is to enable the tool to offer multiple representations of knowledge, such as distinguishing between hierarchal and pedagogical relationships between graph nodes, as discussed in section 6.2.7.

Despite the importance of these suggestions for enhancing knowledge representation, the main challenge lies in translating these enhancements into useful additions inside the hypertext of Web pages in order to achieve the desired adaptation. Such translation should be considered carefully and should be based upon strong pedagogical foundations to avoid misconceptions or complications that may complicate learning instead of simplifying it. They also require further research into several cognitive and psychological issues related to learning and their effects on hypertext comprehension.

7.3.2 Knowledge Sharing

Knowledge construction from the Web opens up a new research direction, namely the possibility of sharing knowledge obtained from the SDL process. Plenty of Web technologies, such as Internet forums and social networking techniques, enable users to share and discuss their views collaboratively. Other technologies, such as social bookmarking, enable people to save and share Web resources that they feel are useful or interesting. However, in our opinion, it could be more valuable and intuitive for users to save and share their perceptions and views alongside the Web resources and navigation paths that fulfil their goals. Why don't we think of Web pages as annotation boards that can be interlinked, highlighted and annotated with self-perceptions and then shared to reflect certain interests and/or sustain particular views? Educationally, this type of knowledge sharing has many advantages:

1. It could be used as an interactive teaching activity over the Web. The teacher could reuse existing learning material on the Web, under licensed conditions, and then annotate it with comments, hyperlinks and any further illustrations without the need to have actual ownership of the material. In addition, various versions of the same material, but with different attached components, could

be posted to adapt to new situations and/or the heterogeneous needs of learners. This could provide an effective remedy for the traditional "one-size-fits-all" approach.

2. The ability to attach comments and hyperlinks to existing pages could give a strong boost to collaboration over the Web. Learners who can share their views alongside the navigation paths would give them an opportunity to get reviews from peers who are more experienced in the subject under discussion. By tracing posted navigation paths and any attached components, peers could easily understand what learners are trying to say and what potential mistakes they have made while learning from the Web. In addition, the shared navigation paths would enable users to learn from others' experiences and to discover new knowledge resources related to subjects of interest.

In our future work, however, several research questions need to be addressed in order to achieve this level of knowledge sharing over the Web. These include:

- How should the knowledge be represented in order to achieve a reasonable level of interoperability among heterogeneous Web applications?
- How can the shared knowledge be reused by backend users in such a way that it fulfils the desired goals without disrupting learning?
- To what extent the proposed ideas fit various learning/teaching strategies and circumstances?

7.3.3 Personalized Semantic Annotation

Users, in general, are not guided during Web browsing. Although the annotation of Web pages with complementary information and guiding menus can go a long way in offering the intended guidance, so far little attention has been paid to personalization issues. Annotations offered by SWLinker will be the same for all users regardless of the different interests and goals of Web users. While these annotations may be useful to some users, they may not satisfy the requirements of others who are looking for either more advanced or less complicated information. Further research is required to investigate how the annotation of Web pages can be personalized to cope with individual needs. For example, particular annotations or semantic links could be recommended by the system according to the user's goals and interests, which could be deduced from his/her user-profile and/or browsing activities. In addition, the level of guidance offered through the attached menus could be adapted according to the expertise of the user. For instance, when a term is annotated by a novice user, the attached menu could provide links to introductory resources while detailed semantic links are provided to the expert user.

However, such a level of personalized annotation will require special mechanisms of personalization and user-modelling. Existing personalization mechanisms on the Web require users to log in to multiple Websites and the user's profile is likely to change from site to site. There is a need for generic user-profile and personalization architectures, which can achieve the desired adaptation on diverse Websites. Our hypothesis is that Semantic Web technologies can offer the solution to these problems. Ontology-based user-profiles are interoperable, and they can be easily extended and combined with semantic metadata on the Web. In our future work, we will attempt to extend SWLinker to be a personalized Semantic Web browser, which can be used to support browsing by users using semantic and adaptive links. This intended development will be based on the integration of ontology-driven user-modelling into the Semantic Web browser in order to personalize the annotation of Web pages and thus to offer a service that corresponds to the users' browsing preferences and needs.

7.4 Conclusion

The field of learning from hypertext and hypermedia is vast and this thesis represents a step towards more profound research in the educational domain. Conceptually, the research reported in this thesis has emphasized the importance of fulfilling users' needs and designing user-centric tools that move towards more interactive learning without introducing high cognitive loads in terms of usability and browsing effort. This thesis has also contributed to the endeavour of incorporating constructivism and semantics with hypertext as ways to improve learning. Finally, though this research has considerably achieved its intended goals, there are still many research questions that need to be answered. These questions include, but are not limited to, the following:

- To what extent can the proposed technologies be successfully integrated into traditional browsing tools?
- How do the ongoing advances in Web design affect the applicability of the proposed technologies?
- How can the potential usability concerns of the reported technologies be resolved?

It is hoped that such questions will help to shape the future directions of researchers interested in facilitating self-directed learning from hypertext resources.

Appendix A

The Knowledge Puzzle Tool's Questionnaire

With respect to the Knowledge Puzzle tool which you are asked to use to do your assignment, please respond to ALL of the following items by selecting the response that best matches your opinion:

1. The Knowledge Puzzle tool helped me link separate pieces of information on the Web.
   O Strongly agree O Agree O Disagree O Strongly disagree

2. The Knowledge Puzzle tool helped me navigate a path effectively during the learning task.
   O Strongly agree O Agree O Disagree O Strongly disagree

3. The Knowledge Puzzle tool helped me directly access pieces of information inside the Web pages.
   O Strongly agree O Agree O Disagree O Strongly disagree

4. The new type of links attached to the Web pages by the Knowledge Puzzle tool helped me navigate through the Web pages.
   O Strongly agree O Agree O Disagree O Strongly disagree

5. The Knowledge Puzzle tool effectively reduced the amount I need to remember.
   O Strongly agree O Agree O Disagree O Strongly disagree

