

Semantic Rule-based Approach for Supporting Personalised Adaptive E-Learning

Maryam YARANDI

A thesis submitted in partial fulfilment of the requirements
of the University of East London for the degree of
Doctor of Philosophy

School of ARCHITECTURE, COMPUTING & ENGINEERING

Director of Study: Dr. Hossein JAHANKHANI

Supervisor: Dr. Abdel-Rahman H. TAWIL

June 24, 2013

Abstract

Instructional designers are under increasing pressure to enhance the pedagogical quality and technical richness of their learning content offerings, while the task of authoring for such complex educational frameworks is expensive and time consuming. Personalisation and reusability of learning contents are two main factors which can be used to enhance the pedagogical impact of e-learning experiences while also optimising resources, such as the overall cost and time of designing materials for different e-learning systems.

However, personalisation services require continuous fine tuning for the different features that should be used, and e-learning systems need sufficient flexibility to offer these continuously required changes.

The semantic modelling of adaptable learning components can highly influence the personalisation of the learning experience and enables the reusability, adaptability and maintainability of these components. Through the discrete modelling of these components, the flexibility and extensibility of e-learning systems will be improved as learning contents can be separated from the adaptation logic which results in the learning content being no longer specific to any given adaptation rule, or instructional plan.

This thesis proposes an innovative semantic rule-based approach to dynamically generate personalised learning content utilising reusable pieces of learning content. It describes an ontology-based engine that composes, at runtime, adapted learning experiences according to learner's interaction with the system and learner's characteristics. Additionally, enriching ontologies with semantic rules increases the reasoning power and helps to represent adaptation decisions. This novel approach aims to improve flexibility, extensibility and reusability of systems, while offering a pedagogically effective and satisfactory learning experience for learners. This thesis offers the theoretical models, design and implementation of an adaptive e-learning system in accordance with this approach. It also describes the evaluation of developed personalised adaptive e-learning system (Rule-PAdel) from pedagogical and technical perspectives.

Acknowledgement

I have no words for thanking my director of studies, Dr. Hossein Jahankhani, and my supervisor, Dr Abdel-Rahman Tawil for their continued support and encouragement throughout the past four years. I am very grateful to them for their invaluable advice and insightful guidance to make this work possible.

Thanks to everyone in the university for helping to create such a constructive and enjoyable working environment. I'd particularly like to thank Professor Hamid Jahankhani and Professor Cornelia Boldyreff for their helpful feedback and comments on my work. I'd also like to thank Dr. Caroline Dunmore for coordinating helpful researcher development programmes.

As importantly I would also like to thank my mother and father for their unflagging love, prayers and for the support I have received from them throughout has been overwhelming. I will be forever indebted with them. I would like to say very big thanks to my husband for his encouragement, loving assistance, unconditional support and continued believe in my education. I also would like to extend my heartfelt gratitude to my son Ali, for his expertise in English and Computer Science, in proofing this thesis, his helpful guidance in the implementation part of this work and his patience to read my thesis in progress. My appreciation will not be complete without showing sincere gratitude to my younger son, Amir Ali for his patience during the whole time of my Ph.D. research.

Contents

Acknowledgement	i
1 Introduction	1
1.1 Motivation	1
1.2 Research Question	4
1.3 Aims and Objectives	4
1.4 Research Contribution	6
1.5 Thesis Overview	7
2 Learning Theories	9
2.1 Introduction	9
2.2 Learning Theory	10
2.2.1 Behaviourism	10
2.2.2 Cognitivism	11
2.2.3 Constructivism	12
2.2.4 Learning Theory in Adaptive E-learning	13
2.3 Learning Styles	14
2.3.1 Kolb's Learning Style Model	15
2.3.2 Felder-Silverman Learning Style Model	16
2.3.3 VARK Learning Style Model	18
2.4 Item Response Theory	19
2.4.1 Item Characteristic Curve Models	21
2.4.2 Estimate Learner's Ability	22
2.4.3 Test Calibration	24
2.5 Summary	25

3	Adaptive E-Learning Approaches	27
3.1	Introduction	27
3.2	Adaptive Hypermedia	27
3.3	Methods of Adaptation in Educational System	28
3.3.1	Adaptive Presentation	29
3.3.2	Adaptive Navigation	29
3.4	Adaptive Educational Hypermedia systems	31
3.4.1	Different Generation of AEH Systems	32
3.5	Specifications and Standards for Metadata	34
3.6	Semantic Modelling using Semantic Web Technologies	36
3.6.1	Ontology	39
3.6.2	Ontology Languages	41
3.6.3	Using Ontology in Personalised E-learning Systems	46
3.7	Extending OWL with Rules	48
3.7.1	Semantic Web Rule Language (SWRL)	49
3.8	Survey of Personalised E-learning Systems	51
3.8.1	ADAPT ²	51
3.8.2	Personalised E-Learning in the Semantic Web	51
3.8.3	Personalisation Services	52
3.8.4	DIOGENE	53
3.8.5	TANGRAM	54
3.8.6	Curriculum Content Sequencing System	55
3.8.7	Protus 2.0	56
3.8.8	Analysis of Personalised E-learning Systems	57
3.9	Summary	60
4	Specification and Design	62
4.1	Introduction	62
4.2	Research Methodology	63
4.2.1	Research Approach	64
4.2.2	Methods of Data Collection	65
4.3	Justifying Semantic Rule-based Approach For Supporting Personalised Adaptive E-Learning	66

4.3.1	Model Separation and Model Types	69
4.3.2	Ontology and Rule for Model Representation	70
4.3.3	Abstraction through Ontological Modelling	72
4.3.4	Instructional Objects using content on finer granularity level	73
4.3.5	The System Architecture	75
4.4	Semantic Rules for Describing Adaptivity	79
4.4.1	Adaptive Navigational Support	80
4.4.2	Adaptive Presentation	82
4.4.3	Adaptive Guidance	82
4.5	Design Issues in Modelling Learning Content	83
4.5.1	Content Structure and Granularity	84
4.5.2	The Content Model Ontology	85
4.6	Modelling the Learners	87
4.6.1	Pedagogical Issues	88
4.6.2	Describing the Learner model ontology	89
4.6.3	Acquiring and Updating the Learner Model	91
4.7	Describing the Domain Model	92
4.7.1	Defining Topics and Pedagogical Relations	93
4.8	Describing the Assessment Model	95
4.8.1	Test Calibration using Item Response Theory	96
4.8.2	Describing the Assessment model ontology	97
4.9	The required Technologies for the Semantic Rule-Based Approach	98
4.9.1	Data Modelling with OWL	99
4.9.2	Semantic Rules using SWRL	100
4.9.3	OWL Manipulating, Serialising and Parsing using OWL API	101
4.9.4	OWL Reasoning using Pellet	101
4.9.5	Data Representation using XHTML	102
4.9.6	Delivering Content with Apache Tomcat	102
4.10	Summary	103
5	Implementation	104
5.1	Introduction	104
5.2	Technological Architecture of Rule-PAdel	104

5.3	Implementation Issues of Content Modelling	108
5.3.1	Domain Structure	108
5.3.2	Instructional Objects	110
5.3.3	IO Annotation	110
5.4	Implementation Issues of Learner Modelling	113
5.4.1	Ontology-based annotation of Learner	113
5.4.2	Initialising Learner Model	120
5.4.3	Updating and Maintaining the Learner Model	121
5.5	Adaptation Model	121
5.5.1	SWRL Rules for Adaptation	122
5.5.2	Adaptive Course Structure	122
5.5.3	Adaptive presentation of Learning Content	124
5.5.4	Adaptive guidance	129
5.6	Rule-based Adaptation Process	131
5.6.1	Course Structure Constructor	132
5.6.2	Recommender	133
5.6.3	Guide	134
5.6.4	Assessment Analyser	136
5.7	Creating a Personalised Learning Content in Rule-PAdel	138
5.7.1	Improve the Efficiency of System	138
5.8	Instance of Rule-PAdel Personalised Learning on Fraction	140
5.9	Summary	141
6	Evaluation	142
6.1	Introduction	142
6.2	Methodology	143
6.3	Learner Satisfaction	144
6.3.1	Overview	145
6.3.2	Learner's Evaluation	146
6.4	Learner Performance	155
6.5	Teacher Satisfaction	159
6.6	Flexibility and Extensibility of System	160
6.6.1	Flexibility through Generating On-the-fly Personalisation	161

6.6.2	Flexibility and Extensibility in Using Different Components of Personalisation	161
6.7	Reusability of the Components of Adaptivity	167
6.8	Comparing Rule-PAdel with other similar Adaptive System	169
6.9	Summary	170
7	Conclusion and Future work	173
7.1	Introduction	173
7.2	Achieved Objectives	173
7.3	Contribution to Knowledge	177
7.4	Future Work	178
A	Publications	197
A.1	Journal and Conference Publications	197
A.2	Academic Posters	198
B	Academic Posters	199
C	SWRL rules	201
C.1	Annotated course structure	201
C.2	Ability Level	201
C.3	Non-numeric terminology	202
C.4	Learning Style	202
C.5	Recommended IO	203
C.6	Guidance	204
D	Screenshots of the Programme	205
E	Domain Mathematics Ontology	212
F	Student Evaluation Questionnaire	214

List of Figures

2.1	Item Characteristic Curve (ICC)	20
2.2	Item Characteristic Curve for a two parameter model with $b=1$ and $a=1.5$	22
3.1	Semantic Web Stack (Berners-Lee, 2006)	37
3.2	Example of a small ontology	40
3.3	A screen shot of Protégé editor	42
3.4	RDF graph of a single statement	43
3.5	An architecture for personalisation services (Dolog et al., 2004)	52
4.1	Three dimensions of adaptivity	74
4.2	The architecture of system	75
4.3	The adaptive engine	78
4.4	The different kind of Learning Object	85
4.5	The Content model Ontology	86
4.6	The different types of Instructional Object	87
4.7	Learner Model Ontology	89
4.8	A segment of domain topics	94
4.9	An excerpt from the Domain ontology	95
4.10	Test Ontology	98
5.1	Technological Architecture of Rule-PAdel	105
5.2	Graphical representation of the Learner model	114
5.3	Rule-PAdel Personalised Learning Process	139
6.1	Learners' response to Comparing adaptive and non-adaptive system questions	149
6.2	Learners' satisfaction level about adaptive navigation	150

6.3	The suitability of personalised content	151
6.4	The usage of adaptive guidance	152
6.5	The effect of adaptive system in promoting learners interest	153
6.6	Summery of learners' evaluation	154
6.7	Knowledge gain vs. pre-test scores: Left Topic Set 1; Right Topic Set 2 .	158
6.8	Annotated Course Structure of Fraction topic (version 1)	165
6.9	Annotated Course Structure of Exponents topic (Version 2)	166
6.10	A segment of domain ontology for science subject	166
6.11	Screenshot of the system with first instructional plan (version 1)	167
6.12	Screenshot of the system with second instructional plan (version 1)	168
B.1	Academic poster by Maryam Yarandi; won the Perryman award for the best poster presentation at Advances in Computing & Technology Conference, 19 Jan 2012	199
B.2	Academic poster by Maryam Yarandi; the winner of the poster competition at BCS e-Learning SGs INSPIRE 2012 Conference, 23 Aug 2012	200
D.1	Login Page	205
D.2	Registration form	206
D.3	Learning style Test	207
D.4	Prior knowledge form Test	208
D.5	Annotated course structure	208
D.6	definition of proper fraction	209
D.7	Example of proper fraction	209
D.8	Exercise about proper fraction	209
D.9	Recommended adaptive guide	210
D.10	Recommended adaptive guide	210
D.11	Recommended adaptive guide	210
D.12	Recommended adaptive guide	211
E.1	Domain ontology topic Fraction	212
E.2	Domain ontology topic Exponent	213

List of Tables

5.1	The reaction of Guide component to different guidance	135
6.1	Fraction topics used in the evaluation	146
6.2	The questions associated in different areas of questionnaire	148
6.3	Test and knowledge gain result (Topic Set 1)	156
6.4	Test and knowledge gain result (Topic Set 2)	156
6.5	The result of the Matched-Pairs T-Tests between Score _{pre-test} and Score _{post-test}	157
6.6	Statistics about weaker learners for learning	158
6.7	Different adaptive guidance	162
6.8	The result of comparison Rule-PAdel with some other adaptive e-learning systems	170

List of Abbreviations

ACS	Annotated Course Structure
AE	Adaptive Engine
AES	Adaptive Educational System
AH	Adaptive Hypermedia
AHS	Adaptive Hypermedia System
CAT	Computerised Adaptive Test
CSC	Course Structure Constructor
DL	Description Logic
ICC	Item Characteristic Curve
IO	Instructional Object
IRI	Internationalised Resource Identifier
IRT	Items Response Theory
ITS	Intelligent Tutoring System
LMS	Learning Management Systems
LO	Learning Object
LSI	Learning Style Inventory
MLE	Maximum Likelihood Estimator
OWL	Ontology Web Language

RDF	Resource Description Framework
RDFS	RDF Schema
Rule-PAdel	Rule-based personalised adaptive e-learning system
SWRL	Semantic Web Rule Language
VARK	Visual Auditory Read/write Kinaesthetic

Chapter 1

Introduction

1.1 Motivation

Rapid advancements in the design and integration of networked technologies into everyday activities have created new perceptions about the exploitation of these technologies into teaching and learning. E-Learning is often described as the use of network technology (e.g. the internet) to design, deliver, select, administer and extend learning (Hamdi, 2007). E-Learning has broken down geographical barriers so that education can now occur in an active and interactive way at anytime, anywhere and without necessarily the presence of a human instructor (Kabassi & Virvou, 2004; Grigoriadou & Papanikolaou, 2000). As a result, it is now more important than ever that most appropriate learning systems and resources are utilised to provide a satisfactory level of support for learners.

To achieve these higher levels of learner's satisfaction the next generation of e-learning systems need to provide an adaptive and flexible learning environment to support learner's requirements. However, most current e-learning systems are based on one-size-fits-all approach to learning content delivery irrespective of learner's background knowledge, abilities, preferences or learning style. This problem of delivering the same content to all learners can be addressed by using personalisation strategies to adapt learning content to the learner's requirements. Therefore, one of the key issues in the next generation of e-learning systems is to identify learners' needs, educational behaviours and pace, and to design a curriculum that is tailored to individual learner's abilities. By adapting the

curriculum to an individual learner, modern e-learning systems can help learners in learning more effectively and efficiently.

Adaptation is an important issue for web-based educational systems from two aspects: firstly, most web-based e-learning systems are used by a much wider diversity of learners than any standalone application; secondly, learners usually work with web-based e-learning systems individually and therefore can't get the intelligent and personalised assistance that a teacher can provide in a normal classroom (Brusilovsky, 1999). Accordingly, many researchers have recently attempted to provide personalisation mechanisms for web-based learning systems (Brusilovsky, 1999; Chen, 2008; Brusilovsky & Peylo, 2003; Chen et al., 2006; Milosevic et al., 2006; Henze et al., 2004; Baylari & Montazer, 2009; Jeong et al., 2012; Jovanović et al., 2009; Trella et al., 2005).

Web-based Adaptive Educational Systems (AES) inherit from both intelligent tutoring systems and adaptive hypermedia systems. The most significant aim of an Intelligent Tutoring System (ITS) is to support adaptive learning by using knowledge about the domain, the learner and teaching strategies (Brusilovsky, 1998). The main goal of ITS systems is to tailor and personalise learning contents for the individual learners. Therefore, such ITSs are adapted in a certain domain and the adaptation strategies are embedded into the learning content. The learner model is also highly dependent on how the adaptive engine at the heart of ITS is designed. However, reusing different elements of personalisation are difficult as the strategies for tailoring the content to the learners are usually embedded into the adaptive engine. ELM-ART (Brusilovsky et al., 1996), Interbook (Brusilovsky et al., 1998) and early versions of AHA! (De-Bra & Calvi, 1998) are implementation of such systems. For example, the Adaptive Hypermedia Systems (AHS) apply different types of learner models to adapt the content and the links of hypermedia pages to the learners (Brusilovsky, 1998). In AHS although the learner and content model are separated, the adaptive strategies are still entwined in the content model, learner model or the system's business logic. Therefore the use of different pedagogical models or adaptation strategies would involve the re-authoring of learner model, content model and the system's business logic.

Semantic web technologies in particular ontologies provide opportunities for developing modern e-learning systems. An ontology is a formal, explicit specification of a conceptu-

alisation (Gruber, 1993). The *conceptualisation* emphasises that ontology represents an abstract model of a phenomenon in the world as it helps to identify appropriate domain concepts and semantic relationships among these concepts with formal definitions in terms of axioms (Chi, 2009).

Ontologies support the separation of the domain knowledge from the operational knowledge (Noy & McGuinness, 2001). Through this separation, in educational systems, the components of adaptivity can be defined using separate ontologies which allow course authors (in particular for those with limited programming expertise) to modify these components without the need to change the implementation of the whole system. In addition, through this separation, the system becomes flexible and extensible by using various adaptation techniques and instructional plans across the same domains and contents. Moreover, the explicit conceptualisation provided through ontology allows for a higher reusability for these components. Well-designed ontologies facilitate the representation of learning contents in various granularity levels which results in more effective personalisation.

Ontologies together with reasoning services play a key role in developing high quality and cost-effective systems, where they are the most suitable means for representing knowledge due to their flexibility and extensibility in designing concepts and their relationships (Tserendorj, 2010). In addition, ontologies have some basic inference mechanisms which give users the ability to reason on their knowledge-bases and have the potential to provide fine grained semantic structures (Berners-Lee et al., 2001).

Ontologies enable machines to use knowledge-bases and apply reasoning techniques to infer adaptation decisions (Golemati et al., 2007). Ontology's reasoning mechanisms are derived from description logic which can be further empowered by rule-based reasoning to express relations which cannot be represented by ontological reasoning such as a prerequisite relationship between two topics in a learning domain. With such integration, rule empowered ontological systems can derive further information, have higher expressiveness to represent various formalisms and they can support adaptation strategies in educational systems. Therefore, this thesis focuses on designing a suitable architecture for adaptive systems by using ontology and semantic rule technologies which can then offer personalised e-learning. The proposed model should provide sufficient flexibility in using various adaptive strategies to enable educationally sound learning paths satisfying the learners'

pedagogical needs.

1.2 Research Question

Clearly, next generation of e-learning systems need to offer higher adaptability and flexibility to be able to support today's learning requirements. This research considers the advancements in semantic web technologies as well as the needs of the current students to improve these two qualities for learning environments. Therefore, the research questions addressed by this work are:

Is it possible to model the components of personalised learning (e.g. learner profile, learning content, domain and adaptation rules) independent from the system's core business logic using semantic ontologies enriched with inference rules?

This research will demonstrate whether the generated adaptive learning paths are pedagogically effective and satisfactory for the students. For such novel model, we will also investigate the reusability of the components of adaptivity and the flexibility and extensibility of the systems that implement this approach.

1.3 Aims and Objectives

The aim of this thesis is to propose an ontology-based approach for supporting dynamic personalised learning with sufficient flexibility which will be looked at from two aspects. First, whether it dynamically generates flexible personalised learning experiences for different learners based on their progress. Second, does it facilitate developing different personalised e-learning systems with different adaptation techniques, instructional plans, concept sequencing and domain of knowledge, without having an impact on the existing implementation (e.g. the adaptive engine, business logic)? To achieve this aim, separate ontological components are designed to describe the structure of the knowledge domain, characteristics of learners, features of learning content, instructional plan and assessment criteria. The separation enables course authors to use different adaptation techniques and instructional plans to generate learning courses that will be pedagogically satisfactory to

the learner. Additionally, the semantic modelling of adaptation components can highly influence the personalisation of the learning experience and enable the reusability and maintainability of the components of adaptivity. Moreover, the adaptability of content is based on different factors; one of them being the learner's ability which is calculated based on Items Response Theory (IRT) (Baker, 2001) in order to improve the accuracy of adaptation.

Therefore, the main objectives of this research with respect to e-learning systems implementing the proposed approach can be summarised as follows:

- Design an independent adaptive engine this does not contain any knowledge about a particular domain or specific teaching strategy.
- Design different components of adaptivity separately.
- Represent adaptation techniques with Semantic Rules to express adaptation strategies independently of an application's core engine.
- Utilise Item Response Theory to calculate learner's ability.

This thesis claims that through ontological modelling of different adaptivity elements and using rule-based reasoning for creating adaptation strategies, it is possible to design a flexible, extensible and independent architecture for adaptive e-learning systems.

In this model, system's independence is achieved through implementing an ontology-based adaptive engine, which is not entwined to any particular domain or specific teaching strategy. The flexibility is achieved in the following two ways: through dynamically generating personalised learning paths at runtime based on learner's interaction with the system; and by requiring authors to modify only the adaptation techniques, instructional plans and concept sequencing in order to achieve different e-learning systems for different pedagogical needs. Extensibility in this context can be realised by allowing new models to be added to the system without having any impact on the system's architecture or implementation.

To achieve these objectives a semantic rule-based approach for designing and implementing a flexible and extensible adaptive e-learning system is proposed. The proposed approach enables course authors to use different adaptation techniques, instructional plans, concept sequencing and knowledge domain.

1.4 Research Contribution

This thesis proposes a semantic rule-based approach to developing personalised e-learning system. Our approach proposes an ontological architecture featuring an independent adaptive engine. This engine does not include any knowledge about a particular domain or any adaptation strategy; it gets all the necessary information from related ontologies. Both the engine and the adaptation process are kept separate and are therefore reusable independently of each other. This separation together with the discreteness of the components of adaptivity allows course authors to update and improve the adaptive e-learning system iteratively without the need to change the existing implementation. This is a significant contribution to the state of the art where many existing adaptive e-learning systems require a complete reconstruction to implement any modifications.

The explicit conceptualisation of components of adaptivity in the form of ontologies is an innovative feature of our work. To the best of our knowledge, this form of conceptualisation has not previously been used in adaptive e-learning systems. Our approach enables course authors to define learning content in fine granularity levels to describe the adaptive features of each piece of content and to define different aspects of adaptivity (e.g. adaptation techniques). Moreover, the ontology-based approach addresses problems of maintenance and component reusability.

Representing adaptation techniques with semantic rules is also another innovative solution for expressing adaptation strategies independently of an application's core engine. Through this independence, the maintainability of systems is improved. Rules can also be employed to represent adaptation strategies in a way that the authors, who are non-professional programmers, would be able to develop and modify the rationale behind the adaptation process. Using rules also features a separation between adaptive techniques and the learning contents they are reasoning over. This separation facilitates the flexibility and extensibility of e-learning systems in using different contents or adaptation strategies. Adaptation using rule-based reasoning also offers personalisation at runtime based on the interaction of a learner with the system. This is another contribution of this work, as this form of personalisation has not previously been used in existing adaptive e-learning models. Through the interaction of learners with the system, the knowledge is updated. After the factual knowledge is changed, the rules are executed and as its result, the knowledge base

will be updated again with the new *inferred* knowledge.

1.5 Thesis Overview

The rest of this thesis is organised as follows:

- Chapter 2 reviews the necessary background knowledge on learning theory which is crucial for developing personalised e-learning systems. Firstly, it presents an overview of three popular learning theories, namely behaviourism, cognitivism and constructivism and also the effect of each of these theories in developing an e-learning framework. It continues by presenting the most famous learning style models, as learning style may be used to influence the personalisation of a learning content. Finally, the chapter reviews the Item Response theory which is used to calculate learners' ability accuracy. Learner ability is a learner's property which can influence the personalisation process.
- Chapter 3 presents a background about adaptive e-learning systems. This chapter starts by outlining different methods of adaptation in educational systems. It continues with a summary of semantic modelling using semantic web technologies, in particular ontologies enriched with rule-based reasoning. Finally, it reviews different ontology-based approaches to implementing adaptive e-learning systems and concludes with a brief analysis of these approaches.
- Chapter 4 introduces the design of a novel approach to supporting personalised adaptive systems. Firstly, it presents the details of the semantic rule-based approach to supporting adaptive e-learning systems. More specifically, it discusses the main ideas, principles and assumptions behind this approach and presents an ontology-based architecture for the systems that choose to implement it. After that, it highlights the importance of semantic rules for describing adaptivity followed by four sections, each describing how to model the four components of adaptivity (i.e. content, domain, learner and assessment model) and the design issues concerning each of them. Finally there is a discussion on the technologies which support the semantic rule-based approach.

- Chapter 5 then describes a prototype implementation of the semantic rule-based approach we use for supporting Personalised Adaptive e-learning (Rule-PAdel) systems. It starts with illustrating the technological architecture of Rule-PAdel systems, including the essential technologies for implementing each of its components. This is followed by individual sections detailing implementation issues of components of adaptivity. Then, it discusses how semantic rules represent different adaptation strategies. After that, it realises the adaptation process, rendering and delivery issues. This chapter also discusses the process of creating personalised learning in Rule-PAdel. Finally, this chapter describes an instance of Rule-PAdel for teaching fractions in mathematical domain.
- Chapter 6 presents the results of the evaluation of our approach, proposed in this thesis. This chapter presents learners' and teachers' satisfaction when using a Rule-PAdel system. It also presents the significant effects of an adaptive system on students' learning. Finally, discussed in this chapter is the technical evaluation of the proposed system.
- Chapter 7 concludes the thesis. It outlines the objectives achieved and the key contributions made in this work. Finally, it discusses the potential directions for future works.