6. Grouping and structuring information components in a single page helped me quickly review the knowledge gained from the Web.
   O Strongly agree O Agree O Disagree O Strongly disagree

7. The Knowledge Puzzle tool is easy to use.
   O Strongly agree O Agree O Disagree O Strongly disagree

8. The user interface of the Knowledge Puzzle tool is friendly.
   O Strongly agree O Agree O Disagree O Strongly disagree

9. What single aspect most supported your learning?

10. Do you have any suggestions to improve the Knowledge Puzzle tool?

Thank you

Appendix B

SWLinker's Questionnaire

With respect to the task which you are asked to complete, please respond to ALL of the following items by selecting the response that best matches your opinion:

1. SWLinker provided relevant information about unknown terms in the given task.
   O Very Good O Good O Average O Poor O Very Poor

2. The menus attached by SWLinker provided relevant guidance to learn about unknown terms in the given task.
   O Very Good O Good O Average O Poor O Very Poor

3. Information and guiding menus attached by SWLinker matched my needs.
   O Very Good O Good O Average O Poor O Very Poor

4. SWLinker effectively supported the learning task compared to the task without it.
   O Very Good O Good O Average O Poor O Very Poor

5. Information and guiding menus attached by SWLinker did not disturb browsing activities.
   O Very Good O Good O Average O Poor O Very Poor

6. SWLinker is easy to use.
   O Very Good O Good O Average O Poor O Very Poor

7. What particular aspect(s) of SWLinker did you *like*?

8. Do you have any suggestions to improve the system?

Thank you

Appendix C

Expert Judgement on the Semantic Relatedness between Web pages

One a scale from 1 to 5, please rate the degree of relatedness between the Web pages in the following pairs. Please take the following points into consideration:

- 1 on the scale means that the Web pages are completely **unrelated** (i.e. they explain unrelated topics). This is the least possible relatedness value.
- 5 on the scale means that the pages explain **the same identical topic**. This is the most possible relatedness value.
- The degree of relatedness in between is based on **the extent to which the topics of the pages are related** or **how important the pages are to each other**. For example, pages are considered highly related if the topic of the first is a sub topic of the second's, if they depend on each other (i.e one is prerequisite for the other) or if they share the same parent topic. On the other hand, pages are considered slightly related if they share a single or few ideas, and their subject matters are not prerequisites (e.g. it is not necessary to study one to be able to study the other).

P01. http://xml.coverpages.org/uddi.html
http://webservices.xml.com/pub/a/ws/2001/04/04/webservices/index.html

O 1 O 2 O 3 O 4 O 5

P02. http://en.wikipedia.org/wiki/XML
http://webservices.xml.com/pub/a/ws/2001/04/04/webservices/index.html

O 1 O 2 O 3 O 4 O 5

P03. http://en.wikipedia.org/wiki/Representational_State_Transfer
http://webservices.xml.com/pub/a/ws/2001/04/04/webservices/index.html

O 1 O 2 O 3 O 4 O 5

P04. http://en.wikipedia.org/wiki/SPARQL
http://infomesh.net/2001/swintro/

O 1 O 2 O 3 O 4 O 5

P05. http://www.walthowe.com/navnet/faq/telnet.html
http://infomesh.net/2001/swintro/

O 1 O 2 O 3 O 4 O 5

P06. http://www-ksl.stanford.edu/kst/what-is-an-ontology.html
http://infomesh.net/2001/swintro/

O 1 O 2 O 3 O 4 O 5

P07. http://www.irisa.fr/lande/genet/crypto.html
http://www.instantssl.com/ssl-certificate-support/we_care.html

O 1 O 2 O 3 O 4 O 5

P08. http://www.irisa.fr/lande/genet/crypto.html
http://www.securityfocus.com/infocus/1181

O 1 O 2 O 3 O 4 O 5

P09. http://www.irisa.fr/lande/genet/crypto.html
http://www.webopedia.com/DidYouKnow/Internet/2005/phishing.asp

O 1 O 2 O 3 O 4 O 5

P10. http://en.wikipedia.org/wiki/JavaServer_Pages
http://www.w3schools.com/php/default.asp

O 1 O 2 O 3 O 4 O 5

P11. http://en.wikipedia.org/wiki/JavaServer_Pages
http://www.w3schools.com/js/default.asp

O 1 O 2 O 3 O 4 O 5

P12. http://en.wikipedia.org/wiki/XML
http://www.answers.com/topic/html

O 1 O 2 O 3 O 4 O 5

P13. http://en.wikipedia.org/wiki/XML
http://www.w3.org/TR/rdf-primer/

O 1 O 2 O 3 O 4 O 5

P14. http://www.w3.org/DOM/
http://en.wikipedia.org/wiki/Simple_API_for_XML

O 1 O 2 O 3 O 4 O 5

P15. http://www.w3.org/DOM/
http://en.wikipedia.org/wiki/XML_schema

O 1 O 2 O 3 O 4 O 5

P16. http://www.w3.org/DOM/
http://www.w3.org/TR/rdf-primer/

O 1 O 2 O 3 O 4 O 5

P17. http://en.wikipedia.org/wiki/Digital_subscriber_line
http://sdsl.ca/

O 1 O 2 O 3 O 4 O 5

P18. http://compnetworking.about.com/od/networkdesign/a/topologies.htm
http://www.answers.com/topic/local-area-network

O 1 O 2 O 3 O 4 O 5

P19. http://www.answers.com/topic/local-area-network
http://www.cisco.com/en/US/docs/internetworking/technology/handbook/Intro-to-WAN.html

O 1 O 2 O 3 O 4 O 5

P20. http://www.answers.com/topic/local-area-network
http://en.wikipedia.org/wiki/Ethernet

O 1 O 2 O 3 O 4 O 5

P21. http://www.xml.com/pub/a/2002/12/18/dive-into-xml.html
http://www.adaptivepath.com/ideas/essays/archives/000385.php