Chapter 2

Learning Theories

2.1 Introduction

In recent years the rapid development of educational technologies has had a major impact on the way that people learn. These technologies potentially provide new means for lessening the cognitive load for learners. The learner will not have to memorise information as it is readily available at any given time. (Vavoula et al., 2009) . However, the technology is just a means for achieving a required purpose which is education in this thesis, not technology. Therefore, it is important to understand what is learning and how students learn. However we should mention that the technologies form an important aspect of e-learning systems and therefore should not be ignored.

In psychology and education, there exist different definitions for learning. However, the most common definition of learning is “ learning is a process that brings together cognitive, emotional, and environmental influences and experiences for acquiring, enhancing, or making changes in one’s knowledge, skills, values, and world views. “(Illeris, 2004; Ormrod, 1995). As we can understand from this definition learning is a process which focuses on what happens when learning occurs, however, learning theories are needed to explain how the learning process is done. Therefore, Section 2.2 of this chapter reviews the main categories of learning theory behaviourism, cognitivism and constructivism with focus on Constructivism.

Additionally, students have their own preferable approach for learning and they can learn

best if the learning processes adapt to their own approach. The way in which the students prefer to learn is called their learning style. Many learning style models have been developed to categorise the ways of learning from various perspectives. Section 2.3 reviews the most popular models of learning style.

Besides learning style, learner ability is a learner's property which can influence the personalisation process. When the learning material is too easy or too hard for a learner, the outcome of learning will be suboptimal. Therefore, adapting learning material to learners' ability is an important factor to help learners learn more effectively and efficiently. The item response theory is a popular theory in education which is applied to obtain a more precise estimation of learner ability. Accordingly, the last section provides an overview of Item Response Theory (IRT) and IRT models to estimate learner ability.

2.2 Learning Theory

The learning theories explain how students learn, so to implement an adaptive e-learning system, it is important to be identified with different learning theories and the potential effect of them on developing an adaptive e-learning system. Three famous categories of learning theories are behaviourism, cognitivism and constructivism which are described in this section.

2.2.1 Behaviourism

Behaviourism (behaviour theorists) focuses on behaviour of humans and ignores the inaccessible mental processes which are performed on the mind (Bechtel & Graham, 1999). They define learning as the acquisition of new behaviour. Therefore, they believe that a learner is a passive recipient of knowledge and a teacher should reinforce correct behaviour to her. They emphasise that behaviour can be modified and learning is measured by observable change in behaviour (Chen, 2009). As behaviourism believes that learning is a stimuli and response process, thus they emphasise that adaptive e-learning systems should produce suitable stimuli which causes behaviour from the learner which is related to successful learning.

The behaviourism is less popular than cognitivism and constructivism theories as it explains behaviour without referring to mental activity. The key principle of this theory is the reward or punishment of a new behaviour for both animal and human. It means that if someone is rewarded for a particular behaviour, she is encouraged to repeat it in similar situations. On the contrary, if she is punished, she is less likely to behave in the same way.

The behavioural theorists believe that the teachers in the case of traditional learning and the systems in the case of e-learning should present knowledge in a predefined order.

2.2.2 Cognitivism

In contrast with behaviourism theory which has a passive view to learning process, Cognitivism theory was developed to make mental processes as a primary object of learning. Cognitivists claim that learning is an internal process and memory is an active processor. The ability of people to learn is based on their prior knowledge and the amount of mental effort expended during the learning process. They have seen knowledge as symbolic mental constructions. Therefore, learners' prior knowledge are symbolic mental constructions in their minds and learning is the process of changing these constructions (Ausubel, 1960; Craik & Lockhart, 1972).

Cognitive scientists state that there exists external reality in environment. The learners receive information through their senses when they attend to information. Then this information is integrated into the pre-exist cognitive structure, converts to knowledge and stored in memory. Finally stored knowledge will be remembered via the retrieval process. Although the Cognitivists agree that a learner has an important role in the learning process, they believe that it is the teacher who has a central role in increasing the learner's attention and motivation and is also responsible for managing the content of learning activities to develop conceptual knowledge.

Although cognitivists and behaviourists have different views to learning, both of them have a same objective view to knowledge (given and absolute). Additionally, from the adaptive e-learning perspective both cognitivists and behaviourists agree that the computers should present information to be learnt and learners practice until they understand it. However, cognitivists are also considered active mental processes of the learner.

Piaget (Piaget, 1985) stated that learning process is iterative, thus new information is integrated with the learner's prior knowledge, and prior knowledge is modified to adapt with the new information. Therefore, to apply cognitivism into instruction in adaptive e-learning systems, cognitivist instructional designers should consider the learner's prior knowledge to learn a new learning objective. They should know that all learners do not have the same prior knowledge or learn in the same way. The lesson should be divided into pieces from simple to complex based on learner's prior knowledge (Piaget, 1985).

2.2.3 Constructivism

Constructivist theories believe that knowledge has to be constructed rather than transmitted. However, they agree with attention, encoding and retrieval of knowledge processes the same as cognitivists.

Although Constructivist theories agree with attention, encoding and retrieval of knowledge processes same as cognitivists, they believe that knowledge has to be constructed rather than passed directly from teacher to learner or from book to learner. They also state that there is no single representation of environment; different people have different interpretations of a world which is constructed from their experience. Therefore, knowing is an adaptive activity and learning is an active process of change in pre-existing knowledge constructed from experience. Learners actively take knowledge, connect it with their prior knowledge and construct their own interpretation (Muijs & Reynolds, 2005).

Constructivism theory recognises learner's prior knowledge as an important factor in learning as it can help or hinder the learner in learning new concepts. When new concepts are introduced, they should be integrated with the learner's knowledge structure. How learning proceeds depends on how new concepts fit into the knowledge structure already present in the learner's mind (Ausubel et al., 1978). In other words, the learner constructs meaning for new concepts by connecting them with the knowledge they already have. Therefore, a learner with a well-organised prior knowledge can learn many concepts more rapidly and easily.

From the constructivist's perspective, the responsibility of a teacher is not to lecture or explain the material in order to transfer knowledge but to create situations for students

which will encourage them to create the required mental constructions. The teachers also must transform the information into a format which is appropriate for learners based on their prior knowledge and current ability of understanding.

From the adaptive e-learning perspective, the information being presented should be relevant to the information which the learners have previously learned (Henze et al., 1999). The presented information should help the learner to recognise the relation between the knowledge and the real-world competencies. Additionally, the importance and generality of concepts being presented may be shown by explaining the information in other domains. Moreover, restructuring teaching by problem-solving, trying to identify learner's mathematical thinking and inquiring-oriented learning can promote the effectiveness of learning (Glaserfeld, 1995).

The constructivists emphasise that the traditional teacher-centre approach should be changed to student-centre approach as learners learn the most when they are with a teacher or computer than from a teacher or a computer. Therefore, a framework should be created that allows the learners to construct their representations of knowledge during learning and problem-solving and also integrates new concepts with existing knowledge. The teachers are not absent, they should design courses carefully and select learning activities in order to be learning effective and focused.

2.2.4 Learning Theory in Adaptive E-learning

Three previously discussed learning theories have different effects on the development of adaptive e-learning systems as they define how learners can learn effectively using such systems. The behaviourism is not a very applicable theory in implementing adaptive e-learning systems as they do not refer to the internal mental processes of the learner and many of its principles have been used by cognitivism. Behaviourism has only led to development of many intelligent tutoring systems where the computer programs including teaching strategies were used for presenting information and practising the learners to understand it.

Cognitivism considers both the behaviour of the learners and their mental processes which happen in their mind. Therefore, cognitivists approaches might consider learners'

prior knowledge in learning a new concept. Cognitivists have an objective view to the knowledge domain, thus they try to present learners with that same view. They can be used for teaching novice learners using adaptive e-learning systems by practising the learners, like Behaviourists. However, unlike Behaviourism, they consider the individual differences in education.

Compared to cognitivists and behaviourists, constructivists have a different view about the learning process as they claim learners construct their own knowledge domain by integrating new information into their existing knowledge and experience. Interest in constructivism has the following effect in learning: learners play a more prominent role in their own progress and they need more control on their learning. Moreover, instructional techniques tend to be more adaptive, flexible and open-ended and also materials become more interactive. Additionally, assessments become part of the learning process and grades are eliminated. However, some believe that solely using constructivism in education causes uncontrolled learning and made evaluation difficult instead of a systematic and structured instruction of knowledge.

Therefore, in adaptive e-learning systems there has to be a balance between Cognitivism and Constructivism. Constructivism may facilitate interactive learning environments; it provides some degree of freedom for learners to select their own learning routes and expects some responsibility for learners. However, cognitivism may constrain the knowledge domain. It considers the learners' prior knowledge and provides scaffolding in its teaching.

2.3 Learning Styles

Students learn in different ways. Keefe (Keefe, 1979) defines learning styles as the “composite of characteristic cognitive, affective, and physiological factors that serve as relatively stable indicators of how a learner perceives, interacts with, and responds to the learning environment.” In other words, learning style is the method in which an individual prefers to perceive and process information when interacting with the learning environment.

Research on learning styles has shown that identifying and considering learners' learning style in preparing instruction can improve the effectiveness of learning (Dwyer, 1998).

Therefore, by adapting learning material to a learner's learning style it is hoped to provide a powerful personalisation mechanism. This personalisation may result in a deeper understanding of learning materials and shorter learning time. This section reviews the most popular learning style models used in adaptive e-learning systems in detail.

2.3.1 Kolb's Learning Style Model

Kolb (Kolb, 1984) introduces the Learning Style Inventory (LSI) based on two major dimensions: *abstract/concrete* and *active/reflective*. In addition to LSI, Kolb also developed a theory of experiential learning. Kolb's theory (Kolb, 1984) states that learning has four stages which follow each other. First concrete experiences provide a basis for observation. In the next step the learner reflects on these observations and builds a theory of what this information might mean. Next, the learner creates abstract concepts based on their hypothesis. Finally, the implications of these concepts are tested in new situations. Then the process cycles back to the first stage of the experiential process. According to Kolb, the learner must complete the cycle of learning through all parts to fully understand the topic. Kolb identified four types of learning style that each places the learner in a line between concrete experience and abstract conceptualization; and active experimentation and reflective observation. These learning styles are as follows:

- **Converger (Active/Abstract):** Learners with this learning style like to experiment with new ideas and to work with practical applications. They prefer technical tasks and will use learning for problem-solving and decision-making. They have abilities in the areas of abstract conceptualisation and active experimentation. Learners with a Converging learning style have abilities to find practical uses for theories and ideas. They like to work actively on well-defined tasks. They prefer to deal with technical tasks and problems rather than with social and interpersonal discussions. They like interactive instruction, not passive.
- **Diverger (Reflective/Concrete):** Learners with this learning style have abilities in the areas of concrete experience and reflective observation. They are best in generating new ideas such as brainstorming. Divergers are best in viewing things from different perspectives. They prefer to use imagination to solve problems. They also tend to

be strong in the arts such as artists and musicians. People with the Diverging style prefer to work in groups.

- Assimilator (Reflective/Abstract): They are skilled in the areas of abstract conceptualisation and reflective observation. They need clear explanation rather than practical applications. They prefer inductive reasoning and logical approach for organising a wide range of information. They are more interested in abstract ideas and creating theoretical models and less focussed on interaction with others. Assimilators tend to work in math and the basic sciences and also enjoy work that involves planning and research.
- Accommodator (Active/Concrete): Accommodators are strongest in concrete experience and active experimentation. They prefer to do experiments and performing plans in the real world. They also prefer to work in groups to complete tasks. People with this learning style rely on other people's analysis rather than perform their own analysis. They like to work in technical fields or get jobs requiring action such as sales and marketing.

2.3.2 Felder-Silverman Learning Style Model

Felder and Silverman (Felder & Silverman, 1988) state that learners have different ways of receiving and processing information, thus they learn in different ways. The Felder-Silverman learning style model (Felder & Silverman, 1988) rates the learner's learning style in a scale of four dimensions. The learning style of each learner is determined through the result of a questionnaire which was developed in 1991 by Richard Felder and Barbara Soloman (Felder & Soloman, 1991). This questionnaire consists of 44 questions that classify a learning style across the following dimensions:

- Active and Reflective: Active learners can learn much by doing something in external world with information, i.e. explaining it to others, discussing or applying it. However, Reflective learners tend to think about information quietly first. They may be more interested in reviewing other learners' and professionals' opinions rather than doing real activities. Active learners work better in groups while reflective learners work better by themselves. Felder and Silverman (Felder & Silverman,

1988) state that Active learners do not learn much in situations that require them to be passive and reflective learners do not learn much in situations that provide no opportunity to think about the information being presented.

- Sensing and Intuitive: Sensing and intuition are two ways in which people tend to perceive the world. Sensing learners like observing and gather data through the senses, while, intuitive learners perceive data indirectly by way of the unconscious and imagination. Sensors tend to be patient with details and good at learning facts and doing laboratory work. For example, sensing learners will be interested in additional materials. Intuitive learners often prefer discovering possibilities and relationships. They prefer to learn abstractions and mathematical formulations. Sensors often like solving problems by well-established methods and dislike complications and surprises. Intuitive learners like innovation and dislike repetition. Sensing learners are strong in memorising facts; while intuitive learners are strong in grasping new concepts. Sensors are careful but may be slow; intuitors are quick but may be careless.
- Visual and Verbal: Having pictures, demonstrations, films, time lines, flow charts and diagrams is most effective for visual learners (Klašnja-Milićević et al., 2010). Verbal learners on the other hand, remember best when they deal with words, either in a written form or as spoken explanations. They acquire information when they engage in discussion, like verbal explanation and visual demonstration. When they explain things to others they can learn effectively.
- Sequential and Global: Sequential learners tend to learn in linear steps and solve problems by following linear reasoning patterns. They also have the ability to use materials when they understand them partially or superficially. Global learners on the other hand tend to feel out-of-step and understand materials in an arbitrary fashion. They usually make intuitive jumps when solving problems but find it difficult to explain how they found the solutions.

2.3.3 VARK Learning Style Model

Fleming (Fleming, 1987) developed an inventory which referred to VARK learning styles to help students learn more about their learning preferences. The acronym VARK stands for Visual, Aural, Read/write and Kinaesthetic sensory modalities that are used for learning information. The VARK Learning model does not influence the sequence or structure of learning material. It only influences the nature and form of the delivered learning material. The learner's VARK preferences are determined using a questionnaire. Explanations of the four different preferences of the VARK learning style are as follows:

- **Visual (V):** Visual learners learn best by seeing; graphic displays such as diagrams, flow charts, graphs, labelled diagrams and all the symbolic arrows, allow learners to interpret data in a logical manner, visual learners prefer to use these graphic displays as learning tools. Learners who prefer this type of learning see information presented in a visual rather than in written form. Visual learners like to use images, pictures and maps to organise information.
- **Auditory (A):** Auditory learners learn best by hearing information and speech. They can get much information from lectures, tutorials, tapes, group discussion, reading aloud to themselves and email. Learners with this particular learning style remember most things they are told. Auditory individuals benefit from background music when they work. They are also able to debate and discuss with one another in a group setting.
- **Read/write (R):** The learners who prefer this modality best understand information displayed as words. They strongly prefer text-based learning materials. They are also able to read widely and write the material learned in a structured form.
- **Kinaesthetic (K):** Kinaesthetic learners can learn better through performing the required tasks. They prefer to use their hands and body movements. Learners with this particular learning style tend to gain knowledge via demonstrations, simulations, videos and movies of “real” things, as well as case studies and practice. They are also good in applying the concepts in real-life scenarios. When kinaesthetic learners are studying, they focus in doing practical problems instead of reading over a text book.

2.4 Item Response Theory

In education there are many situations in which there is the need to deal with the measurement of fundamental variables such as intelligence (Baker, 2001). Usually, these variables, called latent trait, are not measured directly and are understood intuitively. Similarly, the learner's ability and its signs such as achieving good grades, learning new concepts easily and using time effectively, are identified as latent traits. Although these variables can be easily defined, their measurement (such as height and weight) is not directly possible. The main purpose of the item response theory is to measure learner's latent trait. These latent traits are generally called *ability* in item response theory (Baker, 2001).

Item response theory (Baker, 1992) is a model-based approach to select the most appropriate items for learners based on mathematics relationship between abilities and item responses. It is called Item Response Theory because the theory focuses on the item, by modelling the response of a learner of given ability to each item in the test. The idea of IRT is based on the assumption that the probability of a correct answer to an item is a mathematical function of person and item variables. The person variables, which are the human capacity or attribute measured by the test, might be a cognitive ability, physical ability, skill, knowledge or personality characteristic. The item variable is referred to as the item difficulty, item discrimination, and the effect of random guessing.

The assumption is that the learner's ability can be measured based on a scale of real numbers between positive and negative infinity, with zero as the midpoint and one as the unit of measurement. If we can physically measure the learner's ability then we can compare the ability of several learners with each other. While in theory, learner's ability can range from negative infinity to positive infinity it will however be restricted to -3 and $+3$ for some practical considerations.

In order to measure learner's ability, a test consisting of several items is developed. After attempting all the items in the assessment, the system receives learner's responses and scores the items dichotomously. This means that the learner gets one for a correct answer and zero for an incorrect answer. Each learner has a unique ability which is represented by a numerical value on the ability scale. This ability is symbolised with θ . Furthermore, in each level of ability, there will be a probability that a learner with this ability responds

correctly to this item. (There will be the probabilities of giving the correct answer across different levels of ability.) This probability is shown be $P(\theta)$. This probability is large for a learner with high ability and is small for learners with low ability. Item Characteristic Curve (ICC) presents the relationship between probabilities and abilities as shown in Figure 2.1(Yu, 2007).

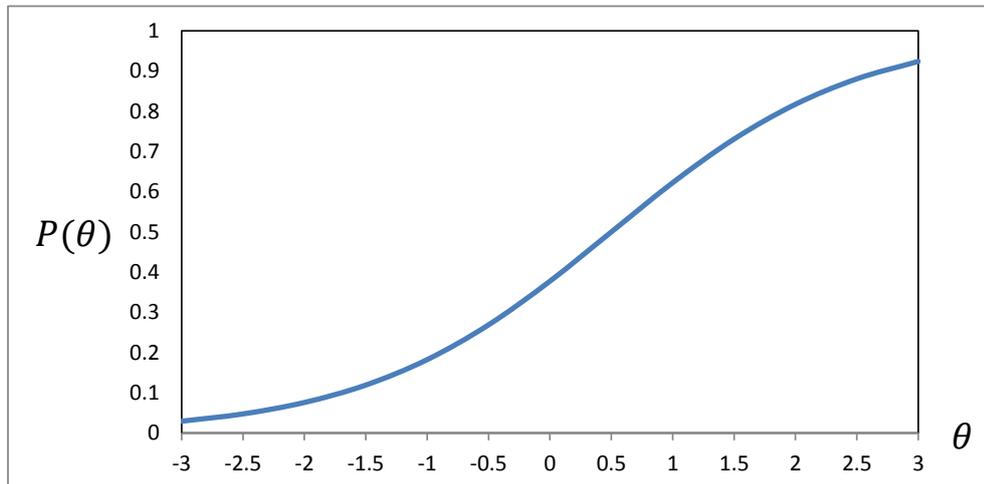


Figure 2.1: Item Characteristic Curve (ICC)

As the figure shows, the probability of having a correct answer in the lowest level of ability is nearly zero. As the ability increases, the probability also increases until it reaches its highest level. At which point, the probability of having a correct answer will be nearly one.

Each item in test has its own Item characteristic curve. However, the shape of classic Item Characteristic Curve is s-shape. The Item Characteristic Curve is the basic building block of Item Response Theory and other components of theory build on this curve(Baker, 2001).

Item Characteristic Curve has two technical attributes. The first is the difficulty of an item which describes the position of ICC in relation to the ability scale (Hambleton et al., 1991) and the second is discrimination parameter which discriminates between high-proficient learner and less-proficient learner (Yu, 2007). The slope of Item Characteristic Curve reflects the discrimination parameter (Baker, 2001). The steeper curve demonstrates a much better discrimination than the flatter curve.

2.4.1 Item Characteristic Curve Models

Under item response theory, the standard mathematical model for the item characteristic curve is the cumulative form of the logistic function (Baker, 1992). These models are used to calculate the probability of a correct answer in a different level of ability. There are three common mathematical models for Item Characteristic Curve according to the number of parameters in logistic function; one Parameter Logistic function (1PL), Two Parameter Logistic function (2PL) and Three Parameter Logistic function (3PL) models (Wang, 2006).

In the 2PL model each item i is characterised by two parameters. Based on this model, the difficulty and discrimination parameters can take on different values. The equation for this model is given by the following:

$$P_i(\theta) = \frac{1}{1 + e^{-a_i(\theta - b_i)}} \quad (2.1)$$

Where:

θ is the ability level of learner

b_i is the difficulty parameter of item i ($-3 \leq b_i \leq +3$)

a_i is the discrimination parameter of item i ($0 \leq a_i \leq +1.7$)

$P_i(\theta)$ is the probability that learner with ability θ can response correctly to the item i

The slope of item characteristic curve is changed as a function of ability level. It reaches its maximum value when the ability level equals the item's difficulty. Therefore, the slope of item characteristic curve, at the point which $\theta = b$ is equal to $a/4$. Figure 2.2 shows the item characteristics curve for two parameter model with $b = 1$ and $a = 1.5$

The 1PL model or Rasch model was first introduced by Danish mathematician Georg Rasch. Based on this model the value of discrimination degree of the 2PL model for all items is a constant value of 1. It means that only the difficulty parameter can get different values. The equation for this model is given by the following:

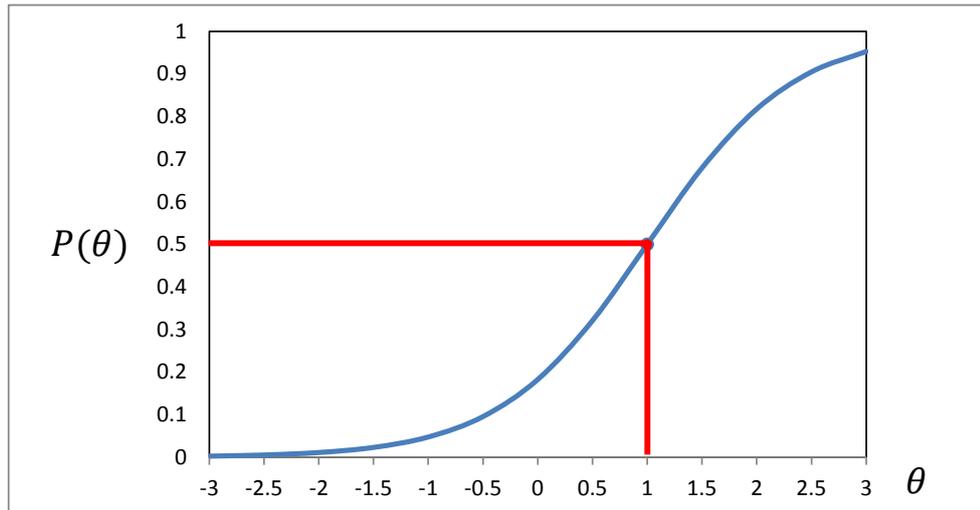


Figure 2.2: Item Characteristic Curve for a two parameter model with $b=1$ and $a=1.5$

$$P_i(\theta) = \frac{1}{1 + e^{-(\theta - b_i)}} \quad (2.2)$$

In testing there is a small probability that learners can get a correct answer by guessing. Thus, a guess degree c_i is added to the 2PL model and resulting model has been known as the 3PL model. The equation for the 3PL model is:

$$P_i(\theta) = c_i + (1 + c_i) \frac{1}{1 + e^{-a_i(\theta - b_i)}} \quad (2.3)$$

The parameter c shows the probability of getting a correct answer for the item only by guessing. As the value of c is not dependent on ability level, thus, the probability of getting a correct answer by guessing for learners with different (who have highest or lowest) ability level are equal. In theory, the parameter c has a range of $0 \leq c \leq 1.0$, but in practice values above 0.35 are not considered.

2.4.2 Estimate Learner's Ability

The main purpose of applying a test to a learner is estimating her ability. In order to estimate a learner's ability, it is assumed that the values of different parameters for all items in the test are known. After the test is administered and the learner answers all the items, the responses will be dichotomously scored. This means that the learner gets one

for a correct answer and zero for an incorrect answer. Hence, we will have a response pattern $(U_1, U_2, U_3, \dots, U_j, \dots, U_n)$ which is called the test response vector, where $U_j = 1$ represents a correct answer given by the learner for the j^{th} item in the test. On the contrary, $U_j = 0$ represents an incorrect answer given by the learner to the j^{th} item in the test. Under item response theory, maximum likelihood Estimator (MLE) is used to effectively estimate learner's ability (Hambleton et al 1991). This assumption enables an estimation method called maximum likelihood estimator (MLE) to effectively estimate item parameter and learner's abilities (Hambleton et al 1991). MLE is an iterative procedure. It gets the items' parameters and a priori value for the learner's ability. After that, the probability of the correct response to each item is calculated for the learner based on logistic function. Then, according to the calculated probability, the MLE procedure estimates the new ability of the learner. The procedure is repeated until the change in the estimated ability is less than a threshold value (i.e. becomes stable). The estimation equation is as follows:

$$\theta_{s+1} = \theta_s + \frac{\sum_{i=1}^N -a[u_i - P_i(\theta_s)]}{\sum_{i=1}^N a_i^2 P_i(\theta_s) Q_i(\theta_s)} \quad (2.4)$$

Where:

θ_s is the estimated ability of the learner within iteration s

a_i is the discrimination degree of item i

u_i is the response made by the learner to item i:

$u_i = 1$ for a correct response

$u_i = 0$ for an incorrect response

$P_i(\theta_s)$ is the probability that learner with ability θ can response correctly to item i within iteration s.

$Q_i(\theta_s) = 1 - P_i(\theta_s)$ is the probability that learner with ability θ responses incorrectly to item i within iteration s.

According to Thompson (2009) the purpose of IRT is to provide a framework in order to evaluate how well tests and individual items work. Thompson indicates that in education

IRT helps researchers in three ways:

- Developing and refining exams
- Selecting appropriate items for exams
- Estimating a learner's latent ability

As we know, in classical test theory, the learner gets a low score on a difficult exam and a high score on an easy one. Therefore, the result of the exam cannot be relied upon to prove the underlying ability of the learner. However, under item response theory, as the values of all the item parameters are in a common metric and they measure the same latent trait, the learner's ability is invariant in different exams. In other words, if a learner takes a hard or an easy exam, she obtains the same estimated ability. On the other hand, if a remedial learning is delivered to the learner, the learner's ability level would be changed in the exam. Accordingly, the learner's ability level may change. Consequently, based on the estimated ability, the learner model is updated and in the next step of the learning process an IO is recommended to the learner according to her new ability.

2.4.3 Test Calibration

To estimate a learner's ability, it will be assumed that the numerical values of the parameters of the test items are known. It means, the numerical parameters of the items in the assessment are known. Consequently, the metric of the ability scale will be the same as the metric of the known item parameters. Although this assumption has been beneficial for introducing the fundamental concepts of IRT it cannot state the actual testing situation. In IRT, an important task is to determine the value of item parameters and learner's ability in a metric for underlying latent trait, which is called test calibration. The aim of test calibration is to provide a reference framework to interpret the result of test.

For calibrating items, the test developer designs a suitable test with N items. Then the test is presented to a group with M students to answer them. After doing the test and receiving the response of the learner to all items of test, the collected items responses are scored dichotomously. This means that the learner gets one for the correct answer and zero for the incorrect answer. After that, an ability scale can be created by applying a mathematical algorithm on the item response data which is unique to the set of students and the items of

the test. Then the value of item parameters and the estimated learners' ability are stated in this unique metric.

The technique which is used for calibrating a test was introduced by Birnbaum and has been implemented by some computer programs such as LOGIST and BICAL. This technique is an iterative method which uses two stages of maximum likelihood estimation. In one stage it estimates the parameter of N items and in the other it estimates the ability of M learners. These two stages are executed iteratively until a stable set of estimated parameters are determined. Therefore, a calibrated test and an ability scale are defined. This means that, each item has its own difficulty, discrimination, and guessing Parameter which can be used to estimate learner's ability.

Item Response Theory is used in the Computerised Adaptive Test (CAT) to determine the best items for examinees based on their abilities. Currently, the CAT concept has been successfully used in many real applications such as GMAT, GRE and TOEFL.

Researchers have applied this theory for constructing e-learning systems in different ways. Some of these applications are as follows: using IRT for self-assessment in adaptive e-learning system (Guzman & Conejo, 2005), constructing a personalised e-learning system based on IRT which considers learner's feedback and difficulty level of course materials (Chen et al., 2005), proposing personalised curriculum sequencing using modified IRT (Chen et al., 2006), developing intelligent tutoring system based on fuzzy item response theory (Chen & Duh, 2008) and recommending a genetic-based curriculum sequencing based on the evolution technique through computerised adaptive testing (CAT) (Huang et al., 2007).

2.5 Summary

This chapter reviewed a number of learning theories and their influences on adaptive e-learning systems. In addition, since learning styles have significant influence on the way learners perceive new information and construct their knowledge, we also reviewed some main learning style models that may be pertinent for personalisation. Finally to promote learning effectiveness it is important to consider learner's ability. Therefore, an overview of item response theory was presented in this chapter as a way to estimate learners' ability

accurately. These reviews are important when proposing new approaches for supporting an adaptive e-learning system as knowing about different pedagogical approaches has influence in successfully adopting the appropriate ones.

Chapter 3

Adaptive E-Learning Approaches

3.1 Introduction

Adaptive e-learning is proposed to make an e-learning system more effective by adapting the presentation of information and overall linkage structure to individual learners based on their knowledge and needs. Each learner has different learner's characteristics; thereby diverse educational settings may be more suitable for one group of learner than for another. Adaptive e-Learning systems try to tackle this issue by acquiring knowledge about a particular learner and adapt the learning path and learning to the given learner. The aim of adaptive e-learning is to provide appropriate information in order to optimize the learning outcome.

This chapter presents an overview on adaptive hypermedia. It also reviews different methods of adaptivity and the way they can be applied in e-learning systems. This chapter also reviews semantic modelling using semantic web technologies, in particular ontology for presenting elements personalisation on e-learning systems, and concludes with surveys on current personalised e-learning systems.

3.2 Adaptive Hypermedia

Hypermedia systems are environments which supply an experience for users composed of free navigation in a large hyperspace of information. However, a problem with these

systems is that users can easily get disorientated when looking for information and lose their sense of location due to cognitive overload. Adaptive Hypermedia (AH) systems attempt to alleviate these difficulties by personalising the presentation aspect of the system and supporting users in their navigation. This can be achieved by building a model of needs, preferences and knowledge for each individual user, thus adapting the information and linkage of hypermedia pages to the profile of users (Brusilovsky, 1996).

AH can be particularly beneficial in application areas where the domain is reasonably large and is used by different users with diverse needs and background knowledge (Brusilovsky et al., 2012). People are usually interested in different pieces of information that correspond to their particular needs and may use different links in order to navigate to corresponding pages. AH systems can assist users in their navigation by referring to their corresponding user model which consists of user's knowledge, needs and preferences, thus adapting information and links for the user (Brusilovsky, 1998).

Adaptive systems collect information about its users and adapt the system's behaviour accordingly. However, in general, there is the possibility that the system makes mistakes in its attempts in understanding the user's preferences, therefore there is the need for the user to have some level of control on the adaptivity of the system. On one hand, we have systems that allow the user to have full control on some or all aspects of the system's adaptivity which are called adaptable systems. On the other hand, there are systems which provide the adaptivity solely based on the assumptions they hold about the user's preferences. These systems are called adaptive systems (Fink et al., 1996). Researchers have suggested that by balancing the adaptivity control between the system and the user, the benefits of each of the aforementioned approaches will be combined.

In the following section we will present the methods of adaptation in detail.

3.3 Methods of Adaptation in Educational System

There are two main techniques that adaptive systems use to perform adaptation: adaptive presentation and adaptive navigation support (Brusilovsky, 2001*b*). Using adaptive presentation, the system may display different pieces of content for different users. In adaptive navigation, the presentation of links is adapted to the user.

3.3.1 Adaptive Presentation

Adaptive Presentation tailors the content for the user by matching it with learning characteristics specified in the user's model. In a system with adaptive presentation, instead of using static pages, pieces of information are integrated upon user's request to generate pages adapted to each individual user. For example, novice users are given extra explanation while advanced users are given deeper and more detailed information (Brusilovsky, 1998). Depending on variables in the user's model, the length of the presentation, the difficulty level of the presented content and the media type varies.

The process of adaptive presentation can be divided into two main sub processes. The first one involves selecting and coherently structuring the most relevant content to user's interests. The second one involves adapting the media type in order to effectively present the selected content to the user (Bunt et al., 2007).

The granularity of selecting relevant content may vary from a fragment of information to a page. A fragment of information is the smallest self-contained instructional unit serving an independent pedagogical role such as a paragraph or a picture. In page-variant, the adaptation mechanism selects one page, most appropriate to the current interaction context to be presented. Consequently, having only a few versions of the same page is feasible. As an example, page variants are used in KBS (Knowledge Based Systems) Hyperbook system (Henze & Nejdil, 2000) to develop educational courseware for Java programming. Fragment-variant technique performs adaptation at a finer level of granularity. More specifically, the page presented to the user is not selected from a pool of fixed pages. Rather, it is constructed by performing some manipulations to fragments such as adding, removing, resorting, alternating or dimming them. Therefore, systems adopting this approach are able to form many versions of the same page. For instance, in AHA (Adaptive Hypermedia for All) (De Bra et al., 2003), fragment-variant technique is used which enables it to present the same entity in different forms based on user's background knowledge.

3.3.2 Adaptive Navigation

Adaptive Navigation support techniques guide users to find the most relevant information by adapting link presentation to users' needs (Brusilovsky, 2007). The most popular

techniques to adapt presentation of links are direct guidance, adaptive link sorting, adaptive link hiding and adaptive link annotation. These techniques change the structure and presentation of links, as the main navigational means, throughout the learning environment; in contrast to adaptive presentation which adapts the content presented to the learner.

Direct guidance is the primary technique of adaptive navigational support in which the learner is shown the best possible link(s) in any given situation, based on the information in the user profile. This technique is criticised for being somewhat limited as it does not support users who would not want to follow the system's suggestions. Therefore, user's freedom in navigation, which is one of the advantages of hypermedia systems, is lost (Conlan, 2006). Although this technique has been widely used in early educational AH systems such as ELM-ART (Brusilovsky et al., 1996), or InterBook (Brusilovsky et al., 1998), due to its limitation it is mostly replaced by other adaptive navigation support techniques.

Link ordering is a technique in which all the links are sorted according to their relevance to a user model. The system orders the links in a way that the more relevant a link, the higher its place in the list. In other words, the set of links in a page is not fixed and may change each time the user visits a page based on various criteria. This technique is used in HYPERFLEX (Kaplan et al., 1993) and Adaptive HyperMan (Math & Chen, 1996). The problem with this technique is having non-stable link order which leads a novice user, with poor domain knowledge, to disorientation.

Link hiding tries to prevent the user from following irrelevant paths by hiding, removing or disabling links to unrelated pages. Links may be removed completely or disabled to prevent users from accidentally accessing them (De-Bra & Calvi, 1998). In link hiding, the cognitive load on the user can be reduced by restricting navigational spaces. It also supports a stable ordering by adding links to existing navigational structures. Adaptive hiding techniques have been used and explored mainly in educational hypermedia systems like the ISIS-Tutor (Brusilovsky & Pesin, 1998). It shows fewer links when the student begins to interact with the system but after developing the learner's knowledge on the subject, the system gradually increases the number of visible links. However, this technique limits the user's ability to create a correct mental map from the content structure as the user would not know, from the beginning of her learning experience, how the content is

entirely structured.

Link Annotation is the technique in which extra information is added to a link, for example applying visual cues, in order to inform the user about the content behind the link (Brusilovsky, 2007). It can be utilised in different ways such as using different icons, colours, and text sizes for the links to advise the users on the relevancy of the linked content to the user. Most of the current web browsers support link annotation by using link states with two values: unvisited links (usually in blue) and visited links (usually in purple). AHA (De Bra et al., 2003) expands on this idea to define links with three states: suitable to be viewed, unsuitable to be viewed and visited links. Adaptive systems can support links with many states such as visited, unvisited, current and suggested (Conlan, 2006). Link annotation defines a stable order for the links and prevents the users from making incorrect mental maps. However, it does not prevent cognitive overload, but such systems can support link dimming by presenting certain links with a lower opacity level to offer the effect of hiding without restricting the user by completely hiding it.

3.4 Adaptive Educational Hypermedia systems

From the very early days of AH systems, education was one of the most promising application areas. In an educational context, users with alternative learning abilities and background knowledge essentially require different learning paths (Brusilovsky et al., 2012). “AH Systems make a connection between computer driven tutoring systems and learner driven educational environments” (Conlan, 2000) to prepare a personalised adaptive learning experience for learners.

Adaptive educational hypermedia (AEH) systems are able to tailor learning content and learning paths to individual’s abilities, preferences, needs and knowledge in order to enhance the learner’s understanding of the learning content (Conlan, 2006; Berlanga & nalvo, 2008; Chellatamilan & Suresh, 2011; Ghazal et al., 2011; Idris et al., 2009; Liu et al., 2009; Meccawy, 2009). From the implementation aspect, these systems tend to separate the learner model from the domain model (Conlan, 2006). The learner model stores various characteristics of individual users as attribute-value pairs for each learner and learners’ profiles are updated according to their interactions with the system. The domain model

represents the structure of a knowledge domain as a set of concepts or knowledge elements (Ghazal et al., 2011; Brusilovsky & Peylo, 2003; Mulwa et al., 2010). The information in domain and learner models together with a set of prerequisite relationships allow the AEH systems to decide whether a learner is ready to be introduced to a new concept or not. Additionally, these systems apply the annotated links technique to inform the learner about the content beyond the links, whether it is what they already know, new knowledge which they do not know, or advanced knowledge that the user is not necessarily ready for at that level. Therefore, the learner will be able to make informed decisions regarding where they navigate.

3.4.1 Different Generation of AEH Systems

From the developer's point-of-view adaptive educational hypermedia systems can be classified into three generations (Meccawy, 2009). The first generation of AEH systems were pre-web systems developed between 1990 and 1996 by two groups of researchers (Brusilovsky, 2001a). The first group were working on intelligent tutoring systems (ITSs). They tried to append hypermedia components to students' profiles and adaptation approaches which were developed in this field. The second group of researchers were working on educational hypermedia and tried to add adaptivity features to their systems. Examples of this generation's systems includes Anatom-Tutor(Beaumont, 1994), Hypadapter (Böcker & Hohl, 1990), MetaDoc (Boyle & Encarnacion, 1994), KN-AHS and C-book (Kay & R.J.Kummerfeld, 1994).

The majority of second generation of AEH systems were developed between 1996 and 2002. Since 1996, due to the rapid increase in the use of the World Wide Web (WWW), many researchers focused on creating Web-based educational systems. Since a huge number of users with different abilities and needs were working with web-based educational systems, the need for personalisation would clearly follow. Therefore, web-based educations were the main factor for driving second generation of adaptive educational hypermedia systems (De-Bra & Calvi, 1998), and Multibook (Seeberg et al., 1999). The works of researchers who developed the second generation of AEH systems can be roughly divided into three groups:

1. First were those who worked on web-based education. They created web-based

educational systems with elements of adaptive hypermedia. The focus of this group was on producing teaching systems and not to adopt new technology. They instead reused pre-existing technologies on various subject areas.

2. The second group of researchers were those working in the field of ITS or adaptive hypermedia. They focused more on producing new techniques for adaptive hypermedia and new technologies. For example, AHA! (De-Bra & Calvi, 1998) applied several approaches to *link removal* or MANIC (Stern & Woolf, 2000) proposed innovative approaches for user modelling and adaptive presentation.
3. The work of the third group was focused on developing authoring tools and frameworks for adaptive educational hypermedia. The framework offers a generic architecture which is reusable across a range of adaptive systems with low overheads.

The second generation systems have not made the jump into practical web-based education for after 10 years only a few of them were used for teaching courses. Therefore, the current third generation of adaptive hypermedia research is motivated by the challenges concerning the integration of adaptive hypermedia technologies into the regular educational processes. Consequently, different groups of researchers focused on different research directions to dominate Learning Management Systems (LMS) (Watson et al., 2007). However most recent projects do not focus on promoting the current LMS's, but tend to upgrade them with more modern LMS's which are based on system interoperability and content reusability. They focused on using standard-based mechanisms in existing adaptive hypermedia technologies to facilitate the reusability of learning resources (Conlan et al., 2002; Morimoto et al., 2007). However, some researchers argue that the current generation of standards are insufficient in supporting the demands of adaptive learning (Rey-lópez et al., 2008). Therefore, in this research a different direction has been adopted in attempting to model and represent the knowledge through semantic web technologies such as Resource Description Framework (RDF) and ontologies to provide improved interoperability, explicit semantics, formal representation and formal reasoning (Dolog & Nejd, 2007). Yet, this direction requires a lot of effort and consideration to promote the flexibility and extensibility of systems and reusability of content.

In the next section the current standards and specifications for metadata on representing learners and digital learning resources will be reviewed. In addition, the limitations of

these standards in representing the features of adaptation will be discussed.

3.5 Specifications and Standards for Metadata

Metadata specifications and standards are developed to support description, packaging, sequencing and delivering of learning resources, learning activities and learners' profiles. Currently, a number of standards have been defined by different organizations such as IEEE Learning Technologies Standardization Committee (IEEE LTSC)¹, Instructional Management System Global Learning Consortium (IMS GLC)², Aviation Industry CBT Committee (AICC)³, Advanced Distributed Learning Initiative (ADL)⁴, the Dublin Core (DC) Metadata Initiative⁵.

The following are the most famous metadata standards for representing Learning Resources, Learner Models, Packaging and Managing Content.

- IEEE LOM (Duval, 2002) is developed by IEEE LTSC Learning Object Metadata Working Group which focuses on detailed descriptions of Learning Objects. It consist of 9 different categories: General, Technical, Meta-metadata, Life Cycle, Educational, Rights, Annotation, Relation and Classification.
- IMS Learning Resource Metadata Specification (Barker et al., 2004) is a standard derived from the above mentioned IEEE LOM specification version 3.5. IMS produces the best practical implementation guide and XML bindings.
- Dublin Core Metadata Initiative (DCMI) (DCMI, 2012) is developed to reach a core set of metadata for describing internet-based information resources. DC meta-data provides a set of simple and flexible elements to facilitate sharing, describing, finding, and managing information. One of the goals of Dublin core is to facilitate interoperability. The DC contains fifteen basic DC metadata elements: Contributor, Coverage, Creator, Date, Description, Format, Identifier, Language, Publisher, Relation, Rights, Source, Subject, Title and Type (Dublin Core, 1999). DC is being

¹ [http:// www.ieeeeltsc.org](http://www.ieeeeltsc.org)

²<http://www.imsglobal.org/>

³<http://www.aicc.org/>

⁴<http://www.adlnet.gov>

⁵<http://dublincore.org>

adopted by many communities because of its simplicity and flexibility.

- IMS Learner Information Packaging (LIP) (Smythe, 2005) specification includes information about a learner or a producer of learning content. The IMS LIP enables the interoperability of learner information between different Internet-based systems (LMS's). In the IMS LIP, learner information is divided into different segments: identifications, goals, qualifications, activities, interests, competencies, accessibilities, affiliations, relationship, security keys and transcripts.
- The IEEE Public and Private Information (PAPI) specification (Farance, 2001) is develop to represent a student's record and to support the exchange of learner's profile between different systems. Profile information of a learner is divided into four areas: Personal Information, Preference, Performance Information and Portfolio.
- IMS Content Packaging (Smythe & Jackl, 2004) describes data structures that can be used to exchange data between systems via content creation tools, learning management systems, and runtime environments. This specification provides a mechanism for structuring learning content in order to be imported, exported, aggregated and disaggregated between systems with minimum effort.
- The Sharable Content Object Reference Model (SCORM) (ADL, 2004) is a set of technical standards and specifications for web-based e-learning content sharing. The main goal of the SCORM is to enable interoperability between different LMS through packaging content in zipped files, thus supporting accessibility, reusability and durability of learning content. SCORM 2004 introduced a sequencing attribute that specifies the order in which a learner may experience learning objects. SCORM is a collection of standards and specification based on XML.

The goal of using metadata standards for representing a learner or contents is to facilitate searching and retrieval of learning resources (i.e. LOs) and learner's profile in order to support their reuse. However, reusability involves a wider set of issues with regards to the content, context, pedagogy and learner that the standard metadata cannot express sufficiently. As a consequence, standard metadata does not capture enough information necessary for advanced levels of personalisation as they are developed primarily with LMS's in mind. For instance, IEEE LOM metadata, the most prominent standard metadata, contains over 80 different metadata constructs. However, only a few of these constructs

support the creation of personalised learning (Brooks et al., 2006). Accordingly, many researchers argue that current standards are unable to support the needs for adaptive learning (Rey-lópez et al., 2008; Mödritscher et al., 2004).

The aforementioned standards do not support the following requirements which are essential for creating effective personalisation:

- Representing fundamental information relevant to the pedagogical role of each content unit (Ullrich, 2005).
- Accessing learning content, which comprises the structure of a course in a lower granularity level (Jovanović et al., 2009).
- Presenting characteristics of learners which are required for specific learning designs (Gašević et al., 2008).

In order to overcome the above limitations, in this research we focus on utilising Semantic Web technologies - ontologies in particular - for representing knowledge, formally modelling components of personalisation, and reasoning about conceptual knowledge (Dolog & Nejd, 2007; Henze et al., 2004; Aroyo et al., 2002; Mohan & Brooks, 2003; Gašević et al., 2007). Therefore in the next section we will review the semantic web technologies in particular ontology and its use in modelling the components of personalisation. Moreover, reviewed in this section are the most important ontology languages for representing knowledge such as Resource Description Framework (RDF) and Ontology Web Language (OWL).

3.6 Semantic Modelling using Semantic Web Technologies

The Semantic Web is considered as the next generation of the Web where information is given “a well-defined meaning”, better enabling computers and people to work in cooperation (Berners-Lee et al., 2001). Semantics enable computers to process, transform and assemble information to make smarter decisions. Several technologies have been developed for shaping, constructing and developing the semantic web. Such technologies support fine grained semantic structures for web resources (Dolog, 2006). Nowadays, these

technologies are being applied in many practical applications to semantically model the knowledge in their respective domains. In the field of personalised e-learning, ontologies are applied to model knowledge about learning content, learner's profile and teaching strategies.

Figure 3.1 shows the semantic web stack, which illustrates the hierarchy of the involved technologies (Berners-Lee, 2006). The technologies from the bottom of the stack up to OWL are currently standardised.

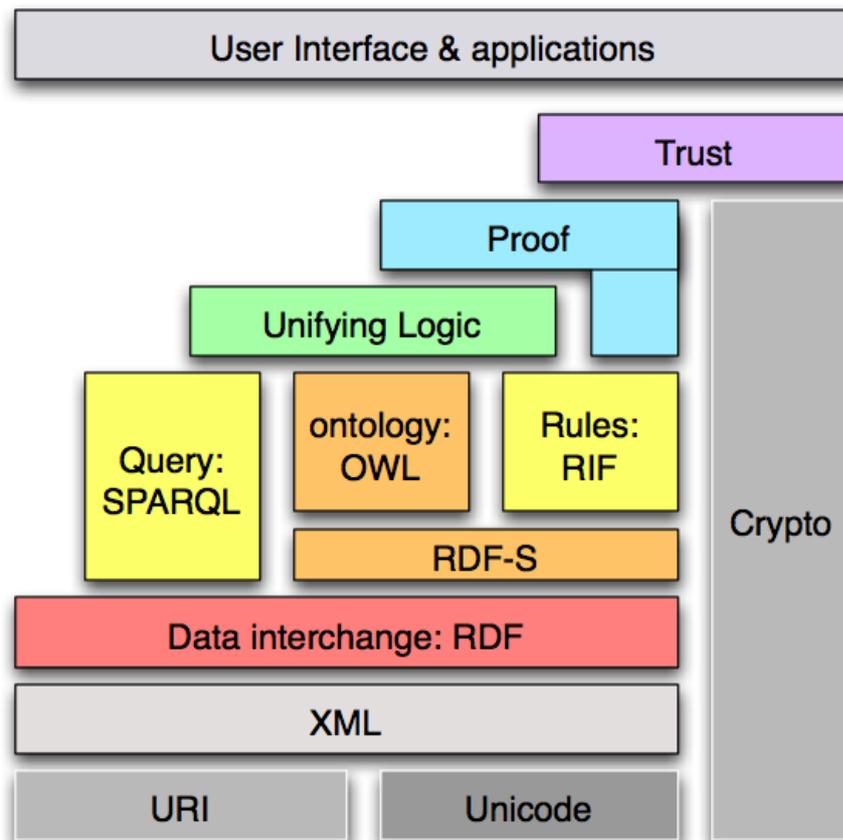


Figure 3.1: Semantic Web Stack (Berners-Lee, 2006)

Figure 3.1 guides the description of standards associated to each of the semantic web stack layers. Each layer builds on the lower layer. The first two layers provide a common syntax corresponding to the information and data view while the next three layers add the semantics to the Web and allow inferring new knowledge from the explicitly provided information. The layers above those, which are not standardised yet, are not directly relevant to this research. Short definition of technologies shown in the five first layers is as follows:

- The URI (Uniform Resource Identifier) layer provides a global standard to uniquely identify semantic web resources. Unicode is considered as the global standard encoding system for computer character representation. It serves to represent text in many human languages (Medić & Golubović, 2010).
- XML is a mark-up language used to describe information in all the upper layers. It is widely known in the World Wide Web community. This is due to having a flexible text format and providing a uniform representation and presentation of documents.
- The Resource Description Framework (RDF) is the first layer of the Semantic Web. RDF is a framework for describing the Web resources and representing information in the web. RDF is a method to create statements in the form of triples.
- RDF Schema (RDFS) provides basic vocabulary for RDF. It provides a set of classes and properties to structure RDF resources.
- Ontology web Language (OWL) is an extension of RDFS and allows more expressivity. It adds more advance constructs to describe semantics of RDF statements. It provides reasoning power to the semantic web based on description logic.
- SPARQL is a query language for RDF to retrieve and manipulate information stored in RDF formats.
- RIF or SWRL are rule languages. They express relations that cannot directly be described using OWL.