O 1 O 2 O 3 O 4 O 5

P22. http://www.w3schools.com/js/default.asp
http://www.adaptivepath.com/ideas/essays/archives/000385.php

O 1 O 2 O 3 O 4 O 5

P23. http://en.wikipedia.org/wiki/XMLHttpRequest
http://www.adaptivepath.com/ideas/essays/archives/000385.php

O 1 O 2 O 3 O 4 O 5

P24. http://www.irchelp.org/irchelp/security/trojan.html
http://www.cknow.com/vtutor/IntroductiontoViruses.html

O 1 O 2 O 3 O 4 O 5

P25. http://www.irchelp.org/irchelp/security/trojan.html http://www.webopedia.com/DidYouKnow/Internet/2005/phishing.asp

O 1 O 2 O 3 O 4 O 5

P26. http://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol
http://www.irchelp.org/irchelp/security/trojan.html

O 1 O 2 O 3 O 4 O 5

P27. http://en.wikipedia.org/wiki/Internet_Protocol
http://www.webopedia.com/TERM/h/hub.html

O 1 O 2 O 3 O 4 O 5

P28. http://en.wikipedia.org/wiki/Internet_Protocol
http://www.answers.com/topic/ipv6

O 1 O 2 O 3 O 4 O 5

P29. http://en.wikipedia.org/wiki/Internet_Protocol
http://en.wikipedia.org/wiki/Domain_Name_System

O 1 O 2 O 3 O 4 O 5

P30. http://en.wikipedia.org/wiki/Internet_Protocol
http://www.answers.com/topic/intranet

O 1 O 2 O 3 O 4 O 5

Thanks

References

ADL SCORM, (2004), Sharable Content Object Reference Model (SCORM), 2nd edition Overview, Retrieved November 15, 2008 from http://www.adlnet.org/scorm/history/2004/index.cfm

Antoniou, G., and van Harmelen, F., (2004), A Semantic Web Primer, *The MIT Press.*

Aroyo, L., and Dicheva, D., (2004a), Authoring Support in Concept-based Web Information Systems for Educational Applications, *International Journal of Continuous Engineering Education and Lifelong Learning* 14(3): pp. 297-312.

Aroyo, L., and Dicheva, D., (2004b), The New Challenges for E-learning: The Educational Semantic Web, *Educational Technology and Society* 7(4): pp. 59-69.

Azouaou, F., Chen, W., and Desmoulins, C., (2004), Semantic Annotation Tools for Learning Material, in: *The International Workshop on Applications of Semantic Web Technologies for E-Learning)*, Winston Salem, USA.

Azouaou, F. and Desmoulins, C., (2006a), Using and Modeling Context with Ontology in e-learning: the Case of Personal Annotations, in: *The International Workshop on Applications of Semantic Web Technologies for e-Learning*, Dublin, Ireland.

Azouaou, F. and Desmoulins, C., (2006b), Teachers' document annotating: Models for a digital memory tool, *The International Journal of Continuing Engineering Education and Life-Long Learning* 16(1/2): pp. 18-34.

Azouaou, F. and Desmoulins, C., (2005), Semantic Annotation for the Teacher: Models for a Computerized Memory Tool, in: *The International Workshop on Applications of Semantic Web Technologies for e-Learning*, Amsterdam, The Netherlands.

Barr, A. and Feigenbaum, E.A., (1982), Planning and Problem Solving, *Handbook of Artificial Intelligence*, Anonymous Pitman.

Bayerl, P. S., Lüngen, H., Gut, U. and Paul, K. I., (2003), Methodology for Reliable Schema Development and Evaluation of Manual Annotations, *Workshop Notes for the Workshop on Knowledge Markup and Semantic Annotation, The 2nd International Conference on Knowledge Capture*, Sanibel, Florida, USA, pp. 17-23.

Bechhofer, S., Yesilada, Y., Stevens, R., Jupp, S., and Horan, B., (2008), Using Ontologies and Vocabularies for Dynamic Linking, *IEEE Internet Computing*, 12(3): pp. 32-39.

Berners-Lee, T., Hendler, J., and Lassila, O., (2001), The Semantic Web, *Scientific American*, 284(5): pp. 34-43.

Bigus, J., (2004), Reflection on Behaviorism, Cognitive Information Processing, and Constructivism in Distance Learning, Retrieved April 9, 2009 from http://www.bigusbooks.com/mastersportfolio/products/ReflectionsOnLearningTheoriesAndDL.pdf

Blanchard, E., Harzallah, M., Briand, H., and Kuntz, P., (2005), A typology of Ontology-Based Semantic Measures, *Open Interop Workshop on Enterpise Modelling and Ontologies for Interoperability.*

Bloom, B. S., (1956), Taxonomy of Educational Objectives, *the classification of educational goals – Handbook I: Cognitive Domain*, New York: McKay.

Brase, J., and Painter, M., (2004), Inferring Metadata for a Semantic Web Peer-to-Peer Environment, *Educational Technology and Society* 7 (2): pp. 61-67.

Brickley, D., and Guha, R.V., (2004), RDF Vocabulary Description Language 1.0: RDF Schema, W3C Working Draft, Retrieved November 15, 2008 from http://www.w3.org/TR/PR-rdf-schema.

Brin, S., (1998), Extracting Patterns and Relations from the World Wide Web, *WebDB Workshop at 6th International Conference on Extending Database Technology.*

Brown, A. L., (1987), Metacognition, Executive Control, Self-Regulation, and Other More Mysterious Mechanisms, In F.E. Weinert and R.H. Kluwe (Eds.), *Metacognition, motivation, and understanding*, Hillsdale, NJ, Erlbaum, pp. 65-116.