As we explained above, ontology is one of the essential components of the semantic web technologies which support modelling and reasoning of specific domain knowledge. Research shows that semantic web technologies can also be used to address the personalisation decisions in e-learning systems by providing formalisations on domain ontologies and the metadata created from them (Dolog & Nejdli, 2007; Jovanović et al., 2009; Henze et al., 2004; Jia et al., 2011; Vesin et al., 2012).

In personalised e-learning systems, it is essential to have knowledge about learners, content and adaptation strategies and to identify how this knowledge should be represented in order to generate optimise learning paths. An ontology-based semantic model provides high level modelling capabilities to represent major components of personalisation in e-learning systems and also provides reasoning mechanisms to accomplish further semantic

enrichment steps that can perform the adaptation process. Therefore, it is evident that domain, learner and adaptation models are major components of personalisation in e-learning systems. Domain models can describe both the semantics and structure of learning contents. Learner's characteristics necessary for the personalisation are retained in the learner model. Adaptation models contain a set of adaptation strategies that describe the details of the educational process (e.g. to suggest a document for learning, or to generate reasonable learning paths, etc.). In the next section we present an overview of ontology and its languages as a background to explain its applications in personalised e-learning.

3.6.1 Ontology

Ontology is a branch of philosophy where it refers to the theory of existence (Smith, 2003). The study of ontology goes back to the works of Plato and Aristotle which consists of hierarchically categorising different kinds of entity and their features (Horrocks, 2008). Ontology has been widely used in recent years in the field of artificial intelligence and computer science, especially in domains such as intelligent information integration (Seng & Kong, 2009), cooperative information systems (Ouksel & Sheth, 1999), knowledge representation (Brewster & O'Hara, 2007), information retrieval and extraction (Müller et al., 2004), and database management systems (Necib & Freytag, 2003; Snae & Brüeckner, 2007).

Available literature on ontological engineering points to a number of definitions for describing what ontology is. One of the most widely used definitions of ontology is by Gruber (Gruber, 1993) where Ontology is defined as a formal, explicit specification of a conceptualisation. *Conceptualisation* emphasises that ontology represents an abstract model of a phenomenon in the world as it helps to identify appropriate domain concepts and semantic relationships among these concepts with formal definitions in terms of axioms (Chi, 2009). Such axioms are declarative which enables ontology to represent the conceptualisation declaratively. *Explicitness* in the ontology definition states that the type of concepts and the constraints on their use are defined explicitly. *Formality* means that the ontology prevents unexpected interpretation of the concept, relations and constraints. Therefore, it enables ontology to be machine-readable. Studer (Studer et al., 1998) defines ontology as a formal, explicit specification of a shared conceptualisation. *Shared* means

that an ontology captures agreed knowledge; it is not private for some individual and accepted by a group (Studer et al., 1998).

Knowledge representation using ontologies facilitates organising the metadata of complex information resources. These metadata provide syntactic and semantic information about information resources which are encoded as instances in the ontology (Sheth et al., 2002). Formal representation of ontologies and metadata created from them enables reasoning for the purpose of accessing inferred knowledge (Dolog & Nejd, 2007).

In addition, ontologies can represent a conceptual model in a domain by describing the main concepts of domain and their relationships through defining classes and properties and providing semantic constraints among concepts through defining various axioms (Sheth et al., 2002). For example, a part of an ontology in an educational domain is provided in Figure 3.2 in which *user*, *learner*, *lesson*, *content unit* and *topic* are concepts. It is specified not only that a *learner* is a *user* and that a *content unit* is a part of *lesson*, but also that a *content unit* has a *topic* and is suggested to a *learner*. Different relations between concepts are defined in this example. For instance, a hierarchical relation is defined between *user* and *learner* (i.e. *is a*) which specifies that all learners are also a user. Additionally, the figure shows that *content unit* has different relations with different concepts.

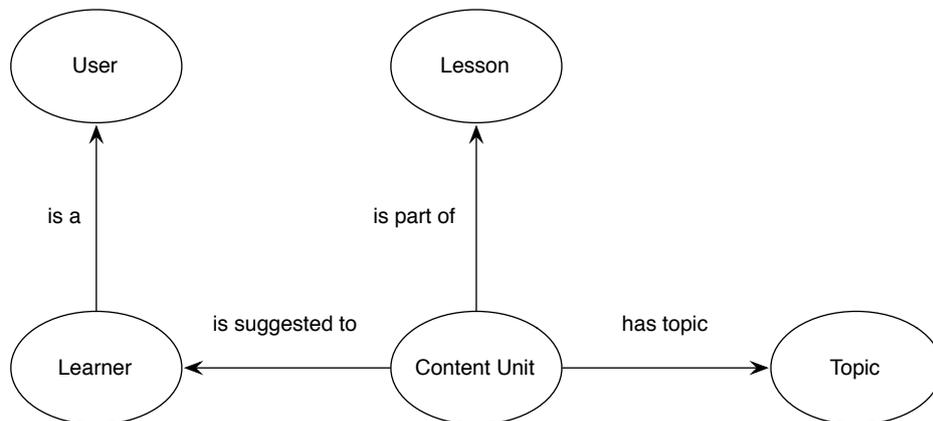


Figure 3.2: Example of a small ontology

This makes sure that the specification of the domain knowledge in the ontology can be logically interpreted in an appropriate way to enable automatic reasoning over the explicit knowledge of the domain. So the essential benefits of ontologies are:

- Encoding knowledge about specific domains.

- Advanced analysis and reasoning.

For example, we know that all learners are users; and if Michael is a learner, then we can deduce that Michael is a user. In predicate logic this information is expressed as follows:

```
Learner(X) → User(X)
Learner(michael)
∴ User(michael)
```

As it can be seen in the above example, logic can be used to expose implicit ontological knowledge in addition to exposing unexpected relationships and inconsistencies.

3.6.2 Ontology Languages

An ontology language is a formal language for encoding an ontology. They are the foundation of ontological systems which allow for constructing knowledge about specific domains and often include reasoning rules that support processing of that knowledge.

Several ontology languages have been developed during the last few years such as Knowledge Interchange Format (KIF)⁶, Cyc1⁷, FLogic (Kifer et al., 1995) and LOOM⁸ (based on description logic). There are however some other languages which are based on XML syntax such as Ontology Exchange Language (XOL)(Karp et al., 1999), SHOE⁹, Resource Description Framework (RDF)¹⁰ and RDF Schema¹¹. Finally, three additional languages have been developed on top of RDF(S) to improve its features: Ontology Inference Layer (OIL) (Fensel et al., 2000), DAML+OIL (Horrocks, 2002) and OWL (Patel-Schneider et al., 2004).

Most recent ontology developers have used graphical ontology editors for creating or manipulating ontologies. These editors prevent the developers from having to manipulate ontology language codes. The output of these editors will be in one of the web ontology languages supported by ontology editors. Some of the more popular ontology editors are Protégé (Noy & Musen, 2000)(see Figure 3.3), OWL-P (Desai et al., 2005) and OilEd (Bechhofer et al., 2001).

⁶<http://suo.ieee.org/SUO/KIF/suo-kif.html>

⁷<http://www.cyc.com/>

⁸<http://www.isi.edu/isd/LOOM/>

⁹<http://www.cs.umd.edu/projects/plus/SHOE/spec.html>

¹⁰<http://www.w3.org/TR/PR-rdf-syntax/>

¹¹<http://www.w3.org/TR/rdf-schema/>

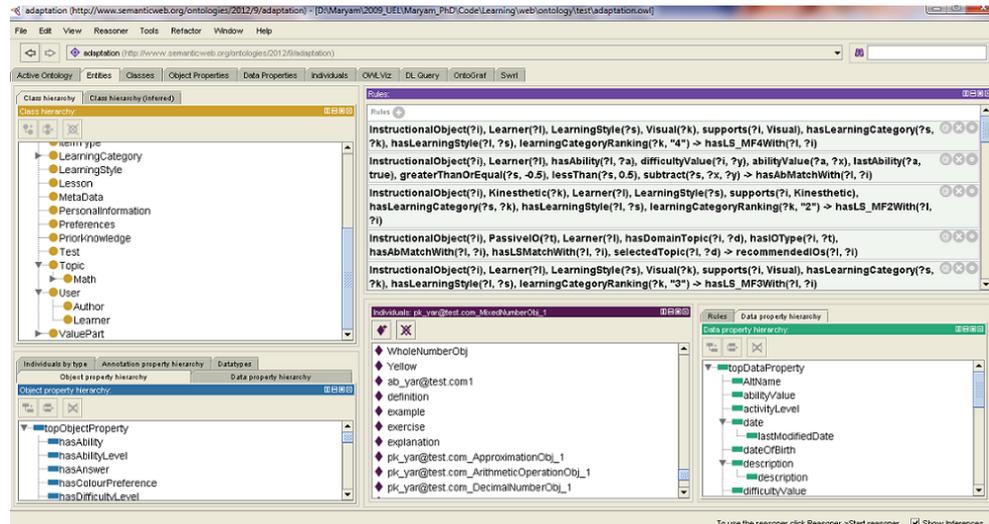


Figure 3.3: A screen shot of Protégé editor

In the next section the most popular web languages for representing ontologies will be reviewed. These languages are Resource Description Framework (RDF), RDF Schema (RDFS) (Antoniou & Harmelen, 2008) and Ontology Web Language (OWL) (Antoniou & Harmelen, 2004). They are based on the XML syntax (Antoniou & Harmelen, 2008), but have different terminologies and expressions.

Resource Description Framework/Schema (RDF/S)

RDF (mcbride, 2004) is a language recommended by the World Wide Web Consortium (W3C) to describe web resources and their relationships. In addition, it is a standard model for demonstrating metadata. Regardless of the representation syntax, RDF models use typical knowledge representation methods so they can offer higher-standard semantic interoperability and the ability to be understood by computers.

RDF uses Internationalised Resource Identifiers (IRIs) to identify resources. An IRI is a long string of characters which allows RDF to directly refer to non-local resources. The RDF is a very simple language, based upon the idea of making statements about resources in the form of subject-predicate-object expressions known as *triples*. The subject indicates the resource, and the predicate expresses a relationship between the subject and the object. The simplest way to represent a statement is to use the definition and turn it into a triple. For example, the statement: “The topic of content with id 138 is Add Fraction” can be presented as follows:

< *http : //www.uel.ac.uk/content_id138* >< *http : //www.uel.ac.uk/topic* >
“AddFraction”

Subject: The Subject is the resource we want to make a statement about. In our example we want to make a statement about the content with id 138. In order to express a statement about this content, the IRI “http://www.uel.ac.uk/content_id138” is used to identify this content.

Predicate: The predicate describes the kind of information expressed about the subject. The predicate is also called the property that describes the subject. In our example we want to make a statement about the topic of a content. The IRI “http://uel.ac.uk/topic” presents the topic property.

Object: The object defines the value of the predicate. In our example we want to state that the topic of the content is “Add Fraction”. The object can be a literal, like in our example, or another resource represented with an IRI.

Another way to present a statement in RDF is to use a labelled directed graph representing RDF’s underlying data structure. The graph of the preceding example is shown in Figure 3.4. The figure shows a single edge labelled with the predicate which connects two nodes each labelling subject and object. This kind of graph representation is called a semantic net in the Artificial Intelligence community.

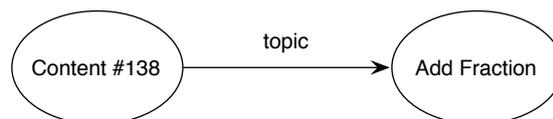


Figure 3.4: RDF graph of a single statement

The code for the preceding statement can also be represented in XML as follows:

```
<rdf:Description about="http://www.uel.ac.uk/content_id138">  
  <Topic>  
    Add Fraction  
  </Topic>  
</rdf:Description>
```

RDF is domain-independent, with no assumptions made about the particular domain of use. It is up to the users to define their own terminologies in a schema language called RDFS (RDF Schema). RDFS uses logic to describe concepts which RDF can use as a predefined structure. It is considered by some that RDF Schema has a similar relationship

to RDF as XML Schema to XML. However, this isn't necessarily true as XML Schema limits the structure of XML documents while RDF Schema defines the vocabulary used in RDF data models. In RDFS we can define the vocabulary, specify which properties apply to which kinds of objects, what values these objects can take and describe the relationships between them.

RDFS provides a modelling that concerns the organisation of vocabularies in terms of hierarchies: subclass and sub-property relationships, domain and range restrictions and instances of classes. It is also a good basic language for building many other languages (e.g. OWL) as it is not very expressive. However, it has limitations in describing resources including descriptions of existence, cardinality, localised range and domain constraints or transitive, inverse or symmetrical properties (detailed in the next section). The need for a new language arose to overcome these limitations which resulted in the development of the Ontology Web Language OWL.

Ontology Web Language (OWL)

To overcome the weaknesses of RDF/S several web ontology languages were proposed by the semantic web research community including SHOE, OIL and DAML+OIL. On the other hand, it was recognised that to develop the semantic web, a standard ontology language will be necessary. Therefore, the World Wide Web Consortium (W3C) working group developed a standard for a web ontology language which resulted in the OWL ontology language standard.

Ontology Web Language (OWL) (Bechhofer et al., 2004) is a language for representing ontologies based on OIL and DAML+OIL, and is integrated with RDF. The integration of OWL and RDF results in OWL being based on RDF's syntax, thus the web-based applications can directly access OWL ontologies. Similar to RDF Schema, OWL can declare classes and properties, organise them in a "subclass" and "subproperty" hierarchy and assign the domain and range of these properties. It can also express which individuals belong to which classes, and what the property values of specific individuals are. Nonetheless, it should be noted that OWL is an extension of RDFS in a higher logical layer. Therefore, it offers the following for expressing meaning and semantics:

- Defining equivalence or difference classes and properties, using properties like *equivalentClass*, *sameAs*, *disjointWith*.
- OWL classes can be specified as logical combinations using Boolean “or”, “and” and “not”, which in OWL is called *unionOf*, *intersectionOf* and *complementOf*.
- Declaring logical properties of properties, like *TransitiveProperty*, *SymmetricProperty* and *FunctionalProperty*.
- Defining inverse relationship between properties using *inverseOf*.
- OWL constructors class have more restrictive mechanisms on the kinds of values the property may take such as specific values, universal or existential quantification using *hasValue*, *allValuesFrom* or *someValuesFrom* respectively.
- OWL allows cardinality restriction using properties like *minCardinality*, *maxCardinality*.

OWL is based on Description Logic (DL) which enables for full formalisation of the meaning of the OWL language propositions. This formalisation provides formal semantics and automated reasoning techniques which allow for consistency checks on classes, individual instances and entailment relationships and places the classes in an organisational hierarchy and the individuals as instances of the classes (Horrocks et al., 2003).

Since developing high quality ontologies and employing them in an application needs complex reasoning, having OWL based on description logic allows for making use of reasoners already available for DL such as FaCT++ (Tsarkov & Horrocks, 2006), Racer (Haarslev & Möller, 2001), and Pellet (Sirin & Parsia, 2004) which are in fact shown to be very effective in reasoning. Moreover, ontology tools, such as SWOOP¹², Protégé, and TopBraid Composer¹³ use such reasoners to provide continuous feedback to the user about the logical implications of their ontological design (Horrocks, 2008).

OWL is also based on OIL which is a frame-based language. Frame paradigm makes ontologies easier to read and understand as it describes the information about each class with a frame. This feature is important for users not familiar with description logic. The frames paradigm has been used in a number of knowledge representation systems such as

¹²<http://code.google.com/p/swoop/>

¹³<http://www.topbraidcomposer.com/>

the Protégé.

OWL is based on both an RDF/XML syntax and an abstract frame syntax, and it has three different sub-languages namely OWL-Lite, OWL-DL and OWL Full. They have different expressiveness and computational complexity (Horrocks et al., 2007).

OWL-full uses all OWL language primitives. It is syntactically and semantically an extension of RDF and RDFS. Therefore, the expressivity of OWL-full is more than the other two sub-languages which leads to it being undecidable.

OWL-DL is a sub-language of OWL-full based on Description Logic (Horrocks et al., 2003). Description Logic is a decidable fragment of First Order Logic which supports formal semantics, thus OWL-DL have decidable inference. The formal reasoning facilitates the use of deduction to infer new knowledge from the information explicitly available in an ontology (Krötzsch et al., 2012). This means that OWL-DL automatically computes classification hierarchy and enables automated reasoning and inconsistency checks in ontologies. OWL-DL has less expressive power thus less computational complexity than OWL-full and is not fully compatible with RDF. Therefore, not every RDF document is a legal OWL-DL document while all OWL-DL documents are in fact an RDF document as well.

OWL-Lite is a syntactic subset of OWL-DL. It has simpler syntax and tractable inference, while its expressiveness is restricted.

In practice, ontologies are often developed using integrated, graphical, ontology-authoring tools such as Protégé (Noy & Musen, 2000), OilEd (Bechhofer et al., 2001) and OntoEdit (Sure et al., 2002). They are used to graphically develop ontologies and modify existing ones. These tools enable authors to edit and develop ontologies concentrating on the domain's concepts and relationships, without worrying much about ontology languages.

3.6.3 Using Ontology in Personalised E-learning Systems

The idea of representing knowledge with ontologies has become popular in many areas of Information Sciences, including personalised e-Learning (Sosnovsky, 2011). Ontologies facilitate formal representation of knowledge and formal reasoning to support explicit semantics and automated processing (Vesin et al., 2012). In an e-learning context, ontologies

can simplify the development of learning content for authors and instructors with improved personalisation, enable knowledge sharing and improve reusability (Sicilia et al., 2011).

The technology of ontologies can provide abstract description of learning contents by providing the terminology and thesaurus functionalities to annotate resources (Chi, 2009). Annotation is a form of attaching information (metadata) to an existing resource (Pahl & Holohan, 2009). The information in learning content document can be categorised into definitions or explanations about a new concept, examples, exercises, procedure and so on. Ontologies provide syntactically and semantically suitable annotations to present this classification explicitly. For example, the TANGRAM system (Jovanović et al., 2006) supports ontology-based, fine-grained annotation of learning content to make individual components of learning contents searchable and reusable. It automates this annotation process using concepts or terms extracted from a number of ontologies. In this system, the ontologies help to specify the structure of learning content formally and annotate individual components of learning content semantically.

Typically learners have very different levels of knowledge, needs and preferences, and learning contents should meet these requirements. Therefore, adapting content to individual learners before the content is presented to the learner is important. Using ontologies and reasoning tools in order to apply personalisation in e-learning systems seems a promising solution. Formal representation of ontologies can be taken as a foundation for this work (Dolog & Nejdl, 2007). To achieve this, several ontologies can be utilised to organise relevant knowledge into domain, learner and adaptation model. Domain ontology defines the concepts comprising a course and supports breaking it down into these concepts in a structured way. An ontological learner model captures the profile of a learner which offers the necessary information about the learner needed for personalisation of the learning process. Lastly, an adaptation model captures the knowledge represented in different ontologies and reasons over them to produce personalised learning paths and educationally sound learning contents in a domain.

Additionally, ontologies support sequencing the content in an educational system by categorising concepts into a hierarchy of classes and subclasses and representing variety of different relationships between them (Sicilia et al., 2011). Different types of relationships (e.g. *hasPrerequisite*) can be defined to arrange the content units in a suitable sequence.

Educational domains often involve relationships that express dependencies such as *hasPrerequisite*, *isPartOf* and *isBasedOn*, and any of these can be selected as the main driver of the sequence, by which concepts are placed in the sequence. The ontological reasoning then can determine the ordering of dependencies.

E-learning systems typically work on a fixed set of learning contents which are designed and fixed to the system in the design time. Therefore, adding new contents or modifying existing ones need a lot of modifications. Ontologies have been proven to be an effective means to overcome this problem by enabling the reuse of knowledge (Dolog & Nejdl, 2007; Jovanović et al., 2009; Henze et al., 2004; Pahl & Holohan, 2009).

In this section, we have seen that the ontology technology, from simple taxonomies to logical reasoning, can be beneficial in developing adaptive e-learning systems. However, ontology offers very basic reasoning features based on description logic. They can however be empowered with rules to express relations that cannot be represented in ontologies. In the next section we will discuss ontology languages extended with rules for enhanced expressiveness.

3.7 Extending OWL with Rules

OWL is an accepted language for representing ontological knowledge bases. It has a well-defined meaning which is used to describe domain concepts and their relationships. However, it has limited expressiveness, particularly in identifying semantic relationships between individuals which is a result of trying to retain the decidability of key inference problems (Horrocks et al., 2005). For instance in OWL, it is impossible to capture relationships between role chains as there is no composition constructor. The famous example which shows the limitation of OWL is its inability in defining the “uncle” property using the “parent” and “brother” properties. Therefore there is an effort on the way to address such expressive restrictions by extending OWL ontologies with rules.

Rules can extend the power of knowledge representation to recognise semantic relationships among individuals. They capture dynamic knowledge as a set of conditions that must be fulfilled in order to derive further information that cannot be achieved by ontology. In the context of the Semantic Web, the Semantic Web Rule Language (SWRL) extends OWL

with Horn-like rules based on the rule mark-up language RuleML (RuleML, 2012). It enables automatic deduction of new knowledge from existing facts. Description Logic and rules have different expressivity, fit with different types of knowledge and support specific reasoning services .

3.7.1 Semantic Web Rule Language (SWRL)

SWRL, the Semantic Web Rule Language (Bechhofer et al., 2004) (Horrocks et al., 2004) is the combination of OWL-DL and Rule Markup Language (RML). SWRL extends OWL-DL with rules while supporting OWL's existing semantic and syntax; semantically, SWRL rules have formal meaning through an extension of OWL-DL model-theoretic semantics; and syntactically, SWRL is based on OWL XML presentation syntax (Horrocks et al., 2005). SWRL adds a new set of axioms to OWL-DL, namely Horn-like rules, to expand the formal representation and semantics of OWL ontologies. It utilises a subset of first-order logic syntax, in order to reason over individuals in a knowledge base (Wang & Kim, 2006). Thus, SWRL rules ultimately increase the expressivity of OWL-DL. Variety of rule engines can be used to reason with SWRL rules as the SWRL Specification does not force restrictions on how reasoning should be performed (Vassileva & Bontchev, 2009).

As we mentioned in the previous sections, an OWL ontology comprises a sequence of annotations, axioms and facts where axioms have various types, such as *equivalentClass* or *disjointWith*. SWRL extends these by adding rule axioms to OWL ontology. Consequently, SWRL enriched ontologies include a combination of rules and other OWL constructs, i.e. ontology annotations, axioms about classes and properties, and facts about OWL individuals.

SWRL rules are of the form of an implication between an antecedent (body) and a consequent (head) in the following form:

antecedent \rightarrow consequent

Both the antecedent and consequent can include multiple atoms (written $a_1 \wedge a_2 \wedge \dots \wedge a_n$) or be empty. Atoms can be written in the following forms:

1. $C(x)$ where C is an OWL description and x is an OWL individual variable or a data value.

2. $P(x, y)$ where P is an OWL object property and x and y are OWL individual variables or data values.
3. $Q(x, y)$ where Q is an OWL data property and x and y are OWL individual variables or data values.
4. $B(x_1, x_2, \dots)$ where B is a built-in relation and x_1, x_2, \dots are OWL individual variables or data values.
5. $sameAs(x, y)$, $differentFrom(x, y)$ where x, y are OWL individual variables or data values.

Multiple atoms in an antecedent are connected via conjunction, while atoms in the consequent should be treated separately, i.e., they must all be satisfied independently of each other (Antoniou et al., 2005). The informal meaning of a rule can be described as: “whenever the conditions specified in the antecedent hold, then all conditions specified in the consequent must also hold”.

The following SWRL rule implies the *hasUncle* relationship from a combination of the *hasParent* and *hasBrother* relationships:

```
hasParent(?x,?y) ^ hasBrother (?y,?z) -> hasUncle(?x,?z)
```

In the above case, the antecedent is the conjunction of two atoms *hasParent* (?x,?y) and *hasBrother*(?y,?z) where variables are indicated by a question mark before them and *hasUncle*(?x,?z) is the consequent. From this rule, if Mikael has Mary as a parent and Mary has Bill as a brother then Mikael has Bill as an uncle:

```
hasParent(Mikael, Mary) ^ hasBrother(Mary, Bill) -> hasUncle(Mikael, Bill)
```

Having a language to define inference rules in knowledge models represented in OWL leads to several studies on using ontologies in conjunction with SWRL for defining adaptation processes in e-learning systems (Vesin et al., 2012; Chi, 2009; Sicilia et al., 2011; Vassileva & Bontchev, 2009; Popescu et al., 2007).