Brown, J.S., (2000), Growing Up Digital: How the Web Changes Work, *Education and the Ways People Learn*, Retrieved November 15, 2008 from http://www.johnseelybrown.com/Growing_up_digital.pdf

Bruijn, J. d. and Fensel, D., (2005), Ontology Definitions, *Encyclopedia of Library and Information Science, Marcel Dekker, Inc.*

Brusilovsky, P., (1999), Adaptive and intelligent technologies for Web-based education, *KUnstliche Intelligenz*, 4: pp. 19-25.

Brusilovsky, P., (1996), Methods and Techniques of Adaptive Hypermedia, *User Modelling and User-Adapted Interaction*, 6: pp. 87-129.

Budanitsky, A., (1999), Lexical semantic relatedness and its application in natural language processing, *Technical report, Computer Systems Research Group, University of Toronto.*

Budzik, J., Bradshaw, S., Fu, X., and Hammond, K., (2002), Supporting Online Resource Discovery in the Context of Ongoing Tasks with Proactive Software Assistants, *International Journal of Human-Computer Studies*, 56(1): pp. 47-74.

Carroll, J.M., Mack, R.L., Lewis, C.H., Grischkowsky, N.L., and Robertson, S.R., (1985), Exploring exploring a word processor, *Human-Computer Interaction*, 1(3): pp. 283-307.

Cennamo, K. S., Ross, J. D., and Rogers, C. S., (2002), Evolution of a Web-Enhanced Course: Incorporating Strategies for Self-Regulation, *Educause Quarterly*, 25(1): pp. 28–33.

Chandler, P., and Sweller, J., (1991), Cognitive Load Theory and the Format of Instruction, *Cognition and Instruction*, 8(4): pp. 293-240.

Chandler, P., and Sweller, J., (1992), The Split-Attention Effect as a Factor in the Design of Instruction, *British Journal of Educational Psychology*, 62(2): pp. 233-246.

Chandrasekaran, B., Josephson, J.R., and Benjamins, V.R., (1999), What are ontologies, and why do we need them?, *IEEE Intelligent Systems*,14(1): pp. 20-26.

Chang, M. M., (2005), Applying Self-Regulated Learning Strategies in a Web-Based Instruction - an Investigation of Motivation Perception, *Computer Assisted Language Learning*, 18(3): pp. 217-230.

Charney, D.H. and Reder, L.M., (1986), Designing Interactive Tutorials for Computer Users, *Human-Computer Interaction*, 2: pp. 297-317.

Chen, W., Hayashi, Y., Kin, L., Ikeda, M., and Mizoguchi, R., (1998), Ontological Issues in an Intelligent Authoring Tool, in Chan T-W., Collins A. and Lin J. (Eds.), *In Proceedings of The International Conference for Computers in Education*, pp.41-50.

Chou, C., (1999), Developing Hypertext-Based Learning Courseware for Computer Networks: The Macro and Micro Stages, *IEEE Transactions on Education,* 42 (1): pp. 39-44.

Chularut, P. and DeBacker, T.K., (2004), The influence of Concept Mapping on Achievement, Self-Regulation, and Self-efficacy in Students of English as a Second language, *Contemporary Educational Psychology*, 29: pp. 248-263.

Cockburn, A., Greenberg, S., (2000), Issues of Page Representation and Organisation in Web browser's Revisitation Tools, *Australian Journal of Information Systems*, 7(2): pp. 120-127.

Conklin, J., (1987), Hypertext: an introduction and survey, *IEEE Computer Society Press*, 20(9): pp. 17-41.

Couto, F. M., Silva, M. J., and Coutinho, P. M., (2007), Measuring Semantic Similarity between Gene Ontology Terms, *Data and Knowledge Engineering*, 61(1): pp.137-152.

Cunningham, H., Maynard, D., Bontcheva, K. and Tablan, V., (2002), GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, *In 40th Anniversary Meeting of the Association for Computational Linguistics*.

Danielson, D., (2002), Web Navigation and the Behavioral Effects of Constantly Visible Site Maps, *Interacting with Computers*, 14: pp. 601–618.

Devedzic, V., (2006), Semantic Web and Education, *Springer's Integrated Series in Information Systems*.

Dias, P., Gomes, M., and Correia, A., (1999), Disorientation in hypermedia environments: Mechanisms to support navigation, *Journal of Educational Computing Research*, 20(2): pp. 93-117.

Dill, S., Eiron, N., Gibson, D.,Gruhl, D., Guha, R., Jhingran, A., Kanungo, T., McCurley, K. S., Rajagopalan, S., Tomkins, A., Tomlin, J. A., and Zienberer, J. Y., (2003), A Case for Automated Large Scale Semantic Annotation, *Journal of Semantic Web*, 1(1): pp. 115-132.

Ding, Y., (2005), Study of Design Issues on an Automated Semantic Annotation System, *AIS SIGSEMIS Bulletin*, 2: pp. 45-51.

Djoudi, M., (1999), Navir, un système d'aide à la navigation virtuelle sur le Web, *Deuxièmes Entretiens Internationaux du CNED, Poitiers.*

Domingue, J.B., Dzbor, M., and Motta, E., (2004), Collaborative Semantic Web Browsing with Magpie, *In Proceedings of the 1st European Semantic Web Symposium*, Greece.

Duffy, T. M. and Cunningham, D. J., (1996), Constructivism: Implications for the Design and Delivery of Instruction, *In D. H. Jonassen (Ed.), Educational communications and technology*, New York: Simon and Schuster Macmillan, pp. 170-199.

Duffy, T. M. and Jonassen, D.H., (1992), Constructivism and The Technology of Instruction: A Conversation, *Lawrence Erlbaum Assoc. Inc*, New Jersey.

Dzbor, M., Motta, E., and Domingue J. B., (2004), Opening Up Magpie via Semantic Services, *In Proceedings of the 3rd International Semantic Web Conference*, Japan.

Elm, W. C. and Woods, D. D., (1985), Getting Lost: A Case Study in Interface Design, *In Proceedings of Human Factors Society, 29th Annual Meeting*, Santa Monica, pp. 927-931.

Ertner, P. A. and Newby, T. J., (1993), Behaviorism, Cognitivism, Constructivism: Comparing Critical Features from an Instructional Design Perspective, *Performance Improvement Quarterly*, 6 (4): pp. 50-72.

Esenther, A. W., (2002), Instant Co-Browsing: Lightweight Real-time Collaborative Web Browsing, *In Proceedings of the 11th International World Wide Web Conference, Hawaii*, pp. 107-114.

Esnault, L., and Zeiliger, R., (1999), Learning, Teaching and Browsing with NESTOR, *In Proceedings of IRMA Conference*, Hershey, Pe., USA.

Fensel, D. and Musen, M.A., (2001), The Semantic Web: A brain for humankind, *IEEE Intelligent Systems,* 16(2): pp. 24-25.

Fischer, G., (1998), Beyond 'Couch Potatoes': From Consumers to Designers, *Paper In proceedings of the 3rd Asia Pacific Computer Human Interaction Conference*, Kanagawa, Japan.

Flavell, J. H., (1987), Speculations About the Nature and Development of Metacognition, *Hillside, NJ: Lawrence Erlbaum.*

Flavell, J. H., (1976), Metacognitive Aspects of Problem Solving, *In L. B. Resnick (Ed.), The nature of intelligence. Hillsdale, NJ: Erlbaum*, pp. 231-236.

Gerjets, P. and Scheiter, K., (2003), Goal Configurations and Processing Strategies as Moderators between Instructional Design and Cognitive Load: Evidence from Hypertext-Based Instruction, *Educational Psychologist*, 38(1): pp. 33-41.

Gómez-Pérez, A. and Manzano-Macho, D., (2004), An Overview of Methods and Tools for Ontology Learning from Texts, *The Knowledge Engineering Review*, 19 (3): pp. 187-212.

Good, T. L., and Brophy, J. E., (1990), Educational Psychology: A Realistic Approach, 4th edition, *White Plains, NY: Longman.*

Gros, B., (2002), Knowledge Construction and Technology, *Journal of Educational Multimedia and Hypermedia*, 11(4): pp. 323-343.

Gruber, T.R., (1993), A translation approach to portable ontology specifications, *Knowledge Acquisition*, 5: pp. 199-220.

Guarino, N., (1995), Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human-Computer Studies*, 43(5/6): pp. 625-640.

Guglielmino, L. M. and Guglielmino, P. J., (2003), Are Your Learners Ready for E-learning? In G. M. Piskurich (Ed.), *The AMA handbook of e-learning: Effective design, implementation, and technology solutions*, New York: AMACOM, pp.87-95.

Hadwin, A. F., Winne, P. H., Nesbit, J.C., and Kumar, V., (2005), Tracing self-regulated learning in the Learning Kit, Paper presented at the annual meeting of the American Educational Research Association, Montreal , QC, Canada.

Hamers, L., Hemeryck, Y., Herweyers, G., Janssen, M., Keters, H., Rousseau, R., and Vanhoutte, A., (1989), Similarity Measures in Scientometric Research: The jaccard index versus salton's cosine formula, *Inf. Process. Manage*, 25(3): pp.315-318.

Handschuh, S., and Staab, S., (2003), Annotation of the Shallow and the Deep Web, *Frontiers in Artificial Intelligence and Applications Volume (96).*

Handschuh, S., Staab S., and Ciravegna, F., (2002), S-CREAM - Semi-automatic CREtion of Metadata, *In Proceedings of the European Conference on Knowledge Acquisition and Management*, Madrid, Spain Springer.

Handschuh, S., Staab, S., and Maedche, A., (2001), CREAM: Creating relational metadata with a component-based, ontology-driven annotation framework, *In proceedings of the International Conference of Knowledge Capture*, Victoria, British Columbia, Canada ACM Press.

Hanna, D. E., Dudka, M. G., and Runlee, S. C., (2000), 147 Practical Tips for Teaching Online Groups: Essentials of Web-Based Education, *Madison, WI: Atwood.*

Harabagiu, S., Moldovan, D., Pasca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., Girju, R., Rus, V., and Morarescu, P., (2000), FALCON: Boosting knowledge for answer engines, In Proceedings of the 9^{th} Text REtrieval Conference (TREC).

Harel, I., and Papert, S., (1991), Constructionism, Norwood, NJ: *Ablex Publishing Corporation.*

Hendler, J., and McGuinness, D., (2000), The DARPA Agent Markup Language, *IEEE Intelligent Systems*, 15(6): pp. 72-73.

Hiemstra, R., (1996), What's in a world? Changes in Self-Directed Learning Language Over a Decade, *In Proceedings of the 1996 International Symposium on Self-Directed Learning*, West Palm Beach, FL. Retrieved November 15, 2008 from http://home.twcny.rr.com/hiemstra/word.html.

Hogue, A., and Karger, D., (2005), Thresher: automating the unwrapping of semantic content from the world wide web, *In Proceedings of the 14th International World Wide Web Conference*, May 10–14, Chiba, Japan, pp. 86–95.

Hübscher, R. and Puntambekar, S., (2002), Adaptive Navigation for Learners in Hypermedia Is Scaffolded Navigation, *In Proceedings of the 2nd International Conference Adaptive Hypermedia*, pp. 185-192.

IEEE LOM, (2002), 1484.11.2 Standard for Learning Technology,- ECMAScript Application Programming Interface for Content to Runtime Services Communication, Retrieved November 15, 2008 from http://ltsc.ieee.org/wg12/.

Jones, W., (1987), How Do We Distinguish the Hyper from the Hype in Non-Linear Text?, *In Proceedings of the 2nd IFIP Conferenceof Human-Computer Interaction-INTERACT 87, Buiilinger, H. J. and Shackel, B. (eds.), Federal Republic of Germany*, 1-4 September, Elsevier Science Publishers, Amsterdam, New York, pp. 1107-1113.