3.8 Survey of Personalised E-learning Systems

There have been many attempts to implement personalised e-learning systems. In recent years researchers have been mainly focusing on applying semantic web technologies to implement personalised e-learning systems. In this section we present and analyse many diverse approaches in implementing personalised e-learning systems in order to highlight the novelty of the approach suggested in this research.

3.8.1 ADAPT²

Advanced Distributed Architecture for Personalised e-learning (ADAPT) (Brusilovsky et al., 2005) is created to apply a higher level mechanism for ontology-based interoperability between self-contained adaptive web-based systems. It uses original protocols for interactions between various components. The main idea of ADAPT² is the use of an ontology server (OS) to user model exchange. The OS stores ontological structures of educational domains to resolve possible conflicts in domain models for specific applications. It acts as a common central storage of user's knowledge for different concepts reported by any *user model server*. User Model Server stores students' activities and infers their learning characteristics, which form the basis for personalisation. ADAPT's architecture allows for multiple OS - as several ontologies even for the same domain are possible.

3.8.2 Personalised E-Learning in the Semantic Web

Henze et. al. (Henze et al., 2004) proposed an ontology-based framework for personalised e-learning. The framework has several ontologies, each corresponding to a component of an adaptive hypermedia system: a domain ontology, a user ontology, an observation ontology and a presentation ontology.

This approach employs semantic web technologies like the Resource Description Format (RDF) (RDF, 2004) or RDF schema (RDFS) (RDF, 2004) to define vocabularies for metadata records and to describe resources. On top of the RDF and the ontology-layer, a layer of rules is used to enable reasoning over distributed information resources in order to dynamically derive hypertext relations. Rules are defined using TRIPLE (a rule language

for querying and transforming RDF models) (Sintek & Decker, 2002) to reason about RDF-annotated information resources.

3.8.3 Personalisation Services

Dolog et. al. (Dolog et al., 2004) proposed an approach to provide personalisation in the semantic web in the area of education and learning. This approach fills the gap between the current adaptive educational systems with personalisation functionality and open, dynamic learning repository networks. The approach is based on semantic distributed services and employs Semantic Web technologies to represent information about resources, learners and services, and extends a number of e-learning standards. It defines an architecture which integrates distributed learning repositories and services without the need for a centralised control. The overall architecture for personalisation services, presented in Figure 3.5, is distributed, and integrates system's components and distributed learning resources.

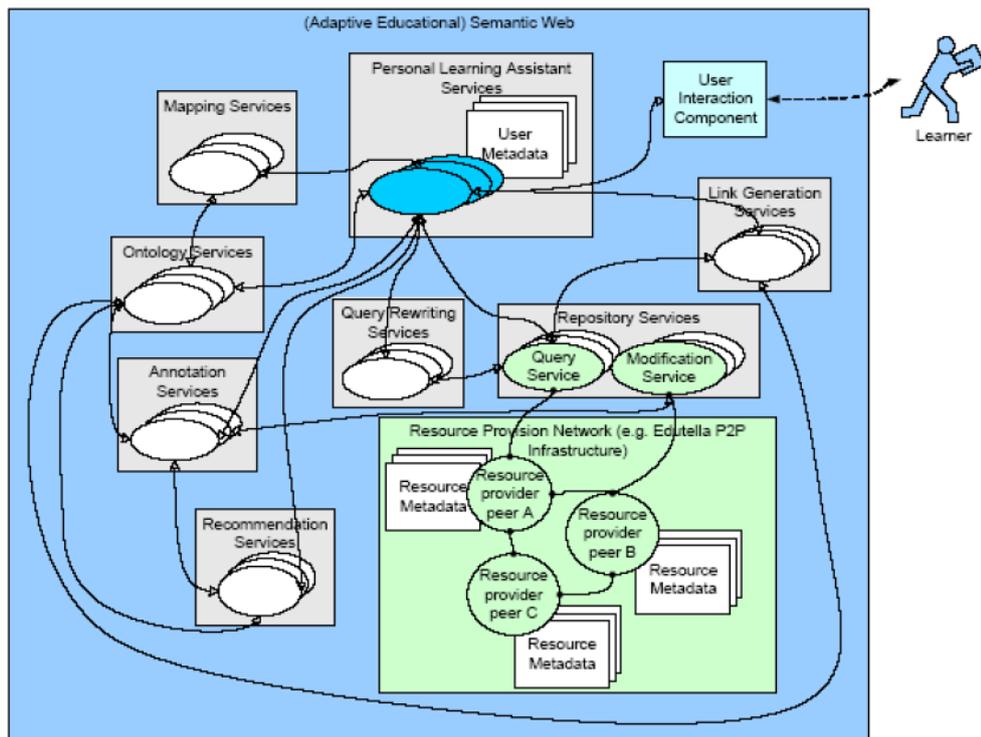


Figure 3.5: An architecture for personalisation services (Dolog et al., 2004)

The central component of the architecture is Personal Learning Assistant Service which integrates personalisation and other supported web services to find learning resources, courses, or complete learning paths suitable for learners. The architecture benefits from

different semantic web technologies. For instance, OWL is used to describe information and learning resources provided in various connected systems, DAML-S (DAML-based Web service ontology) (Burstein et al., 2002) is used to describe services which carry out personalisation functionalities and WSDL (web service description language)(Chinnici et al., 2007) and SOAP (Simple Object Access Protocol)(Mitra & Lafon, 2007) are used for accessing personalisation functionalities in the form of web services. This system integrates recommendations and link generation services to provide personalised access to learning resources.

3.8.4 DIOGENE

DIOGENE (Sanginetto, 2008) is a Learning Management System (LMS) and an adaptive e-learning platform which generates personalised courses by assembling materials using both static and statistical knowledge. Statistical knowledge is information about the learner during her interactions with the activities executed at the end of each learning session. It is the student's knowledge and preferred learning modalities which are continuously updated based on the learner's online feedback. Static knowledge includes information concerning available learning objects in a specific domain (e.g., "Euclidean Geometry," "Object Oriented Programming Languages," etc.) which is presented in a machine understandable form and also an ontological description about didactic relations (i.e. prerequisite, ordering and hierarchical relations) between the concepts of that specific domain. A semantic network is used to present ontologies in which a node represents a concept domain and oriented edges linking different nodes represent the relationships between ontologies. Each single learning object is described by an associated metadata which is represented by IMS Metadata Standard to allow knowledge sharing with different platforms.

The Diogene's student model includes two modules namely the cognitive state and the learning preferences. Cognitive state module describes the student's degree of knowledge about each domain concept of the ontology. The learning preferences module represents the student's learning styles and other student's characteristics which are used during the LO selection process. The learning content is adapted to a student based on the Felder and Silverman learning style(Felder & Silverman, 1988).

3.8.5 TANGRAM

TANGRAM (Jovanović et al., 2009) is a web-based intelligent learning environment for the domain of intelligent information system (IIS). TANGRAM adapts learning content to learner's current level of knowledge, her learning style and other personal preferences. It also facilitates quick access to particular types of content (e.g. example or definition) about the topic the student is interested in (e.g. RDF documents or semantic web which belong to the IIS domain). The approach used in TANGRAM is to use ontologies for the automatic decomposition of Learning Objects (LOs) into reusable smaller units and the dynamic reassembling of such units into personalised learning objects according to student's domain knowledge, preferences and learning styles. It is fully ontology-based and relies on the following ontologies:

- Domain ontology, used to define the topics covered in IIS domain and the semantic relationships between them. To implement this ontology, the W3C SKOS Core¹⁴ ontology is used to structure the IIS domain in a generalised hierarchy and to define semantic relationships between topics.
- Content structure ontology is defined to decompose Learning Objects into content units of various granularity levels. It also includes properties to represent content navigational and aggregation relationships (e.g. *follows* and *hasPart* relations) between content units.
- Content type ontology is defined to formalise the educational context of content units. It specifies the instructional/pedagogical roles of content units with varying granularity levels (e.g. *abstract*, *introduction* or *definition*)
- Learning path ontology specifies pedagogical relations (e.g. prerequisite) among domain concepts to define learning trajectories. It also defines the difficulty level of domain topics.
- Semantic user model ontology enables the formal representation of user information which is essential for TANGRAM's functionality (e.g. learner's learning style, performance and preferences).

¹⁴<http://www.w3.org/TR/skos-reference/>

The components of learning objects are annotated semi-automatically using the proposed ontologies, thus, the system makes all individual components of learning objects, with any type and level of granularity, searchable and reusable (Jovanović et al., 2006). We believe that a similar approach could be used to improve the effectiveness of the adaptivity of our approach by setting the granularity to a finer value.

3.8.6 Curriculum Content Sequencing System

Curriculum content sequencing systems (Chi, 2009) manage adaptive learning routes to help students achieve their learning goals. The system utilises a knowledge-intensive approach to model curriculum contents sequencing expertise into a knowledge base. This approach uses ontologies for representing abstract views of content sequencing and course materials and uses semantic rules for representing relationships between individuals. Practical curriculum sequences and course materials can be inserted in the knowledge base as factual knowledge. This approach is implemented for teaching mathematics in elementary schools using OWL (Bechhofer et al., 2004) and SWRL (Horrocks et al., 2004).

The applications framework for the curriculum content sequencing system contains two subsystems:

- **Knowledge maintenance:** This subsystem is used to create and update knowledge in order to store all the knowledge necessary for curriculum content sequencing. To achieve this goal, curriculum experts determine sequential relationships of a specific course and publishers upload their materials and annotate them to describe them.
- **Knowledge retrieval:** This subsystem gets the user's input about their competence, examines the knowledge base and creates a new learning route from the curriculum content sequencing system by removing the items which the user selected as her competences.

The authors claim that the system is reliable and has a durable knowledge base (Chi, 2009). It is durable as the experimental results demonstrate that the system creates an adaptive learning route for the needs of various users. It is reliable as the knowledge providers who have minimal technical support continually maintain the system rather than system developers.

3.8.7 Protus 2.0

PRogramming TUtoring System (Protus) (Vesin et al., 2012) is a tutoring system for teaching the essence of the Java programming language. The system recommends personalised learning content based on student's learning style (Vesin et al., 2011). Protus 2.0 is the new version of the original system which performs effective personalisation mostly based on the Semantic web technologies. The system relies on the Semantic web standards and technologies, specifically, ontology and adaptation rules for knowledge representation and inference engines for reasoning.

The architecture of the system is designed to improve the ontology utilisation in order to (Vesin et al., 2011):

- promote a clear separation of concerns about the components of the tutoring system;
- make explicit the communication between the components;
- indicate support for building the components;
- emphasise the achievement gained through using the Semantic Web in the development of tutoring systems;

In Protus, several ontologies corresponding to the components of a tutorial system are implemented to achieve easier knowledge sharing and reuse. The implemented ontologies are:

- domain ontology: to describe how the content has to be structured
- task ontology: to give roles to each object of domain knowledge and the relations between them
- learner model ontology: to store learner's preferences and knowledge about the domain concepts
- teaching strategy ontology: to select or compute a specific navigation sequences among the resources and
- interface ontology: to generate an interface view for the learners

Moreover, in Protus 2.0, SWRL rules, implemented for on-the-fly personalisation, are classified into two groups; namely learner modelling rules which identify the learner style

based on observed learning preferences and adaptation rules which are used for content adaptation based on learning style and/or learning preferences Vesin et al. (2012).

3.8.8 Analysis of Personalised E-learning Systems

This section illustrates the differences and similarities of the systems described in this chapter. It tries to show the differences of our approach with the surveyed approaches taken for the development of each system.

Rule-based Reasoning

The analysis of the surveyed systems illustrates that in the recent years, much effort has been made in applying the semantic web technologies to different aspects of e-learning. However, most of these efforts are focused in utilising ontologies for representing knowledge (e.g. user profile, learning content, etc.). For instance, ADAPT² defines *ontology servers* for storing the students' level of knowledge concerning each unit of content in a central storage in order to then exchange the user model between different e-learning systems. Similarly ontology is used in DIOGENSE to represent learning materials. Only a few of the surveyed systems refer to semantic rules in combination with ontologies to provide adaptation in e-learning systems (Henze et al., 2004; Vesin et al., 2012; Chi, 2009; Dolog et al., 2004). In an attempt to enable reasoning in personalised e-learning domains, Henze (Henze et al., 2004) proposed a framework in which reasoning is implemented using the rule based language TRIPLE. Although her framework is used in personalised service oriented systems (Dolog et al., 2004), it does not support reasoning over teaching strategies. Curriculum Content Sequencing System (Chi, 2009) uses ontology solely for the purpose of representing content sequencing and course materials in an abstract fashion. Furthermore, the system uses semantic rules to enable reasoning over specific instances of general concepts (i.e. *individuals*). Finally, Protus 2.0 (Vesin et al., 2012) relies on ontology and adaptation rules in order to enable the authors to define different components of the system's architecture and to further enable communication between them.

Through a review of existing personalised e-learning systems, we found that many adaptive e-learning systems have their adaptation techniques either entwined in the system's business

logic or in their content model. Having adaptation model which uses rule-based reasoning to produce personalised learning experiences, facilitates flexibility and extensibility in generating different adaptive effects and also encourages the reusability and modifiability of the adaptation rules. These features enable instructional designers to easily modify and expand the adaptation techniques of the courses they author, without having any impact on the system as a whole. Rules also provide explicit definition of all personalisation features declaratively and hide many of the functionalities behind them which help to simplify modification of the adaptation rules for instructional designers. Only Curriculum Content Sequencing and Protus2 utilise rule-based reasoning to make further inferences. However, they do not use rules for providing adaptive learning content based on different characteristics of learners or for recommending adaptive guidance based on the learners' progress.

Using Learning Content on Finer Granularity Level

Ontology-based modelling supports describing learning contents in a fine-grained level which can be dynamically assembled to generate personalised learning at runtime. The size of a piece of content has significant impact on its reusability. This is because the adaptive systems are capable of including a discrete and small piece of content within a new personalised learning system and combine it with other pieces. It means that, if learning contents are shaped in a fine-granularity level, it makes it easier to dynamically generate personalised learning content for different learners and therefore increases the reusability of existing learning contents. However, most of the surveyed e-learning systems do not explicitly consider the use of learning content at a fine granularity level to generate on-the-fly adaptive learning content. Among different surveyed systems (Henze et al., 2004; Vesin et al., 2012; Chi, 2009; Brusilovsky et al., 2005; Dolog et al., 2004) only TANGRAM decomposes Learning Objects into reusable fragments, and dynamically reassembles them into personalised learning content by using content structure ontology. TANGRAM's limitation is that it can only guess sequences of content units such as examples associated with a specific topic. This is due to lack of detailed knowledge about the components of individual LOs. TANGRAM lacks the ability to describe domain topics precisely, and fails to assemble learning content units in an adaptive personalised

order. Having been inspired by this work, we overcome this limitation by keeping precise information about finer learning contents. Another shortcoming of TANGRAM is that the performance of learners is updated in the user model based on the visited topics, which is not accurate. When a learner visits a topic recommended by the system, his performance level is assigned to the maximum value. While merely visiting a page is not a guarantee for learning its content. Assignment of a lower value is based on the assumption that the learner, due to the lack of necessary prerequisite knowledge, was not able to fully understand the presented content. In our approach, learner's performance is updated in the user model based on the result of tests taken during the learning process (Yarandi et al., 2012a). Also, the ability of the learner is estimated according to the Item Response Theory (Baker, 2001) in order to select the most suitable learning content for each individual learner.

In TANGRAM, the adaptation techniques are not separated from the implemented code. It means that the adaptation model in TANGRAM is not explicit and is rather embedded in the content or the implemented code. Therefore, modifying the adaptation techniques is difficult and would require re-authoring the content model and redeveloping the implemented code. We try to overcome this limitation by designing a discrete rule-based adaptation model which simplifies the modification of adaptation techniques without impacting the implemented business logic.

Flexibility, Extensibility and Independence

The next generation of e-learning systems needs to provide greater flexibility, extensibility and independence to support today's learning requirements. However, most of the current e-learning systems have performed poorly in these areas. Ontologies can be associated with reasoning mechanisms and rules to support a given adaptation strategy in educational systems in order to increase flexibility, extensibility and independence of these systems through conceptual separation of adaptation rules from the system's core functionalities. Although, the surveyed systems presented in this section use ontologies, some of them do not apply rule based reasoning. For example, DIOGENE and TANGRAM utilise different ontologies to model the domain knowledge and user profile, and these remain separated from the core engine which interprets them. The Adaptation process, however, tends to

be intertwined with the core of the systems. This seriously influences the flexibility of these systems, as the adaptation is dispersed among different system components and therefore cannot be easily updated. Additionally, in order to produce different adaptive learning experiences, one has to manipulate the implementation of the system engine which influences the extensibility and independence of the systems.

Ontology can facilitate the representation of learning contents in various granularity levels, which have positive effects on the flexibility of systems offering more effective personalisation. In most of the surveyed systems (Henze et al., 2004; Vesin et al., 2012; Chi, 2009; Dolog et al., 2004) using this method is limited. Therefore, they cannot generate personalised learning paths dynamically based on learner's interactions with the system. To overcome this limitation, in our approach, ontologies are used to define learning contents at a lower granularity level (e.g. examples, explanations, exercises). In other words, smaller pieces of learning content are annotated through well-designed ontologies. Moreover, rule-based reasoning enables on-the-fly assembly of annotated fragments into structured learning content personalised to the learner's progress. Our approach increases the flexibility of the system to offer advanced levels of personalised learning out of existing small pieces of learning content.

3.9 Summary

This chapter reviewed and analysed the current state of the art ontology-based adaptive e-learning systems. As adaptive e-learning is the research area that we focus on, this chapter covered the methods of adaptivity and also discussed how they may be utilised in e-learning systems. Furthermore, the semantic web technologies provide standards, languages and formalisms which we use as the basis of our approach; thus this chapter described these technologies and illustrated how these technologies, in particular ontologies, can be used in modelling the component of personalised e-learning systems. This has followed with an overview of using rules to empower ontologies for representing knowledge. Rule-enriched ontologies allow us to model the components of adaptivity in order to improve their reusability as well as to enhance the flexibility and extensibility of e-learning systems. Finally, we presented a survey of the current personalised e-learning

systems and highlighted the differences between these systems, with special emphasis on their strengths and weaknesses from different perspectives.

Chapter 4

Specification and Design

4.1 Introduction

In this chapter we will discuss the issues involved in designing an adaptive system towards fulfilling the objectives and goals of this thesis. After this introduction, Section 4.2 explains the methodology of this research. In this section, details on research approaches and methods for collecting data are highlighted. Then Section 4.3 introduces and justifies the semantic rule-based approach for producing adaptive systems. In this section, we will explain the main features of this approach including the separation of the adaptivity models each representing different components of adaptivity, the representation of these models using ontology enriched with semantic rules, the abstraction mechanisms employed through ontological modelling to facilitate this separation, defining content in finer granularity levels and we finish this section by showing the architecture of our system.

Following the introduction of our approach in Section 4.4, we describe the design of semantic rules utilised for describing adaptivity and how these rules provide a flexible way of generating personalised learning paths at runtime based on the interaction of a learner with the system.

We then explain the ontology-based design of core models for enabling our approach including the Content, Domain, Learner and Assessment Models. In Section 4.5 we will cover the designing of content models and its related issues such as content's granularity level. In Section 4.6, we discuss how to model the different characteristics of the learners,

which have influence on the adaptive process. Section 4.7 discusses how generic domain modelling enables domain independence in the semantic rule-based approach. This section also describes how pedagogical relations between domain concepts are defined for enabling adaptive navigation. After that, in Section 4.8 we discuss the importance of assessments in adaptive systems and describe the test calibration process based on item response theory. Finally, Section 4.9 concludes the chapter with a discussion of the technologies that may be used to support the semantic rule-based approach.

4.2 Research Methodology

The purpose of this thesis is to propose a novel approach for supporting adaptive personalised e-learning systems. This novel approach aims to improve flexibility, extensibility and reusability of such systems while offering a pedagogically effective and satisfactory learning experience for learners. This study is conducted in three phases:

The first phase was performing the library research which refers to the secondary data and its analysis. The goal of this phase was to review and analyse a number of diverse approaches in implementing personalised e-learning systems in order to present the differences between these systems and highlight the shortcomings of the existing models.

In the second phase, an innovative semantic rule-based approach is proposed to overcome the deficiencies of the current approaches in designing and implementing personalised adaptive e-learning systems. In order to achieve the aim of this research, which is mentioned above, the approach proposes an ontological architecture featuring an independent adaptive engine. This engine does not include any knowledge about a particular domain or any adaptation strategy; it obtains all the necessary information from the respective ontologies. The approach also presents adaptation techniques with semantic rules for expressing adaptation strategies independently of the application's core engine. Finally, a semantic rule-based approach is designed and the Rule-PAdel which is an e-learning system implementing this approach is developed.

In the last phase, the proposed approach will be evaluated from to aspects:

- **Effectiveness of Rule-PAdel:** Examining the adaptive learning paths generated by

Rule-PAdel are pedagogically effective and satisfactory for learners and teachers.

- **Flexibility, extensibility and reusability of Rule-PAdel:** Examining the flexibility, extensibility and reusability of Rule-PAdel which are the result of modelling the components of adaptivity using ontologies enriched with semantic rules.

In order to estimate the effectiveness of the proposed approach, two different versions of the system are designed. The first version is a system which adaptively supports the learner through the learning process. The other is a replica of the adaptive system without any adaptive features. An experimental design is employed to assess the effectiveness of Rule-PAdel. During the experiment, each learner experiences with both the adaptive and non-adaptive systems. The learners explored the adaptive system for one set of topics and the non-adaptive system for the other set. Each session contained a 10-minutes pre-test, a 30-minutes interaction with the system and a 10-minutes post-test. Based on the results of the pre- and post-tests, knowledge gained values are calculated to determine the effectiveness of the adaptive learning techniques. Finally, at the end of the experiment each learner fill-out a questionnaire to reflect their perceptions about different aspects of the two systems.

4.2.1 Research Approach

There are two general approaches to reasoning which may result in the acquisition of new knowledge, deductive and inductive approaches. In deduction, one starts from the more general to get to the more specific. The important steps of the deductive approach are development of a theory or hypothesis, observation through data and information and confirmation respectively.

On the other hand, in induction one moves from specific observations to broader theories, thus observation, pattern, tentative hypothesis and theory are important steps of this approach. Based on this definition, we follow the **deductive** approach in our research. It is because we propose a theory in the second phase our research and collect data in the third phase in order to observe and confirm our hypothesis. The hypothesis of our research is as follows:

- It is possible to model the components of personalised learning independent from

the system's core business logic using semantic ontologies enriched with inference rules.

- The adaptive learning paths generated by the proposed approach are pedagogically effective and satisfactory for the students.
- The adaptive system implementing this approach is flexible and extensible in using different adaptation strategies, instructional plans and domain knowledge.

In the third phase of this research, we collect and analyse the data obtained through experiments to accept or reject these hypotheses.

4.2.2 Methods of Data Collection

Overall, research methods are divided into two basic categories: quantitative and qualitative. Quantitative research gathers data in numerical form which can then be categorised or ranked in order while qualitative researches involve gathering information not in any numerical form.

This research uses a mixed design method; we employ both quantitative and qualitative methods for our experiment. In the third phase of the research, information is collected in the following three ways in order to accept or reject the hypothesis made in the beginning of the research:

1. **The score of learners in the pre and post-tests:** This data is quantitative. We collect and compare learners scores in pre and post-test to calculate and compare the effectiveness of the adaptive and non-adaptive systems on learners' progress.
2. **The result of the questionnaire:** The questionnaire comprises of twenty questions and an open comment section. In the questionnaire learners were asked to express their opinion about their experience with the system, based on the five Likert-type scale (Likert, 1932) ranging from strongly agree to "strongly disagree". The data determined through different questions is quantitative as it scales from 1 to 5 and the data determined through open comments section is qualitative as it is not measured in a numerical form.
3. **Interview with teachers, course authors and instructional designers:** In this part,

we collect information to evaluate the satisfaction of teachers on the effectiveness of the system and also that of course authors and instructional designers about the flexibility, extensibility and reusability of the system. This information is qualitative as it is collected through interview which is not in a numerical format.

4.3 Justifying Semantic Rule-based Approach For Supporting Personalised Adaptive E-Learning

The study of the state of the art discussed in Chapters 2 and 3 impacts the design of the semantic rule-based approach. In order to justify our approach in Section 3.8, we describes many diverse approaches to implementing personalisation in e-learning systems, but these approaches are commonly limited in enabling flexible personalisation at run time. They also neglect the enabling of flexibility for the system in using different adaptation techniques as well as using fine-grained learning content. In other words, these systems have to be used on their own. It is not possible to replace a personalisation component of one system with another, maybe an updated one, without having to reconfigure the whole system, or worst, changing the system's low level implementation code. Embedding personalised components, especially adaptive strategies, into adaptive engines restricts the ability of modifying these components. Moreover, the definition of coarse-grained content restricts the flexibility of the system in producing effective personalisation. Such embedded logic also restricts the instructional designers in modifying adaptation strategies as they should have a tighter collaboration with developers in implementing the new adaptation strategies. However, by clearly separating the adaptation strategies and adaptive engine a variety of different techniques may be applied without having impact on the adaptive engine. This separation also enables the authors of personalised e-learning to simply update and expand the adaptive strategies. Through utilising learning content in finer granularity level the system can benefit from the best piece of content for dominating content composition.