Jovanovic, J., Gaševic, D., Brooks, C., Devedžic, Hatala, V. M., Eap, T., Richards, G., (2007), Using Semantic Web Technologies to Analyze Learning Content, *IEEE Internet Computing*, 11(5): pp. 45-53.

Jovanovic, J., Gaševic, D., and Devedzic, V., (2006), Dynamic Assembly of Personalized Learning Content on the Semantic Web, *In Proceedings of The 3rd European Semantic Web Conference*, ESWC 2006, Budva, Montenegro.

Jul, S. and Furnas, G. W., (1997), Navigation in Electronic Worlds, *a CHI'97 workshop, SGCHI bulletin, ACM press* , 29(4): pp. 44-49.

Kahan, J., Koivunen, M., Prud'Hommeaux, E., and Swick, R. R., (2001), Annotea: An Open RDF Infrastructure for Shared Web Annotations, *In Proceedings of the 10th International World Wide Web Conference, ACM Press*, Hong Kong, pp. 623-632.

Kalfoglou, Y., (2001), Exploring Ontologies, Handbook of Software Engineering and Knowledge Engineering, *World Scientific Publishing*, 1: pp. 863-887.

Karagiorgi Y., and Symeou, L., (2005), Translating Constructivism into Instructional Design: Potential and Limitations, *Educational Technology & Society*, 8(1): pp. 17-27.

Kashihara, A., and Hasegawa, S., (2004), Unknown Awareness in Navigational Learning on the Web, *In Proceedings of EDMEDIA2004*, pp. 1829-1836.

Kashihara, A., and Hasegawa, S., (2003), LearningBench: A Self-Directed Learning Environment on the Web, *In Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2001*, pp. 1032-1039.

Kashihara A., Sakamoto, M., Hasegawa, S., and Toyoda, J., (2000), Making Exploration History Interactive for Web-based Learning, *In Proceedings of International Conference on Computers in Education 2000*, Taipei, Taiwan, pp.934-941.

Kelly, D., and Teevan, J., (2003), Implicit Feedback for Inferring User Preference: A Bibliography. *SIGIR Forum*, 37, pp. 18-28.

Khamidoullina, I., Reggers, T., and Zeiliger, R., (2001), NESTOR-Integrated Tools for Active Navigation and Constructive Learning, In C. Montgomerie & J.

Viteli (Eds.), *In Proceedings of World Conference on Educational Multimedia. Hypermedia and Telecommunications 2001*, Chesapeake, VA: AACE, pp. 959.

Kietz, J.-U., and Volz, R., (2000), Extracting a Domain-Specific Ontology from a Corporate Intranet, *In Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop*, Lisbon, Portugal, pp. 167-175.

Kiryakov, A., Popov, B., Terziev, I., Manov, D., and Ognyanoff, D., (2003), Semantic Annotation, Indexing and Retrieval, *In proccedings of the Second International Semantic Web Conference (ISWC'2003)*, Florida, USA, pp. 484-499.

Knowles, M., Self-Directed Learning: A Guide for Learners and Teachers, (1975), *New York: Association Press.*

Koivunen, M.-R., (2005), Annotea and Semantic Web Supported Collaboration, Invited talk at Workshop on User Aspects of the Semantic Web (UserSWeb), *at European Semantic Web Conference (ESWC 2005)*, Herakliion, Greece.

Landow, G. P., (1992), Hypertext: The Convergence of Contemporary Critical Theory and Technology, *Baltimore: John Hopkins University Press.*

Leacock, C., and Chodorow, M., (1998), Combining Local Context and Wordnet Similarity for Word Sense Identification, *In Fellbaum, C., ed.: WordNet: An electronic lexical database. MIT Press,* pp. 265-283

Lee M., and Baylor, A. L, (2006), Designing Metacognitive Maps for Web-Based Learning, *Educational Technology & Society*, 9(1): pp. 344-348.

Linard, M., and Zeiliger, R., (1995), Designing a Navigational Support for Educational Software, In B. Blumental and J. Gornostaev (eds.), *In Proceedings of EWHCI'95. Moscow: ICSTI*, pp. 130-145.

Liu, B., Chin, C. W., and Ng, H. T., (2003), Mining Topic-Specific Concepts and Definitions on the Web, *In Proceedings of the World Wide Web Conference*, New York, NY, USA: ACM Press, pp. 251-260.

Liu, M., (1994), Hypermedia-Assisted-Instruction and Second Language Learning: A Semantic-Network-Based Approach, *Computers in the Schools*, 10 (3/4): pp. 293-312.

Liu, X., Zhou, Y., and Zheng, R., (2007), Measuring Semantic Similarity in WorldNet, *In Proceedings of International Conference of Machine Learning and Cybernetics*, Hong Kong, pp. 3431-3435.

Long, H. B., (2000), Understanding Self-Direction in Learning, *In H. B. Long & Associates (Eds.), Practice & theory in self-directed learning, Schaumburg, IL: Motorola University Press*, pp.11-24.

Manning, C., Raghavan, P., and Schutze, H., (2008), Introduction to Information Retrieval, *Cambridge University Press.*

Manola, F., and Miller, E., (2004), RDF Primer, W3C Recommendation, Retrieved November 15, 2008 from http://www.w3.org/TR/REC-rdf-syntax/

Mason, C., (1992), Concept mapping: A Tool to Development Reflective Science Instruction, *Science Education*, 76: pp. 51-63.

McAleese, R., (1998), Coming to Know: The Role of the Concept map – Mirror, Assistant, Master?, *ERIC Document Reproduction Service No. ED429968.*

Miles, A., and Brickley, D., (2005), SKOS core guide: W3C working draft, Retrieved November 15, 2008 from http://www.w3.org/TR/2005/WD-swbp-skos-core-guide-20051102/.

Mintzes, J.J., Wandersee, J. H., and Novak, J.D., (1997), Meaningful Learning in Science: The Human Constructivist Perspective, *In G.D. Phye (Ed.), Handbook of academic learning: Construction of Knowledge*, San Diego: Academic Press, pp. 405-447.

Mitsuhara, H., Nakaya, S., Kanenishi, K., and Yano, Y., (2008), E-Notebook Tool for Effective Knowledge Construction from Web: Encouragement of Multi-Perspective Thinking and Prevention of Copy-and Paste, *In Proceedings of the Seventh IASTED International Conference on Web-Based Education*, Innsbruck, Austria, pp.406-411.

Mizoguchi, R., (2001), Ontological Engineering: Foundation of the Next Generation Knowledge Processing, *Web Intelligence 2001*: pp. 44-57.

Moallem, M., (2001), Applying Constructivist and Objectivist Learning Theories in the Design of a Web-based Course: Implications for Practice, *Educational Technology & Society*, 4 (3): pp. 1436-4522.

Narciss, S., Proske, A., and Koerndle, H., (2007), Promoting Self-Regulated Learning in Web-based Learning Environments, *Computers in Human Behavior*, 23: pp. 1127-1144.

Navigli, R., Velardi, P., and Gangemi, A., (2003), Ontology Learning and Its Application to Automated Terminology Translation, IEEE Intelligent Systems, 18(1): pp. 22-31.

Nelson, T. H., (1968), Hypertext Implementation Notes, *Unpublished manuscript*, Retrieved November 15, 2008, from Project Xanadu, Archive Web site: http://xanadu.com/XUarchive/hin68.tif

Nielsen, J., (1990), Hypertext and Hypermedia, *Academic Press.*

Novak, J, D., and Cañas, A, J., (2006a), The origins of the Concept Mapping Tool and the Continuing Evolution of the Tool, *Information Visualization*, 5(3): pp.175-184.

Novak, J, D., and Cañas, A, J., (2006b), The Theory Underlying Concept Maps and How to Construct Them, *Technical Report IHMC CmapTools*, Florida Institute for Human and Machine Cognition, Retrieved November 15, 2008 from http://www.stanford.edu/dept/SUSE/projects/ireport/articles/concept_maps/The%20Theory%20Underlying%20Concept%20Maps.pdf

Oard, D., and Kim, J., (2001), Modeling Information Content Using Observable Behavior, *In Proceedings of the ASIST Annual Meeting*, pp. 481-488.

Okebukola, P. A., (1992), Can Good Concept Mappers be Good Problem Solvers in Science?, *Research in Science and Technological Education*, 10(2): pp. 153-170.

Park, J., and Kim, J., (2000), Contextual Navigational Aids for Two World Wide Web Systems, *International Journal of Human-Computer Interaction*, 12(2): pp. 193–217.

Pedersen, T., Pakhomov, S., Patwardhan, S., and Chute, C., (2007), Measures of Semantic Similarity and Relatedness in the Biomedical Domain, *Journal of Biomedical Informatics*, 40 (3): pp. 288–299.

Pedersen, T., Patwardhan, S., and Michelizzi, J., (2004), Wordnet:: Similarity--Measuring the Relatedness of Concepts, *In Proceedings of the 19th National Conference on Artificial Intelligence, AAAI Press, Cambridge, MA*, pp. 1024-1025.

Popov, B., Kiryakov, A., Kirilov, A., Manov, D., Ognyanoff, D., and Goranov, M., (2003), KIM - Semantic Annotation Platform, *In Proceedings of the 2nd International Semantic Web Conference,* Florida, USA, pp. 834-849.

Porter, M.F., (1980), An algorithm for Suffix Stripping, *Program: news of computers in British University libraries*, 14(3): pp. 130-137, Retrieved November 15, 2008 from http://www.tartarus.org/~martin/PorterStemmer/.

Rada, R., Mili, H., Bicknell, E., and Blettner, M., (1989), Development and Application of a Metric on Semantic Nets, *IEEE Transactions on Systems, Man, and Cybernetics*, 19 (1989): pp. 17–30.

Reeve, L., and Han, H., (2005), Survey of Semantic Annotation Platforms, *In Proceedings of the 2005 ACM symposium on Applied computing*, Santa Fe, New Mexico, USAACM Press.

Resnik, P., Using Information Content to Evaluate Semantic Similarity in a Taxonomy, (1995), *In Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 448-453

Roberson, D. N., (2005), Self-Directed Learning--Past and Present, *ERIC Document No. ED490435.*

Robson, C., (1983), Experiment, Design and Statistics in Psychology, 2nd edition, *Penguin Education*, pp.164.

Sah, M., Hall, W., and De Roure, D. C., (2008), SemWeB: A Semantic Web Browser for Supporting the Browsing of Users using Semantic and Adaptive Links, In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems Doctoral Consortium, Lecture Notes In Computer Science (LNCS)*, Hannover, Germany.

Schunk, D.H., and Zimmerman, B.J, (2001), Self-Regulated Learning and Academic Achievement: Theoretical Perspectives, 2nd edition, *Lawrence Erlaum.*

Shavelson, R., Lang, H., and Lewin, B., (1994), On Concept Maps as Potential Authentic Assessments in science, *CRESST Technical Report 388.*

Smith, M., Welty, C., and McGuinness, D., (2004), OWL Web Ontology Language Guide, *W3C Recommendation*, Retrieved November 15, 2008 from http://www.w3.org/TR/owl-guide/.

Spiro, R., Feltovich, P., Jacobson, M., and Coulson, R., (1991), Cognitive Flexibility, Constructivism, and Hypertext: Random Access Instruction for Advanced

Knowledge Acquisition in Ill-Structured Domains, Educational Technology, 31 (9), pp. 24-33.

Steffe, L., and Gale J., (1995), Constructivism in Education, New Jersey: Lawrence Erlbaum.

Stojanovic, L., Staab, S., and Studer, R., (2001), eLearning in the Semantic Web, *In Proceedings of the WWWNet Conference*, Orlando, USA.