Considering the limitations of traditional approaches in designing personalised e-learning systems, the semantic rule-based approach is designed to meet the following properties:

- To generate a satisfactory and pedagogically suitable learning path.

- To be flexible and extensible in supporting different pedagogical needs by requiring course authors and instructional designers to modify only the adaptation techniques, instructional plans or concept sequencing when there is a need to make a change to any of the mentioned techniques without the need to change the implementation section of the system.
- To have reusable adaptation components.
- To adapt learning contents to the learners accurately based on their current knowledge and abilities.
- Recommend adaptive guidance for learners at runtime based on their interaction with the system.

In Order to achieve these properties, the following technical requirements should be addressed in the design of the system:

- Independent adaptive engine with no knowledge about a particular domain or specific teaching strategy.
- Different components of adaptivity designed separately.
- The use of semantic rules in order to express adaptation strategies independently of the application's core engine.
- The use of Item Response Theory for the calculation of learners' abilities.

Recall that in the analysis of the learning theory discussed in Chapter 2, constructivism has a great influence in education. Constructivism gives more responsibility to the learner during the learning process. In many early Intelligent Tutoring Systems, the learners were not satisfied with the decisions made by the system. Constructivism can remediate this problem by giving the learners appropriate controls over the learning process. By meeting this criterion the learner should be more satisfied with the e-learning experience.

Representing each component of adaptivity by a specific ontology facilitates a clear separation of the components which result in promoting the reusability of these components. In the adaptive e-learning Systems surveyed, many approaches for the implementation of adaptivity are described, but these approaches have one common feature that results in limited reusability of adaptation components. They either partially support the reusability

of the components of adaptivity or they do not support it at all. For example, In TANGOW all media elements appear in a Course Content database, and all the teaching tasks are defined in a Teaching Tasks Repository. The use of database allows for reuse of learning content in different courses (Carro et al., 2002). However, there is no separated domain and content model to present the structure of subject domain and reference to actual learning content separately. Additionally, adaptive strategy is often entwined with the system's business logic which results in no support for reusability.

Another desired property for the system is to be flexible and extensible in supporting different pedagogical needs. This property is achieved by enabling course authors and instructional designer to easily modify the adaptation techniques, instructional plans and concept sequencing separately without the need to change the implementation code. The Adaptive e-learning systems surveyed in Section 3.8 (e.g. TANGOW, DIOGENE and TANGRAM) do not have independent adaptive engines. In those systems, adaptation strategies and instructional plans are embedded into adaptive engine. For example, TANGOW adapts learning content to learners but the adaptation rules are embedded in the business logic. Thus, the modification of these rules is not easy as the developer of the system should change the implementation code. Therefore, this system does not offer enough flexibility and extensibility.

Many authors have suggested that providing adaptation towards a learner's learning style is important to improve the effectiveness of learning process (Fleming & Mills, 1992). However, there is no agreement on how best learning styles should be modelled. For example, DIOGENE and TANGRAM provide adaptation for learners based on Felder and Silverman learning style and preferences. However, a suitably flexible mechanism should be chosen to enable instructional designers to facilitate the implementation of several different learning styles and preferences. This has significance for providing different adaptive effects which is facilitated in our approach by having an independent adaptive engine and representing the adaptation strategies by semantic rules.

This research proposes a semantic rule-based approach for supporting dynamic personalised learning with sufficient flexibility to support different adaptation effects and domains of knowledge.

In order to overcome the limitations of the surveyed approaches, ontology is used to repre-

sent adaptation components explicitly. It allows data to be read, processed and understood by machines precisely and intelligently. Ontology allows for defining the concepts of a domain and their relationships formally and explicitly. They have also the potential to clarify domain's structure of knowledge and to enable reasoning about knowledge domains. Ontology can be used to represent abstract views of concept sequencing and learning content. Therefore, representing each component of adaptivity by a specific ontology makes possible a clear separation of these components. This separation promotes the flexibility and extensibility of adaptive e-learning systems and also the reusability of their adaptation components. Integration of these discrete adaptation components at runtime support flexible personalisation during the learning process.

Additionally, representing adaptation techniques with semantic rules facilitate flexibility and extensibility of the system in implementing different adaptive effects and also improve the reusability and modifiability of the adaptation rules. These features enable instructional designers to easily modify and expand the adaptation techniques of the courses without having any impact on the system as a whole. Rules also provide transparency for many of the functionalities behind them and also provide sufficient flexibility resulting in a simplified modification of the adaptation rules by instructional designers.

This section describes the design of the semantic rule-based approach in a way that remediates the limitations of the current surveyed approaches by achieving both the properties and technical requirements mentioned above.

4.3.1 Model Separation and Model Types

The key components of personalisation are designed into separate models in the semantic rule-based approach. The content model describes the learning content, includes references to the actual content and does not include any adaptation techniques or content sequencing. The domain model represents the structure of subject domain through defining different topics and it also represents the semantic relation between these topics. The adaptation model contains a rule set to match the learning content with learners and infer adaptation decisions.

The adaptation model uses an abstraction mechanism for representing content and does

not include pieces of content directly. Additionally, the important features of learner which are needed for personalisation processes is separately described in the learner model.

The adaptation model does not refer to pieces of content or content models directly; instead it uses an abstraction mechanism. The learner model describes the important features of learner which are needed for personalisation processes. The separation of content specification, domain structure and adaptation strategies in time open the way to adaptivity. When content specification is defined by the course author, the system will access to the best piece of content for learner available at the time of using the system.

4.3.2 Ontology and Rule for Model Representation

Recent developments in semantic web technologies have stimulated a trend in using ontology to promote adaptive learning. Semantic web allows data to be read, processed and understood by machines precisely and intelligently. Ontology, an explicit formal specification of a conceptualisation (Gruber, 1993), is the most suitable means for representing knowledge due to its flexibility and extensibility in designing concepts and their relationships. This definition emphasises that ontology allows defining formally and explicitly the concepts in a domain and their relationships. They also have potential to clarify the domain's structure of knowledge and to enable reasoning about knowledge domains (Chandrasekaran et al., 1999). Therefore, they have proven to be useful for representing knowledge in many domains particularly in the educational environment.

Although ontologies have a set of basic reasoning mechanisms, they have expressive limitations in representing knowledge such as composition constructor. This means that it is impossible to capture relationships between one composite property and another. The standard example is the relationship between the composition of the parent and brother properties and the uncle property (Horrocks et al., 2005). To overcome some of these expressive restrictions, ontologies are extended with some rules to make further reasoning and to express relations that cannot be presented by ontological inference. The semantic rules are used to express adaptation strategies and they can also enable on-the-fly personalisation based on the interaction of learner with the system. Moreover, semantic rules promote the maintainability of the adaptation process as they separate the adaptation logic from programming code. They also allow authors (non-professional programmers)

to develop adaptation techniques.

Well-designed ontologies empowered by rules are of key importance in the semantic rule-based approach. The ontologies are used as the basis for generating a personalised learning path based on learner's needs. The defined ontology should meet the following specification:

- **Flexible** to enable the authors to design different components of adaptivity that influence the adaptive process.
- **Extensible** to allow the authors to insert more elements of adaptivity.
- Semantically rich capable of logically expressing adaptation decisions
- **Independent** to facilitate explicit definition of all components of adaptivity without impacting on the system implementation code
- **Maintainable** to provide sufficient modifiability for personalisation options

To this end several ontology and rules were designed that would enable annotating knowledge domain to achieve these goals. Accordingly, the starting point in our approach was the classification of ontologies in the domain of e-learning which differentiates the following types of ontologies:

- Content ontology that formalises the conceptual structure of content and define features and the relationships associated with each small piece of content.
- Learner ontology is defined to describe learners and their profiles.
- Domain ontology is defined to formally describe the hierarchy of domain topics and semantic relationships between them. This ontology enables a system to be domain independent and a fully ontology-based system.
- Assessment ontology is developed to formally represent relevant information about different assessments, especially the parameters which are used to calculate learners' abilities based on the IRT.

Finally, ontologies and rules interoperate semantically and inferentially to enable reasoning in order to dynamically derive adaptation strategies. These enable the instructional designer to understand, inspect and modify adaptive strategies.

The representation of each element of adaptation by a specific ontology makes a clear separation of the elements and explicit relationships among them. Additionally, presenting adaptation activities explicitly in the form of rule makes the system more manageable by simplifying the process of maintaining the adaptation rules as the instructional designer can easily update and expand them without impacting the software engine implementation. For example, content authors may add new types of content to support different learning styles. Therefore, instructional designers can modify adaptation rules to infer knowledge which enables system to recommend appropriate contents to learners.

4.3.3 Abstraction through Ontological Modelling

The semantic rule-based approach includes an abstraction mechanism which provides flexibility in designing and implementing new adaptive courses. Through this mechanism many people can collaborate to develop personalised courses. For example, when the course expert wants to develop the structure of a course which describes the course sequencing, she needs to define the learning topics and the semantic relations between them needed to realise the course, not in terms of the required learning content. On the other hand, instructional designers define adaptation rules, which can provide pedagogically sound adaptive learning. Similarly, the content author can develop pieces of learning content without concern about the place and the way in which content will be used. This abstraction offers greater independence for the instructional designer and course expert to design the course without concern about pieces of content. Also individual pieces of content can be modified or added without requiring to modify the instructional plan or domain topics.

This abstraction mechanism is facilitated through ontological modelling as ontology is capable of modelling concepts and relationships to a high level of abstraction. For example, several pieces of content may be defined about a particular learning topic; each covers the learning topic in different ways. These differences may be pedagogical or technical. For example, three pieces of content may be defined about adding fraction; one of them is a textual description, another consists of an example and the third one is a Flash animation. The decision as to which content to deliver can be made at runtime based on some information about the learner.

4.3.4 Instructional Objects using content on finer granularity level

Learner satisfaction and educationally sound personalisation are the main goals of the semantic rule-based approach. The size of the pieces of learning content, combined to form the content, is a key consideration to improve the flexibility of the system for offering advanced level of personalised learning. There is an inverse relationship between the size of designed content pieces and the flexibility of system. As the size of these pieces decreases, the flexibility of system to generate personalised content out of existing pieces also increases

For example, if a learning content is comprised of only a few pieces of coarse-grained learning then re-sequencing the form to a new content that supports a different instructional approach or adaptation strategy may not be possible.

“Learning content is usually referred to as Learning Object (LO) and may be defined as a unit of educational content that is delivered via the internet” (Conlan, 2006). A very broad definition of learning object is any entity, digital or non-digital that may be used for learning, education or training (Duval, 2002). This definition allows for an extremely wide variety of granularities (Schluep et al., 2006).

In our proposed approach to improve the flexibility of system, learning objects are defined in different granularity levels. The smallest grain of learning object (e.g. an example, definition and explanation) is called Instructional Object (IO). IOs are elementary fragments of learning content with different instructional roles and can be used to define a concept, explain a process, show an example or present an exercise. IOs as a fundamental building block facilitate generating different learning contents at run-time and shaping up these contents in a different way according to following criterion (see Figure 4.1) (Yarandi et al., 2011b):

- **Instructional plan:** this approach enables an e-learning system to produce several versions of the content each covering the same learning concepts but sequenced in a different manner based on diverse instructional designs. For example, one version of the content may start with some example and derive the definition of the concept. While another may start with the definition, present some examples and finish with an exercise.

- **Learner’s progress:** the system needs to identify how much knowledge learners acquire during interaction with the system. Consequently, different supplementary learning content can be generated to be recommended to particular learners in order to remediate the learning problem which is diagnosed according to the result of analysing previous learning activities. For example, one supplementary content may present some examples and exercises, while alternative content may present explanation of content in more detail.
- **Learner’s characteristics:** Different IOs are assembled in such a way that creates equivalent learning content to satisfy specific needs of individual learners. For example, several IOs may be defined about a particular learning topic, each cover the learning topic in different ways. This difference may be based on VARK learning style. Accordingly, the learners who prefer read/write mode may present a textual description, while visual learners may use visual text.

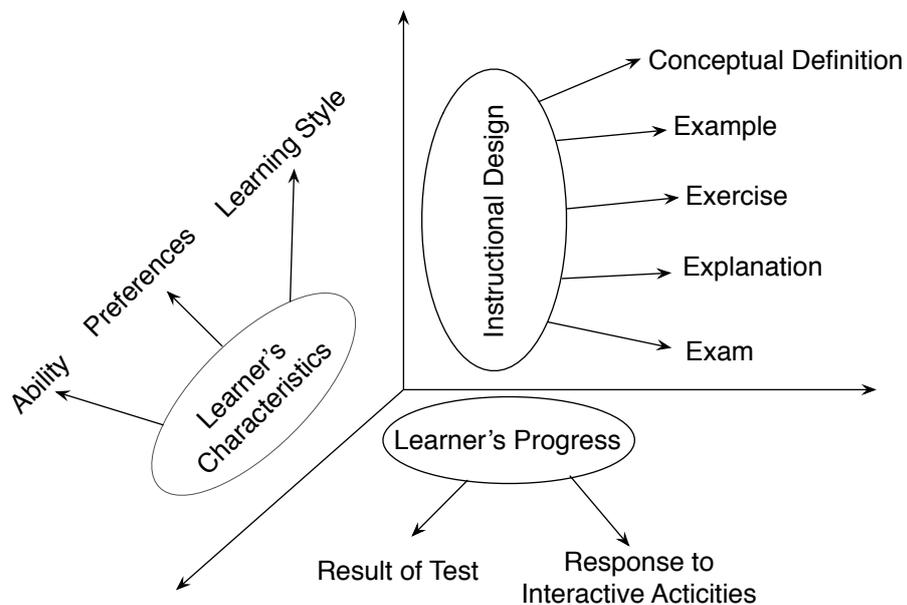


Figure 4.1: Three dimensions of adaptivity

The effective content of these different versions assumes that IOs are sufficiently fine-grained that they may be re-sequenced. Ontology facilitates the representation and re-sequencing of learning contents in various granularity levels. This semantic foundation increases the flexibility of systems offering more effective personalisation. In the proposed approach, IOs are annotated through well-designed ontologies.

4.3.5 The System Architecture

Figure 4.2 shows the system architecture for our semantic rule-based approach containing all the individual components of the system. It consists of a central Adaptive Engine along with four models to access the information about learners, learning contents, assessments and adaptation respectively, as well as a graphical user interface in order to facilitate the communication to the learner in a friendly manner. It also contains a courseware manager to allow course authors to manipulate the learning content, assessment and adaptation strategies.

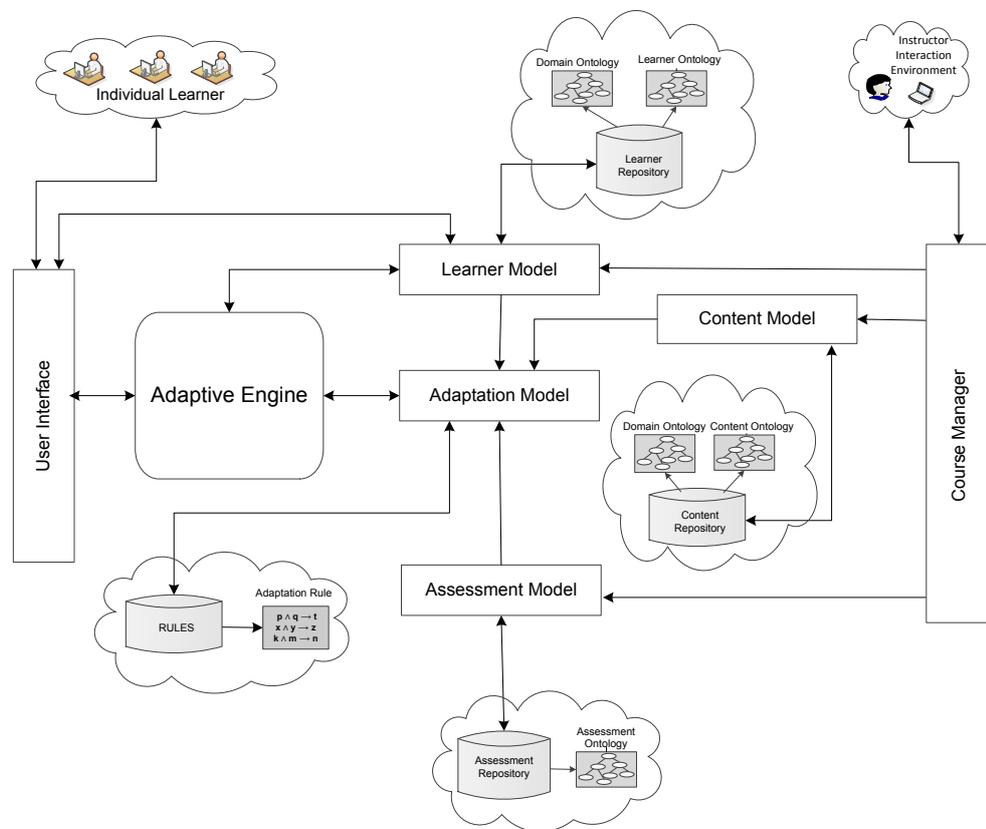


Figure 4.2: The architecture of system

In what follows, the functionalities of the above mentioned components of Figure 4.2 are described:

- **User Interface** is the communication component that controls the interaction between learners and the system. It deals with the learner's account system (including registration and login) and facilitates the learner's interaction with the learning components. It also captures the learner's responses in interactive activities and transfers

them to the Adaptive Engine.

- **Adaptive Engine (AE)** is at the heart of the architecture and is responsible for generating and recommending adaptive learning paths according to instructional plans, learner's progress and learner's characteristics (Yarandi et al., 2013). The detail of the AE subcomponents is explained in next section.
- **Learner Model** contains information about the learners. The system uses this information in order to adapt to learner's individual needs. The system gradually updates the learner model during the learning process, in order to keep track of learner's actions and progress and possibly guide the learner accordingly. Learner model is responsible for retrieving the characteristics of a particular learner, making the necessary changes and sending it to the adaptation model through interaction with the repository. The system also receives the knowledge about new learners from the User Interface and stores it in the learner model. Learner model is updated when it receives new information about the learner from the adaptive engine.
- **Content Model** presents storage for all essential learning content and also describes how the information content is structured. It is responsible for finding the learning objects stored in the repository, which meet some given criteria.
- **Assessment Model** contains all crucial tests specifications to accurately evaluate the learner's level of knowledge. It also searches the assessment repository to find appropriate assessments required by the adaptation model.
- **Adaptation Model** contains rules to support the adaptive functionality of the system. Different conditions are modelled in the body of the rules. These conditions are all obtained from different models such as Learner, Content and Assessment models
- **Courseware Manager** handles requests from the authors and instructional designers for inserting, updating and modifying the structure of course, instances of IOs and adaptation rules. It also allows the test developer to add new assessments and update them through the assessment model.

This architecture is designed in a way to best use ontologies for representing knowledge.

The use of ontologies in representing information:

- Promotes a clear separation of the components of personalisation.

- Improves the flexibility of the system through enabling authors to update different components of personalisation.
- Improve the extensibility of the system by allowing authors to develop new courses without touching any computer code.
- Promotes all the benefits of Semantic Web in the development of a personalised e-learning system.
- Increases the interoperability and reusability of different components of the system.

In the rest of this section, the structure of the Adaptive Engine will be described.

The Adaptive Engine

As said earlier, the Adaptive Engine is like the heart of the system. It is the part responsible for generating personalised learning experiences. Figure 4.3 shows the internal architecture of this key component. Although it does the necessary computations for personalisation, it does not contain the strategies or knowledge for any particular learning domain, concepts sequencing or instructional plan (Yarandi et al., 2013). The knowledge about the domain and the learners are all stored in their respective ontologies. This separation results in the architecture of the system to be highly modular and to have a high level of abstraction.

The AE consists of six major components. These components are described below:

- **Course Structure Constructor (CSC)** creates the annotated course structure by using link annotation and link dimming to offer adaptive navigational support techniques (Brusilovsky, 2007) which helps the learner in navigating the domain space with a lower cognitive load. Links to topics with different educational status are marked differently. CSC makes adaptive decisions about learner's knowledge from adaptation model and about the structure of the course from domain model, to construct the proposed annotated course structures (Yarandi et al., 2012b). Adaptation model gets learner's level of knowledge from learner model and prerequisite relation between different topics from domain model to make related decisions.
- **Guide** suggests the optimised learning path to an individual learner according to the learner's characteristics and responses to interactive IOs. So, it gets the decisions

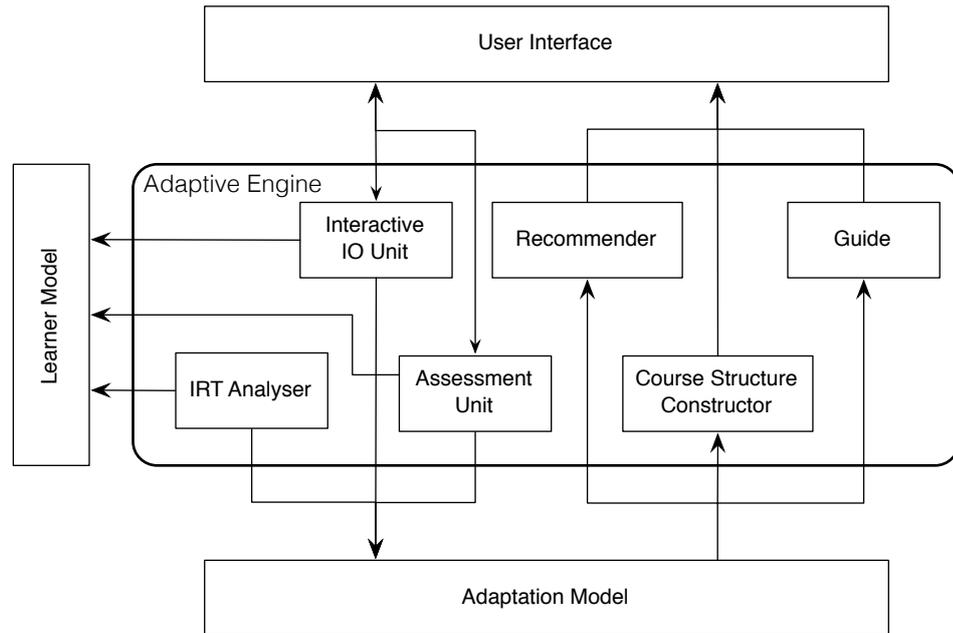


Figure 4.3: The adaptive engine

made in adaptation model which are based on the difficulty level of the interactive IOs, the result of analysing learner's response to them, difficulty level of content and defined activities in instructional design. The recommended learning path is presented to the learner through the user interface. For instance, if a learner is given a learning content with moderate or high difficulty level and she fails even to answer the related easy exercises, the recommender suggests her to repeat learning this topic with a lower difficulty level (Yarandi et al., 2012b).

- Recommender** assembles appropriate IOs to provide personalised learning content dynamically. It knows how to combine the suggested IOs to form a coherent learning content that best suits a particular learner. In more detail, adaptation model selects the IOs which are suitable for the learner. Recommender gets these and also the instructional design from the content model. It deletes the IOs which are presented to the learner in previous learning steps to prevent presenting them again, then organises the remaining IOs according to activities defined in instructional design. Finally, the generated personalised content is delivered to the learner through the User Interface.

- **Interactive IO Unit** receives the learner's responses to interactive IOs in order to then analyse them and provide suitable feedback to learner. This unit updates the learner model based on the information earned during the analysis of learner's responses.
- **Assessment Unit** receives the learner's responses to the presented assessments and analyses them. Then, according to the result of this analysis, the appropriate feedback is provided to the learner and the learner model is updated accordingly.
- **IRT Analyser** receives the learner's responses to tests from the assessment unit. According to IRT, learner's data is analysed to obtain the new learner's ability under 3PL model which is described in Section 2.4. Also learner model is updated according to this new ability.

4.4 Semantic Rules for Describing Adaptivity

The Adaptation Model contains a rule set that allows for rule-based reasoning in order to produce a personalised learning content and recommend adaptive guidance tailored to the learner's progress. Several conditions are held in the body of the rules. As a consequence of executing the rules, the concept of adapted content, adapted navigation and adapted guide are generated for individual learners. For instance, as a consequence of executing a number of rules, a decision can be made on whether a learner is able (has sufficient ability) to understand a particular learning content. Therefore one could say that the adaptation model joins the other models to generate the personalised offering. In e-learning systems the adaptation model utilises different instructional strategy and content recommendation to offer personalised learning. For example, an author may create an adaptation model that contains some rules about the learner's prior knowledge, while another one may create adaptation model with respect to learning style preferences.

Rules provide procedural knowledge to reduce the limitations of ontology inferences and express semantic relations that cannot be represented by ontological reasoning. For example, it could be necessary to express a prerequisite relationship between two topics in order to make correct suggestions to the learner. Thus, ontologies require a rule system to derive further information that cannot be captured by them, and rule systems require

ontologies in order to have a shared definition of the concepts and relations mentioned in the rules (Vesin et al., 2012). Rules also add expressivity to the representation formalism and reasoning on the instances (Henze et al., 2004).