Sweller, J., and Chandler, P., (1994), Why Some Material is Difficult to Learn, *Cognition and Instruction*, 12(3): pp. 185-233.

Sweller, J., Merrienboer van, J.J.G. and Paas, G.W.C., (1998), Cognitive Architecture and Instructional Design, *Educational Psychology Review*, 10 (3): pp. 251-296.

Tauscher, L., and Greenberg, S., (2001), How People Revisit Web Pages: Empirical Findings and Implications for the Design of History Systems, *International Journal of Human Computer Studies*, 47(1): p. 97-138.

Tergan, S.-O., and Keller, T., (2005), Digital Concept Mapping in Learning Contexts: Integrating Knowledge, Arguments and Information Resources, In Proceedings of the IEEE 9th International Conference on Information Visualisation, London, UK, pp. 371-376.

Thiagarajan, R., Manjunath, G., and Stumptner, M., (2008), Computing Semantic Similarity Using Ontologies, *HP Labs Technical Report, HPL-2008-87.*

Thompson, C., and Mooney, R., (1999), Automatic Construction of Semantic Lexicons for Learning Natural Language Interfaces, *In Proceedings of the 16th National Conference on Artificial Intelligence*, Orlando, FL, USA, pp. 487-493.

Tsandilas, T., and Schraefel, M. C., (2004), Usable Adaptive Hypermedia Systems, *New Review of Hypermedia and Multimedia*, 10 (1): pp. 5-29.

Tsandilas, T., and Schraefel, M. C., (2003), User-Controlled Link Adaptation, *In Fourteenth ACM conference on Hypertext and hypermedia*, Nottingham, UK.

Uren, V., Cimiano, P., Iria, J., Handschuh, S., Vargas-Vera, M., Motta, E., and Ciravegna, F., (2005), Semantic Annotation for Knowledge Management: Requirements and a Survey of the State of the Art, *Journal of Web Semantics*, 4(1): pp. 14-28.

van Harmelen, F., (2004), The Semantic Web: What, *Distributed Systems Online*, 5:3.

van Heijst, G., Schreiber, A. T., and Wielinga, B. J., (1997), Using Explicit Ontologies in KBS Development, *International Journal of Human-Computer* Studies, 46(2-3): pp. 183-292.

van Rijsbergen, C. J., Information Retrieval, *Butterworth*, 1979.

Vargas-Vera, M., Motta, E., Domingue, J., Lanzoni, M., Stutt, A., and Ciravegna, F., (2002), MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup, *In Proceedings of the 13th International Conference on Knowledge Engineering and Management*, Spain, pp. 379-391.

Vygotsky, L. S., (1986), The Genetic Roots of Thought and Speech, *In A. Kozulin (Trans. & Ed.), Thought and language. Cambridge, MA: MIT Press.*

W3C SPARQL, (2005), SPARQL Query Language for RDF, W3C, Retrieved April 9, 2009 from http://www.w3.org/TR/rdf-sparql-query/.

White, R., and Gunstone, R., 1992, Probing Understanding, *The Falmer Press.*

Wilson, B. G., (1998), Constructivist Learning Environments: Case studies in instructional design, *New Jersey: Educational Technologies.*

Wilson, B., and Lowry, M., (2000), Constructivist Learning on the Web, Retrieved February 1, 2009, from http://carbon.cudenver.edu/~bwilson/WebLearning.html

Winne, P.H., and Perry, N.E., (2000), Measuring Self-Regulated Learning, In P. Pintrich, M. Boekaerts, & M. Seidner (Eds.), *Handbook of self-regulation*, Orlando, FL: Academic Press, pp. 531-566.

Woukeu, A., Wills, G., Conole, G., Carr, L., Kampa, S., and Hall, W., (2003), Ontological Hypermedia in Education: A Framework for Building Web-Based Educational Portals, *In Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Honolulu, Hawaii, USA.

Wright, P., and Lickorish, A., (1990), An Empirical Study of Two Navigation Systems for Two Hypertexts, *In Proceedings of the 2nd Hypertext Conference, York, Available in McAleese, R. (ed.) Hypertext: State of the Art*, pp. 84-93.

Wu, H., Houben, G.J., and De Bra, P., (1998), AHAM: A Reference Model to Support Adaptive Hypermedia Authoring, *In Proceedings of the "Zesde Interdisciplinaire Conferentie Informatiewetenschap"*, Antwerp, Belgium, pp. 77-88.

WU, S.-H., and HSU, W.-L., (2002), SOAT: A Semi-Automatic Domain Ontology Acquisition Tool from Chinese Corpus, *In Proceedings of the 19th International Conference on Computational linguistics*, Association for Computational Linguistics, Taipei, Taiwan, pp.1-5.

Wu, Z., and Palmer, M., (1994), Verb Semantics and Lexical Selection, *In Proceedings of the 32nd Annual Meeting of the Associations for Computational Linguistics*, pp.133-138.

Zeiliger, R., Belisle, C. and Cerratto, T., (1999), Implementing a Constructivist Approach to Web Navigation support, In B. Collis & R. Oliver (Eds.), *In Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Chesapeake, pp. 403-408.

Zeiliger, R., Reggers, T., Baldewyns, L., and Jans, V., (1997), Facilitating Web Navigation: Integrated tools for Active and Cooperative Learners, *In proceedings of the 5th International Conference on Computers in Education*, Sarawak, Malaysia.

Zhihong, Z., and Mingtian, Z., (2003), A Distributed Description Logic Approach to Semantic Annotation, *In Proceedings of the 4th International Conference on Parallel and Distributed Computing*, Application and Technologies, pp. 219-221.

Zimmerman, B.J., (2002), Becoming a Self-Regulated Learner: An Overview, *Theory into Practice*, 41(2): pp.64-70.

Couto, F. M., Silva, M. J., and Coutinho, P. M., (2007), Measuring Semantic Similarity between Gene Ontology Terms, *Data and Knowledge Engineering*, 61(1): pp.137-152.