A rule is built from an antecedent (a.k.a body) which implies a consequent (a.k.a head). Intuitively the meaning of a rule is: “whenever (and however) the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold” (Antoniou et al., 2005). One of the main benefits of using rules is to extend the expressivity of the ontological model. Another benefit of using declarative and formal rule is that it is understandable by both instructional designers and computer programs. Therefore, it can support the separation of personalisation components from the technical part of system by creating a bridge between the two parts. For instance, before recommending an IO to a particular learner the system should check the difficulty level of the IO and the ability of the learner for a possible match. The following rule declares that a learner with a high ability ‘has the ability to learn’ an advanced IO.

$$\text{Learner}(x) \wedge \text{InstructionalObject}(y) \wedge \text{hasAbility}(x, \text{High}) \wedge \text{difficultyLevel}(y, \text{Advanced}) \rightarrow \text{hasAbilityToLearn}(x, y)$$

Authoring for adaptation and personalisation actually consists of applying adaptation strategies and techniques to gain effective personalised learning content for individual learners and navigation sequencing (Aroyo & Mizoguchi, 2003). The rule language is the best candidate for expressing this adaptation due to its simplicity while they also provide sufficient functionality for instructional designers.

In our approach, Adaptation Model contains three groups of rules with regard to three types of adaptation namely *adaptive navigational support*, *adaptive presentation* and *adaptive guidance*. In the rest of the section, details of each one are described separately.

4.4.1 Adaptive Navigational Support

The goal of adaptive navigational support techniques is to help learners find an appropriate path in the learning by adapting the way of presenting links to the characteristics of an individual user such as current knowledge and needs (Brusilovsky, 2007). Several kinds of adaptive navigation support techniques are presented in the literature that we described in

Chapter 3. In our approach, navigation through the topics is presented to the learner in the form of an *Annotated Course Structure (ACS)* (Yarandi et al., 2013). ACS, which is a semantic representation of the course structure, allows the learner to select a topic by clicking on its related link. ACS uses link annotation and link dimming, two approaches commonly used in adaptive navigation. Link annotation adds different colours to the visible links in order to declare the educational status of the content behind each item in the ACS.

Link dimming is used to reduce the cognitive load which is on the learner as a result of course structure complexity. In our approach, access to topics which are considered too advanced for the learner is disabled, rather than fully hidden (Yarandi et al., 2013). This protects the learner from cognitive overload while also ensuring that the learner does not create a wrong mental map. In other words the dimmed links are visible for the learner, although they cannot be selected.

In our approach, a special set of pedagogical rules is authored to decide which topics should be enabled for the individual learner at a given moment and which should not. These rules take into account the current state of the learner's knowledge as reflected in the user model. For example the following rules are applied to identify a particular learner is ready to be learned a special topic:

```
Topic(t) ∧ Learner(x) ∧ PriorKnowledge(p) ∧ hasPriorKnowledge(x, p) ∧
relatedTopic(p, t) ∧ pkScore(p, v) ∧ greaterThanOrEqual(v, 50.0) → knows(x, t)

Topic(t) ∧ Learner(x) ∧ knows(x, t) → learned(x, t)

Topic(t) ∧ Topic(t1) ∧ Learner(x) ∧ knows(x, t) ∧ isPrerequisiteFor(t, t1) →
readyToLearn(x, t1)
```

The first rule defines that a specific topic is assumed to be known whenever the learner gets a score equal or greater than 50 from the corresponding assessment. The second rule defines that the learner learned a topic whenever she knows it. Finally, the third one defines that a specific topic is assumed to be ready to be learned whenever the learner knows the prerequisite of this topic. As a result of firing these rules the links to the concepts which are ready to be learned will be enabled for the learner.

4.4.2 Adaptive Presentation

The idea of adaptive presentation is to adapt the learning content accessed by a particular learner to preferences, ability, and other characteristics of the learner (Brusilovsky, 1996). For example, visual learners can be provided content with texts and pictures while a kinesthetic learner receives content which is accompanied by exercises so that learners can practice themselves. Therefore, the learning content is not static, but adaptively generated or assembled from existing IOs for each learner. In our approach, instructional designers can define different rules in order to adapt content to individuals according to different criteria. This research applies several instructional rules to infer the semantic relation between IOs and an individual learner to specify which IOs are suitable for the learner according to her needs. For instance, the following rule recommends suitable IO to particular learner based on learner's ability, learning style, language and also instructional plan:

$$\text{Learner}(x) \wedge \text{InstructionalObject}(y) \wedge \text{StaticIO}(t) \wedge \text{Language}(g) \wedge \text{hasAbilityToLearn}(x, y) \wedge \text{LSIsSupportedWith}(x, y) \wedge \text{nextIOType}(x, t) \wedge \text{selectedTopic}(x, d) \wedge \text{hasDomainTopic}(y, d) \wedge \text{hasIOType}(y, t) \wedge \text{hasLanguagePreference}(x, g) \wedge \text{isInLanguage}(y, g) \rightarrow \text{isRecommendedStaticIO}(x, y)$$

The above rules consider the following criteria which determined by other rules in order to recommend each IO to a specific learner (detail in Section 5.5.3):

1. The next learning task (e.g. explanation, example, exercise, test) which the learner is recommended to do through *nextIOType* and *hasIOType* properties.
2. The learner ability should be matched with the difficulties level of the IO through *hasAbilityToLearn* property.
3. Learner's learning style through *LSIsSupportedWith* property.
4. Learner's language preferences through *hasLanguagePreference*, *isInLanguage* properties.

4.4.3 Adaptive Guidance

Learners have a few pedagogical experiences to plan an optimal learning path for themselves. It is better the system guides them to determine an adaptive learning path. Adaptive Guidance implies guiding learners at runtime by observing learner's progress during a

learning session. Because this process occurs during runtime, any learner activity may affect the guiding process. Different guidance like learning a new topic, repeat this topic with more details, read more examples, doing more activities with lower or higher difficulty levels and repeat prerequisite topics, may be recommended to learners according to the learner progress and the relative status of current content. For instance, if the learner is given an interactive IO such as an exercise with moderate difficulty level and gives an incorrect response to it, she is guided to read more examples. After that, if she fails to respond correctly to it again, the learner is recommended to read learning content with lower difficulty level (with more explanation). The following rules define these two situations:

```
Learner(x) ∧ responseToIO(x, false) ∧ Interaction(p) ∧ hasInteraction(x, p) ∧
relatedTopic(p, d) ∧ selectedTopic(x, d) ∧ activityLevel(p, Moderate) →
isGuided(x, moreExample)
```

```
Learner(x) ∧ responseToIO(x, false) ∧ Interaction(p) ∧ hasInteraction(x, p) ∧
relatedTopic(p, d) ∧ selectedTopic(x, d) ∧ activityLevel(p, Moderate) ∧
isGuided(x, moreExample) ∧ difficultyLevel(p, Moderate) → isGuided(x,
repeatWithLowerLevel)
```

Similar rules have been defined for other guides. The defined rules can be executed using a rule engine. After firing the rule, the inferred knowledge can be written to ontology in order to update the knowledge base.

4.5 Design Issues in Modelling Learning Content

One of the key objectives of the semantic rule-based approach is to improve the flexibility of the system for generating personalised learning content. In order to meet this goal, we have defined two technical objectives that need to be fulfilled beforehand. First the learning content should be separated from the adaptation logic, which results in the learning content being no longer specific to any given adaptation rule or instructional plan. Secondly the learning content should be shaped in a fine-granularity level which we call Instructional Objects (IOs). Because the smaller the size of an IO, the higher the chance for it to easily fit in different applications, making it easier to dynamically generate personalised learning content for different learners and therefore increases the reusability of existing IOs. In this section, we describe the modelling of content which realise these two technical objectives.

4.5.1 Content Structure and Granularity

The content model represents the structure of learning objects. It should be able to describe learning content at different levels of granularity. For instance, some models define learning objects as lessons, while others define them as concepts, principles, facts or processes (Verbert & Duval, 2008).

The potential of repurposing a LO is directly related to the granularity of it. For example, if the learning content exists as a set of coarse grained LOs, then the probability of this LO being repurposed is significantly reduced. However, if it includes a set of finer grained learning content, its potential for repurposing is considerably increased. As mentioned in Section 4.3, in our system the smallest of learning objects are called Instructional Object (IO), each of which represents an elementary piece of content with a different instructional role in the given domain.

We should be aware that; if the IOs are ineffectively sequenced then the customised learning content produced may be incoherent. It is the instructional designer role to ensure that the adaptation model contains appropriate rules to produce coherent learning content within a logical flow.

In our approach, the content model distinguishes three types of learning objects at different granularity levels (Figure 4.4):

- **Instructional objects** are defined as content fragments in their most basic form of learning objects. Each of them focus on a single piece of information in order to be used to explain a concept, illustrate a principle or describe a process.
- **Lessons** are an aggregation of both the Instructional Objects and assessments on a specific topic. The assessment evaluates the learner's knowledge about the topic.
- **Courses** are a collection of Lessons that fulfil a learning objective. They represent the coarsest level of learning objects granularity.

The ontology-based content model which defines these aggregation levels and the relationships between them are detailed in next subsection.

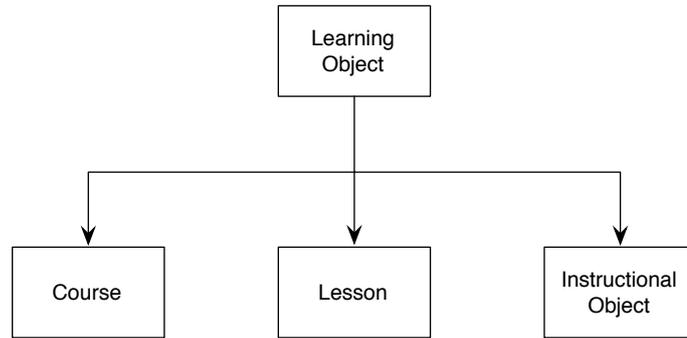


Figure 4.4: The different kind of Learning Object

4.5.2 The Content Model Ontology

The content model ontology provides a common representation of learning objects. It specifies content classifications and semantic relationships between learning content components in order to enable on-the-fly selection and composition of appropriate IOs to generate personalised learning content for different learners with diverse learning needs.

In our approach, an ontology-based model has been developed towards semantically enhancing learning content. A review of the existing literature indicates that semantically organized learning content has better potential to be repurposed (Gašević et al., 2007). We specified an ontology-based model that defines learning object granularity levels in a hierarchical structure with three general aggregation levels namely IO, Lesson and Course. Figure 4.5 depicts the structure of this ontology. Aggregation relationships between these levels are defined in the form of *hasPart* and its inverse *isPartOf* properties. The following classes represent these aggregation levels:

- ***InstructionalObject*** class presents the lowest level of the hierarchy without any child components. IOs are considered from the perspective of their instructional roles (Yarandi et al., 2013).

One should note that an IO can be assigned multiple pedagogical roles: each one defined from different perspective: A static or Interactive IO which is shown in Figure 4.6. Static IO is an instructional object which the learner can only read and there is no interactivity between the learner and the system in this period. Therefore, classes such as *Example*, *Definition* and *References* are defined as subclasses of *emphstaticIO*. Interactive IO represents different tasks that the learner does and

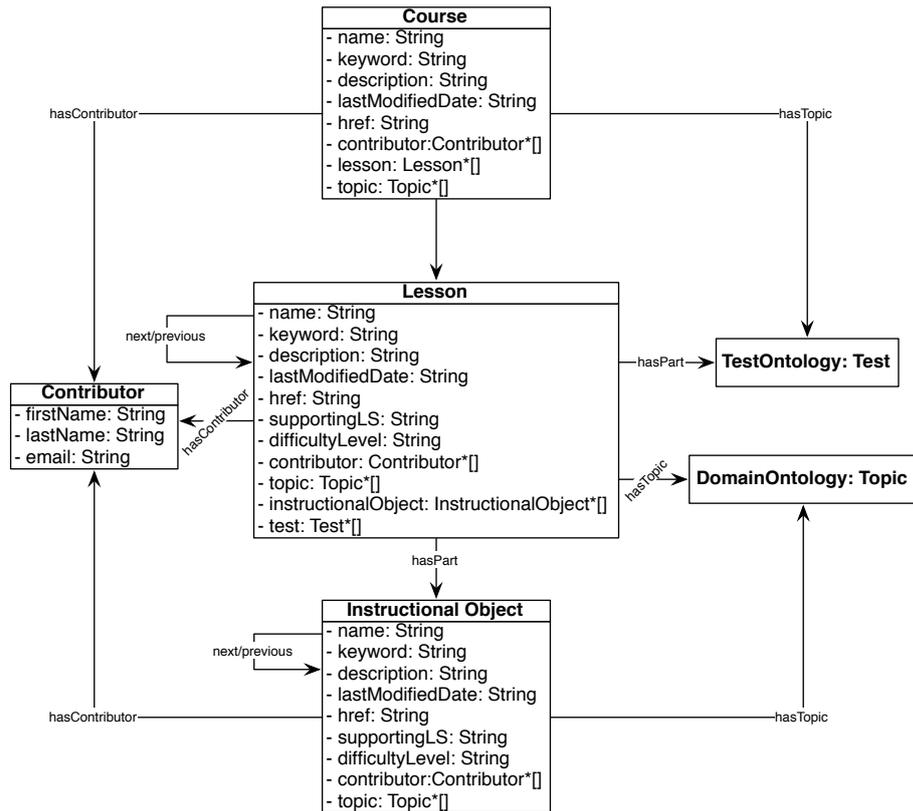


Figure 4.5: The Content model Ontology

while doing, the system receives her responses to it. Therefore, classes such as *Exercise*, *Simulation*, *Game* and *activity* are defined as subclasses of *InteractiveIO*. Interactive IOs are helpful in engaging learners in the learning process. Additionally, the system guides the learner and recommends her based on the learner's responses to interactive IOs.

- **Lesson class** represents an aggregation of both the *InstructionalObject* and *assessment* through the *hasPart* property. *Assessment* is a class of the assessment ontology to represent tests for evaluating learners prior to the start of the next lesson. Assessment ontology describes assessment specification which provides detail on the assessments like assessment topic, type, item complexity and item types.
- **Course class** is the first level of the hierarchy which consist of several *Lesson* classes determined via the *hasPart* property (Yarandi et al., 2012a).

The topic of each IO, lesson and course are determined by domain ontology (described in next sections) in order to semantically annotate them for further enhancing learning

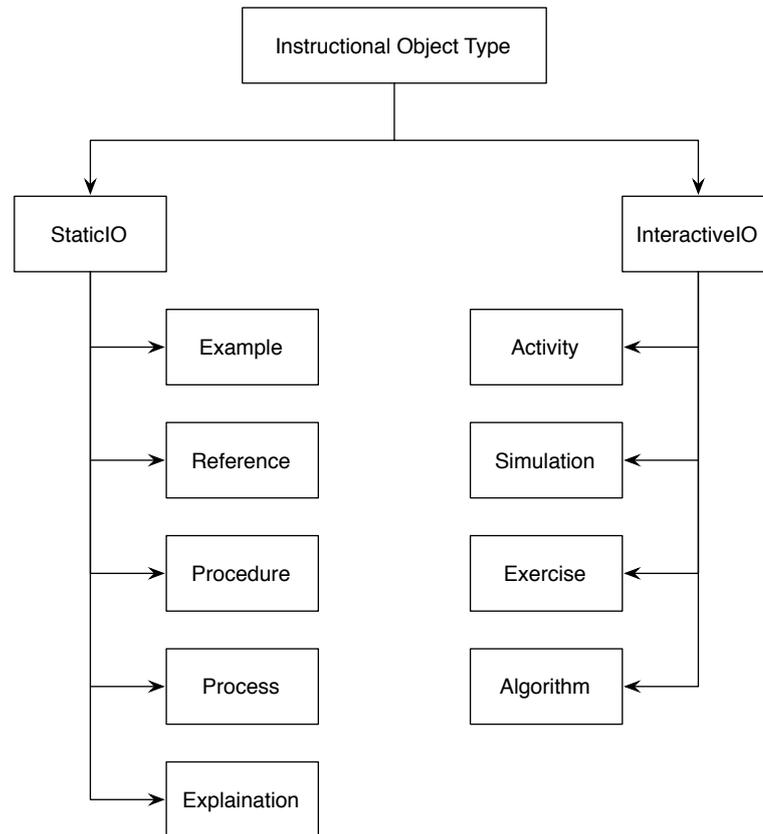


Figure 4.6: The different types of Instructional Object

content. Domain ontology describes learning content in terms of concepts of the topic domain and their semantic relationships.

Each LO is annotated with some metadata in order to be more easily accessible. Some of these metadata are *name*, *keyword*, *difficulty Level*, *href* (points to where the learning content is actually stored), *language*, *supportLS* (referring to the learning styles that an IO is particularly suitable for) and description.

Defined content model ontology facilitates semantically annotating the learning content components. Employing such semantic annotations, allows system to assemble existing IOs into new learning content personalised to the learner's preferences and needs.

4.6 Modelling the Learners

Learner model is a presentation of any information about an individual learner that is essential for providing adaptation in an adaptive system. Thus, building learner model

and updating it based on learner's interaction are important aspects in providing personalisation. This research describes learner's characteristics which have significant impact on the personalisation and how they can be modelled. There are several techniques for modelling the learner and refining this model (Yarandi et al., 2012d). Ontologies have been proven to be an effective means for presenting knowledge within a specific domain in a semantic way (Snae & Brückner, 2007). Consequently, we propose an approach where an ontological model is used to present the learner's characteristics. In this section, before describing the learner model ontology, some pedagogical issues which should be considered for modelling the learner are explained. After that, the applied mechanisms for building and updating this model are described.

4.6.1 Pedagogical Issues

The learner model is responsible for storing learner's information which is then used in the personalisation process to meet the needs of learners with varying backgrounds and preferences. As there is no agreement on the way learners preferred approaches for learning are modelled, the e-learning system should enable the instructional designers to influence the production of personalised courses at the following levels:

- Course structure and concept sequencing
- The actual core content

Our system utilises different domain ontologies, supporting different pedagogical approaches and also considers the learner's level of knowledge in order to structure the content, to be associated with a single course. This association and an appropriate adaptation model enables the system to deliver a personalised course that, while dealing with the same topic, can be structured in a way that best involves the learners characteristics. Using this mechanism the course author can implement several IOs. Each IO can be designed in a way that best supports a learner's preferences, abilities, learning style, and other specific features of a learner that is relevant for the learning process (Yarandi et al., 2013). Using appropriate adaptation rules, the instructional designer can influence the IO selection procedure. For instance, authors can produce alternative IOs based on the VARK model which is one of the most frequently used methods to describe different

learning styles (Fleming, 1987). The learner's VARK is inferred through filling the relevant questionnaires. The result of the questionnaires can be stored in the learner model. Each IO can also be designed in a way that is suitable for one of the VARK's aspects (visual, auditory, read/write and kinaesthetic) (Hosseini et al., 2013). The adaptation rules utilise the learner model and the content model to determine the suitable IO. Therefore, the same learning content can be presented in different ways to tailor learning content according to learner's learning style (Yarandi et al., 2012b).

4.6.2 Describing the Learner model ontology

The learner model ontology represents the personal information and learning characteristics of a particular learner and is utilised in the process of deciding the best teaching strategy for her (Yarandi et al., 2012d). The graphical representation of this model is shown in Figure 4.7. The main purposes of this ontology are to identify the user and the reason for adaptation and guidance based on the captured knowledge of users' profiles. Accordingly, three groups of information are represented in the learner model ontology.

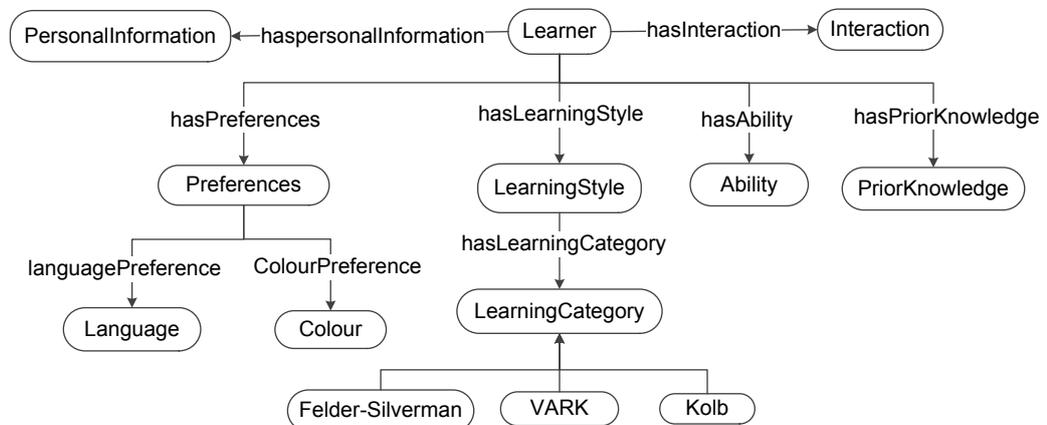


Figure 4.7: Learner Model Ontology

The first group is user identification information which contains an individual's basic information such as name, password and email, stored to identify a user. The second type of information is used for adaptation reasoning to infer the adaptive content and navigation including:

- **Learning Style:** learning styles are typically defined as the way people prefer to learn. For example some students prefer to learn through activities while others prefer

to learn by reading. The User ontology has the potential to represent different learning style models such as VARK(Fleming & Mills, 1992), Felder-Silverman(Felder & Silverman, 1988) and Kolb(Kolb, 1984). Each of these models consists of different categories which are presented in the ontology to realise the learning style of a particular learner (Hosseini et al., 2013).

- **Prior knowledge:** Prior knowledge helps to distinguish what learners already know and what they do not know. Having an idea about the level of learners' knowledge plays a significant role in the process of personalisation, as the rate of learner's knowledge assimilation is related to the learner's previous knowledge about the topic. This knowledge is what is gained from this system and the knowledge which she had when she started using this system. Therefore, learners' knowledge is acquired following the registration process and is updated based on test results from exams taken by the learner during the learning process. According to this captured knowledge, system adapts the annotated course structure. For example, when studying a maths topic such as "Adding Fractions" as a prerequisite, a learner needs to know about the topic of "Equivalent Fractions".
- **Preferences:** The learners may have different preferences related to some aspects of the learning environment such as colour and language. It is important for an adaptive e-learning system to present and organise the learning material based on the learner's preferences. In our approach, learners' preferences regarding language (e.g. English, French and Arabic) and colours, are used to direct the personalisation of adaptive learning.
- **Ability:** Adapting the learner's ability with the difficulty level of content is a critical factor for successful learning. Since learning content that is too easy or difficult to master dissatisfies the learner and results in inconsistent learning. Thus, this adaptation leads to better learning outcomes. It is more important in the e-learning systems, as there is no teacher to help learners. To obtain more precise estimation of learners' abilities, the result of tests taken by learners during the learning process are analysed according to IRT to estimate learner's abilities. Learners with lower abilities are instructed with IOs of lower difficulty levels, whereas learners with high abilities get IOs with higher difficulty levels.

Finally, the last group is interaction information which shows the teaching history to an individual learner. This information is used to identify learner's learning difficulties and to guide her during learning a particular topic. When a learner selects a topic to learn, the system provides personalised learning content and interactive activities for her and receives learner's responses. This information is stored in the learner model and will be updated during the learning process. The diagnosis process, consisting of finding and interpreting the learner's learning difficulties, is performed based on the information in the learner model. Then, the remediation process is done to assist the learner by either offering supplementary content to guide her; the learner model is updated accordingly. The next iterations of diagnosis and remediation process are done, if needed, based on the new information captured in the latest iteration. Finally, system prepares an assessment to evaluate the learner. As an example, the level of delivered activities to an individual learner is based on learner's responses to previous activities. If the learner has completed an activity of moderate difficulty and fails to answer the related questions correctly, she will be recommended another activity with lower difficulty level; while, if answered correctly, she will then be recommended an activity with higher difficulty level.

4.6.3 Acquiring and Updating the Learner Model

In order to provide personalisation in e-learning systems, it is necessary to store the learner's characteristics (e.g. abilities, preferences, prior knowledge and learning styles) in the learner model. Some of these characteristics are static whereas others are dynamic. Static features are initialised in the registration period, and they usually remain unchanged throughout the learning process. These characteristics are learner's email, preferences, etc. On the other hand, dynamic features are updated during learning process based on the interaction of learner with the system, for example, learner's scores, abilities and knowledge.

At the start of the first session learners should complete a registration process. During this process general and educational characteristics of individual learners are recorded and the first version of the learner model is created (Yarandi et al., 2013). For this purpose, the system presents newly subscribed learners with a questionnaire to fill in, in order to determine the learner's learning styles. As the instructional designer can select different

learning style models (e.g. Kolb, Felder-silverman and VARK), the system provides them with facilities to utilise the related questionnaire. For instance, if the instructional designer wants to use the VARK model, they can use its respective questionnaire.

To infer the learner's level of knowledge at the start of the course, there are two optional mechanisms:

- explicitly querying learners
- taking a pre-test from learners

As the adaptive decisions are managed by the course authors, facilities are provided for them to select one of these techniques. As the system is fully ontology based, the different topics or related pre-tests that are shown to the learner, to get direct feedback or test results for estimating his level of knowledge, are from the domain or assessment ontology respectively.

The learner model gets progressively updated following learners' interaction with the system. In details, learners are engaged in learning conceptually pre-defined topics, complete activities and take tests; while the system should continuously recognise changes in the learner's knowledge and ability as they progress and update the learner model accordingly (Yarandi et al., 2011*b*). Tests are utilised to assess learner's knowledge about each topic in the domain. The result of the tests are analysed according to item response theory (see Chapter 2) to estimate learner's ability. Newly acquired learner's ability and knowledge levels are updated dynamically throughout the learning process in order to adapt the learning material to the learner's updated features in the next step (Hosseini et al., 2013).

4.7 Describing the Domain Model

The domain model represents the structure of a course by defining all topics which are covered in the course and the semantic relations between them (e.g. the prerequisite relationship). It describes the concepts sequencing, developed by domain experts to realise the order in which the topics are to be taught. Therefore, the domain experts should define sufficient learning topics and these topics should be sequenced appropriately. The aim of

domain model is to represent the sequence of topics without referring to content model. Defining two discrete domain and content models promote the flexibility of system and the reusability of components as it enables the course author(s) to modify navigation and presentation adaptivity separately. For instance, if an author wants to modify just the adaptive navigation of a course she needs only to change the domain model that implements it.

4.7.1 Defining Topics and Pedagogical Relations

In our approach, domain ontology formally defines the topics covered in the learning system and relations between them. This ontology contains a variety of classes and properties to describe the content structure. The classes are utilised to organise different topics and subtopics in a hierarchy representation and properties are utilised to define pedagogical relations between topics belonging to different branches of the hierarchy.

Semantic rule-based approach is domain independent and fully ontology-based. Domain independence is achieved by using domain ontology as the only source of knowledge about different topics of domain and pedagogical relationship between them. Although the approach supports different learning domains through defining related domain ontology, for the purpose of this thesis the topic of Fraction from the mathematics domain is used to describe the system. A segment of the domain topic is presented in Figure 4.8.

Each topic in the domain has several associated subtopics. If the learner model states an individual learner has learned a higher topic in the hierarchy, we can presume that she has also learned all subtopics for that topic and as a result it saves duplication in the learner's model. Also, if the learner passes all subtopics of a special topic, the system infers that she knows this topic. For example, the "Add Fraction" topic has two subtopics "Add Same Denominator" and "Add Different Denominator". This means that in order to learn the topic "Add Fraction" learners have to learn both topics "Add Same Denominator" and "Add Different Denominator" without considering them in a specific order. To represent the sequencing of topics in terms of the order in which the topics are to be presented to learners, two pedagogical relations are defined namely "is Taught After" and "is Taught Before" between topics in the ontology. Prerequisite relations are represented in the form of "has Prerequisite" and its inverse "is Prerequisite For". If topic T2 *hasPrerequisite* relation with

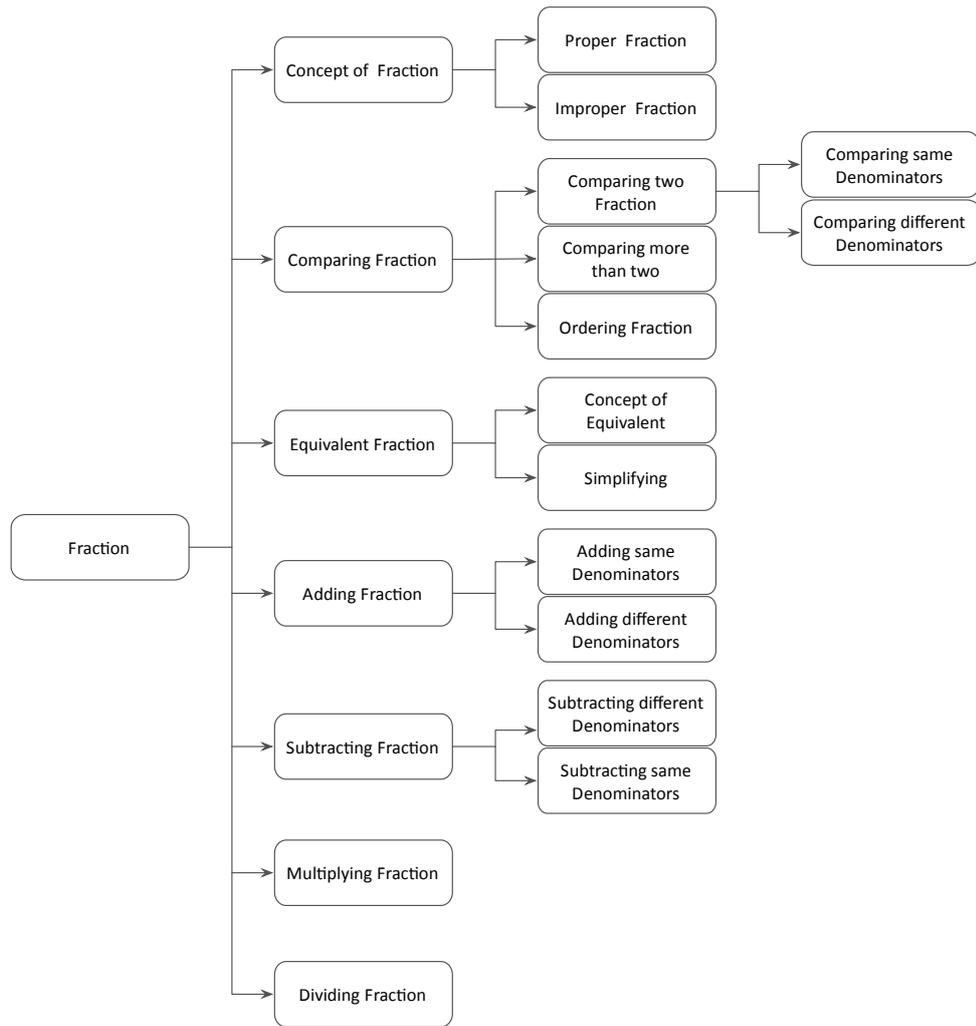


Figure 4.8: A segment of domain topics

T1; it means that the learner must learn topic T1 before topic T2 (it is necessary to know topic T1 for learning topic T2). However, in the case where *isTaughtBefore* relation is defined between topic T1 and T2; this relation states that it is preferable that topic T1 is presented to the learner exactly before topic T2 but this is not mandatory (the sequences of topics). A semantic relationship named *isRelatedTo* is defined to present pedagogical relation between topics belonging to different branches of the hierarchy. Each defined topic is assigned one or more alternative topic names using *preferredName* property. We designed this ontology to annotate LOs and their component regarding to different domain topics. Figure 4.9 depicts an excerpt from the domain ontology which shows a segment of the topics and their relation to each other.

The domain model holds the structure and knowledge of the domain of learning. This

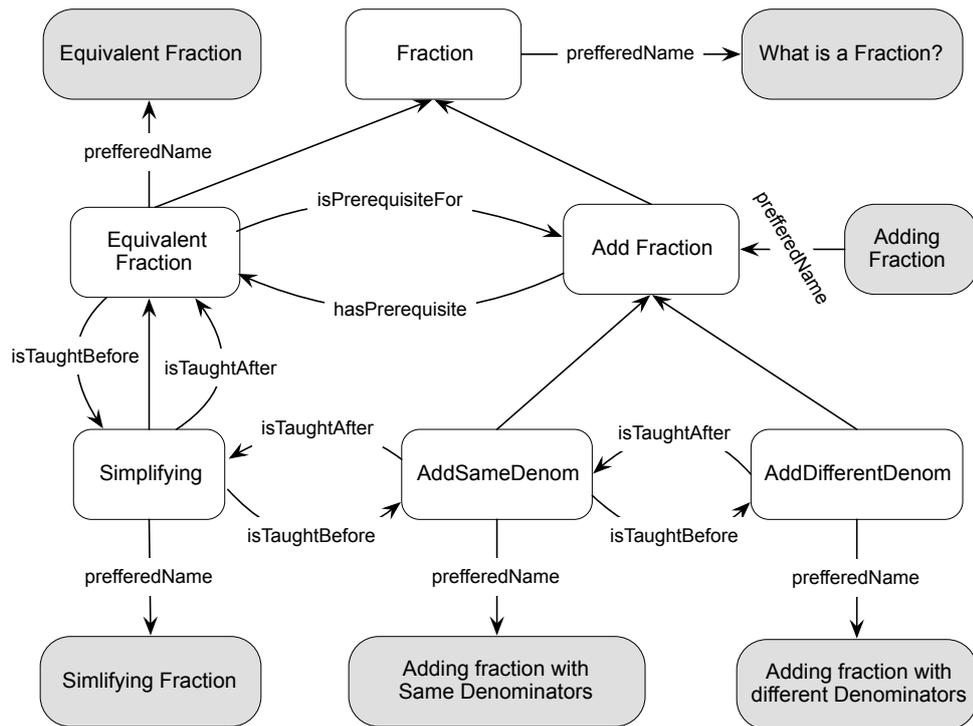


Figure 4.9: An excerpt from the Domain ontology

model is what the domain experts work on in order to update and modify the structure of a course. However, it is a separate and discrete model which is not entangled with the core of the system. Additionally, the pedagogically sound relations defined among topics enable the system to present adaptive structures to learners. The domain experts are responsible to define all topics which are essential to cover a particular course and also to define semantic relations between the topics, which enable system to determine the sequences of the topics.

4.8 Describing the Assessment Model

In our approach, assessments are one of the most significant means of estimating the learner's ability and also identifying how much knowledge the learner has acquired during her interaction with the system. It is usually performed through a set of assessments delivered by the system and solved by learners. However, estimation of a learner's knowledge about a particular topic is not as simple as rating how many correct answers the learner had on the assessment. Many other factors influence her knowledge level, such as the difficulty level of the test items and item's guessing degree. Therefore, the assessment

process aimed at calculating a learner's knowledge has to include advanced mathematical techniques based on the assessment results and diverse factors to generate a more accurate result.

Additionally, assessments are used to guide the selection of the next learning actions. As the quality of the learner model and the adaptation process depend on the quality of information gained from the assessment process, test developer is responsible to design high quality tests and examine their quality constantly and carefully.

Assessment models describe the specification of assessments and their components to facilitate the reuse of assessment components (e.g. item) in different exams. Using semantic technologies in particular ontologies, make it possible to enrich the assessment process in a more effective way. Consequently, ontology-based assessment model facilitates identifying learning needs (through calculating learner's ability), monitoring learner's progress, maintaining learner's teaching history and reusing test components. In this section, we describe the calibrated tests based on item response theory and the details of assessment ontology which is designed to fulfil the objectives of the thesis.

4.8.1 Test Calibration using Item Response Theory

As we mentioned in Chapter 2, the main purpose of Item Response Theory (IRT) is to estimate a learner's ability or proficiency (Wainer et al., 2000) according to her responses to test items (Yarandi et al., 2011a). To estimate a learner's ability, it will be assumed that the numerical values of the parameters of the test items are known. Consequently, the metric of the ability scale will be the same as the metric of the known item parameters. After completing the item and receiving the response from the learner to all items of tests, the items are dichotomously scored. This means that the learner gets one for the correct answer and zero for the incorrect answer. Hence, we will have a list of 1's and 0's for all the items which is called activity response vector. This item response vector and the item parameters are used to estimate the learner's ability. Therefore, it is essential to determine the values of the item parameters in a metric. In IRT, this task is called test calibration.

For calibrating items, at the first step the test developer designs suitable items from each topic. Then these items are presented to students to answer them. The collected item

response data are analysed, according to IRT to obtain the appropriate item parameters. Therefore, each item has its own difficulty, discrimination, and guessing parameter which is called calibrated item. The calibrated item can be used to estimate learner's ability (Yarandi et al., 2011a).

4.8.2 Describing the Assessment model ontology

The assessment and evaluation of the individual knowledge is a major aspect of the learning personalisation. Assessments of what the learner knows are used to personalise the learning curriculum, assigning remedial problems in areas where the learner is weak. In our approach, an ontology-based assessment model is developed to formally represent relevant information about a test, specially the parameters which are used to calculate learners' ability based on IRT. It also enables annotating the tests and their components semantically, which facilitate reusing tests' components in different assessments. The concepts defined in the assessment ontology are shown in Figure 4.10. The main concept of the assessment ontology is the test class which provides a set of properties to model some specific features of a test. Each test is uniquely identified by an ID number, which is kept as a value of the *testID* property. The *isAbout* property refers to the topic of the test which is an element of the domain ontology. Each test has arbitrary items associated; the *hasItem* property holds instances of the Item class.

At the centre of Figure 4.10, the *Item* class is used to describe an item through data and object properties. Each item is identified by an ID via the *itemID* property. One of the characteristics of the IRT is that it measures the ability of learners and items difficulty with the same scale allowing to easily compare and calibrate them. Once the item is calibrated the system will be able to estimate learner's ability levels. Three data properties are attached to the *Item* class to represent the parameters which are needed for calibrating an item. These three properties are *difficultyParameter*, *discriminationParameter*, and *guessingParameter*. Each item has a stem which identifies the question asked from the respondent. The *score* property presents the score that a learner acquires if she answers correctly to this item. Additionally, three types of items are defined in the ontology: *multiple choices*, *short answer* and *true/false*. We utilise these types as they are considered objective assessments, thus they eliminate the subjectivity in the rating.

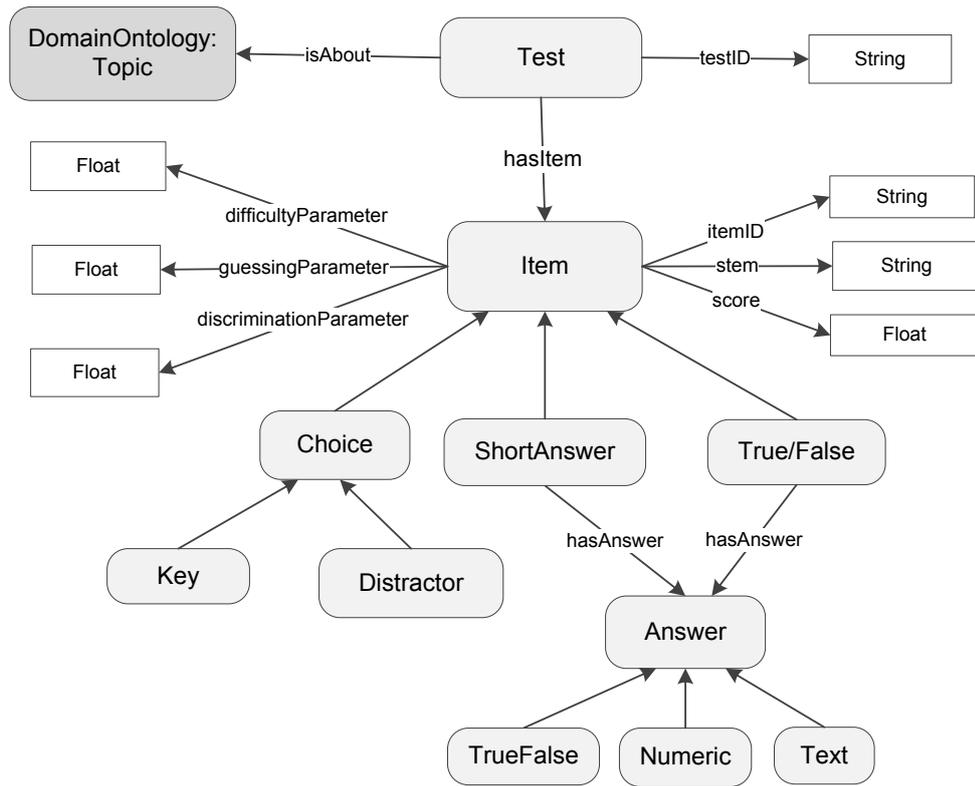


Figure 4.10: Test Ontology

A multiple choice item contains a set of options. The options are the possible answers that contain a key which is the correct answer to the item, and a number of distractors that are plausible but incorrect answers to the item. In short answer item, the learner is required to supply a few words that complete a given statement or to label various parts of a diagram. The answer may be numeric or text. In true/false item, the learner is presented with a statement which is followed by two alternatives (e.g. true/false), only one of which is correct.

4.9 The required Technologies for the Semantic Rule-Based Approach

The semantic rule-based approach calls for some requirements to be fulfilled by the implementations they rely on. The most important of these requirements, which influences the technologies those implementations utilise, is how knowledge is represented.

As data modelling and knowledge representation are crucial for the semantic rule-based

approach, a formal modelling language is necessary to enable us to implement the Adaptive Personalised e-learning system. Therefore, the Ontology Web Language (OWL) which is a family of knowledge representation languages for authoring ontologies is employed for this implementation (Bechhofer et al., 2004).

Additionally, among implementation needs are technologies that support the modelling, manipulating, serialising and parsing of such models. In order to produce adaptive effects, Rule-PAdel needs an expressive rules language and a reasoning engine for interpreting the rules.

4.9.1 Data Modelling with OWL

The Web Ontology Language, OWL (Patel-Schneider et al., 2004), is a knowledge representation language for authoring ontologies. OWL facilitates greater machine interpretability of human knowledge by providing additional vocabulary and formal semantics. In Rule-PAdel system, the components of personalisation are modelled using OWL.

OWL is based on description logic, thus its construction has well-defined meanings which are used to describe domain concepts and their relationships in an ontology. For instance, in the domain of adaptive learning, concepts such as *Learner* or *Topic* might be modelled as classes in OWL. For example, a learner called *John* is created as an individual of the *Learner* class. Also, *Add Fraction* is created as an individual of the *Topic* class. If *Learner* and *Topic* have a relationship such as *Learners know a specific Topic*, this relationship can be created in OWL as a link between *Learner* and *Topic* concepts and named *knows*. The existence of this generic, somewhat abstract relation, would allow stating specific knowledge in a given setting (called facts or assertions), such as *John knows AddFraction*. Furthermore, OWL offers different constructions for expressing further restrictions on the relationships among concepts, including cardinality and domain and range restrictions such as *union* and *disjunction*. It also has a rich vocabulary for describing relations among classes, properties and individuals. For instance, a class can be an *IntersectionOf* or a *UnionOf* some other classes. Additionally, we can state that a property is *Transitive*, *Symmetric*, *InverseOf* another one, or *EquivalentOf* another one. Also, we can specify that a class instance is the *SameIndividualAs* another instance, or is *DifferentFrom* a certain other instance.

As a result of formalising the descriptions of personalisation models in OWL we are able to support reasoning on knowledge base, reusing data and sharing data. OWL also enables the inferring new knowledge that is not explicitly stated in OWL ontologies. It also has some appropriate features including valuable expressive power, formal syntax and semantics, and practical reasoning systems. These features make it a suitable language for representing ontology.

4.9.2 Semantic Rules using SWRL

SWRL (Semantic Web Rule Language)(Horrocks et al., 2004) is a semantic rules language based on a combination of Ontology Web Language and Rule Markup Language (RuleML, 2012) for formalising the expression of rules. It is an emerging XML-based framework for building rules on top of OWL ontology.

OWL has a set of basic implicit reasoning mechanisms based on description logic. OWL needs additional rules to express relations that cannot be represented by ontological reasoning. Ontologies require a rule system to make further inference for deriving further information that cannot be captured by ontologies, and rule systems require ontologies in order to express rules in terms of OWL concepts and relationships. Rules can be used to infer new knowledge from existing OWL knowledge bases. In our approach, SWRL is used as a reasoning and inference mechanism and to express adaptation techniques in Rule-PAdel system (see Section 4.4 for further explanation).

SWRL extends OWL's expressiveness while preserving a simple syntax. It is also compatible with OWL syntax and semantics, since they are both combined in the same logical language. It extends the set of OWL axioms to enable Horn-like rules to be combined with an OWL knowledge base. It also allows developer to use a variety of rule engines to reason with the SWRL rules stored in an OWL knowledge base.

In Rule-PAdel system, adaptation decisions are presented using SWRL that are not easily or naturally modelled within OWL. The logic underlying the adaptation is explicitly captured on the basis of a rule-based model. As a consequence of executing the rules, recommendations are generated in order to implement the concept of adapted content, adapted navigation and learners' guidance. Moreover, the rules can be easily modified for

specific adaptation requirements, thus increasing the flexibility and extensibility of our system. The course author and instructional designer can modify the rules and add new rules in adaptation model for using different pedagogical strategies without impacting on program coding.

4.9.3 OWL Manipulating, Serialising and Parsing using OWL API

The OWL API(Horridge & Bechhofer, 2011) is a high-level Application Programming Interface (API) for creating, manipulating OWL Ontologies. The higher-level abstractions of the API help to insulate developers from underlying issues related to serialisation and parsing of data structures. Serialisation Produce OWL concrete syntax from some internal data structure or representation. Parsing issues a sequence of change events to the API in order to build an in-memory representation of an ontology. It supports reading and writing ontologies in several syntaxes, such as RDF/XML, OWL/XML, OWL Functional Syntax and the Manchester OWL Syntax.

The OWL API supports loading and saving ontologies in a variety of syntaxes. It contains several powerful and flexible packages such as model, change and interface packages for supporting the use of OWL ontologies within applications(Bechhofer et al., 2003). The model package includes methods for accessing the *Classes* defined or used in the ontology, the *Properties* defined or used, *Axioms* asserted and so on. The change package extends the access to ontology by allowing manipulation of those structures for example adds and removes entities and also changes to definitions, axioms and so on. Inference package implements the formal semantics of the language.

Within Rule-PADEL, OWL API is used for inserting and manipulating data in different ontological models such as adding new learners to the ontology, updating the learner model during the learning process and inserting new instructional objects to content ontologies.

4.9.4 OWL Reasoning using Pellet

A semantic reasoner or rules engine is able to infer logical consequences from a set of asserted facts or axioms. Pellet (Sirin et al., 2007) is an open source, Java reasoner for

OWL ontologies. It provides standard and cutting-edge reasoning services and can be used with both Jena (McBride, 2002) and OWL API libraries to provide reasoning. It provides functionalities to see the species validation, check consistency of ontologies, classify the taxonomy and check ontologies (Sirin & Parsia, 2004). Pellet is an OWL DL reasoner using the tableaux algorithms (a decision procedure that aims to determine the suitability of an input formula in a given logic) which is provably complete. Pellet supports reasoning with SWRL rules. Pellet interprets SWRL using DL-Safe Rules notion. There is no need for using any additional utility function to use SWRL in Pellet.

4.9.5 Data Representation using XHTML

XHTML¹ is a family of XML Markup Languages that extend versions of Hypertext Markup Language (HTML), the language in which web pages are written. The structure of the different models used in the semantic rule-based approach is represented through OWL language. However, different models do not include actual IOs and assessments and only include their IDs. Consequently, when the personalised content is ready to be delivered, the actual IOs and assessment are attached to the personalised content. As personalised learning content created by system are delivered via the web, their textual IOs and assessment are written in XHTML and may include image files, Flash animations and audio or video content that can be delivered via a web browser.

4.9.6 Delivering Content with Apache Tomcat

After the personalised learning content is generated in the JSP format, it should then be delivered to the learner using Apache Tomcat², which is a web server for Java Servlets and JSPs. Rule-PAdel system uses Tomcat for robust delivery personalised learning content to client. Tomcat and JSPs provide facilities to access Rule-PAdel system through a web interface. This supports learners' interaction with the system via a web browser.

¹<http://www.w3.org/TR/xhtml2/>

²<http://tomcat.apache.org/>

4.10 Summary

This chapter has presented the design of the semantic rule-based approach for developing adaptive systems. It also illustrated the features of this approach needed for developing a flexible and extensible personalised e-learning system. These features are the separation of the models using ontology, defining adaptation strategies using semantic rules, and defining content in finer granularity levels. In this chapter we also studied the factors that have an effect on designing and implementing the models. With respect to the content model, issues such as content granularity level were discussed. For learner model, pedagogical consideration and model's manipulation were discussed. The domain model was then presented from the perspective of domain independence, and assessment model was detailed with respect to accuracy of evaluation.

Lastly, the required technologies for implementing the Adaptive Personalised e-learning system, based on semantic rule-based approach were discussed.

Chapter 5

Implementation

5.1 Introduction

This chapter presents the technical aspects of the implementation of our semantic, rule-based approach for personalised learning which was described in Chapter 4. Rule-PAdel, the Ontology driven approach for Personalised Adaptive e-learning system, is presented as an implementation of this approach. This chapter will start by illustrating the technological architecture of Rule-PAdel system which presents the technologies used to support our approach. After that, in Sections 5.3 and 5.4 the implementation issues of learning content and learner model, as the two most important models of system, are presented. The implementation of adaptation model using SWRL rules is discussed after that in Section 5.5. Section 5.6 discusses how adaptation rules are executed and how learner's ability is calculated based on item response theory. Also discussed in this chapter is the cycle of creating a personalised learning content for Rule-PAdel. Finally, the chapter finishes by discussing an instance of an e-learning system that has been created with Rule-PAdel to learn fractions in mathematics domain.

5.2 Technological Architecture of Rule-PAdel

This section presents a technological view of the Rule-PAdel architecture which corresponds to the architecture of the semantic rule-based approach described in Section 4.3.

This architecture shows the way technologies are applied to support the implementation of Rule-PAdel system. Figure 5.1 shows the technological architecture of Rule-PAdel.

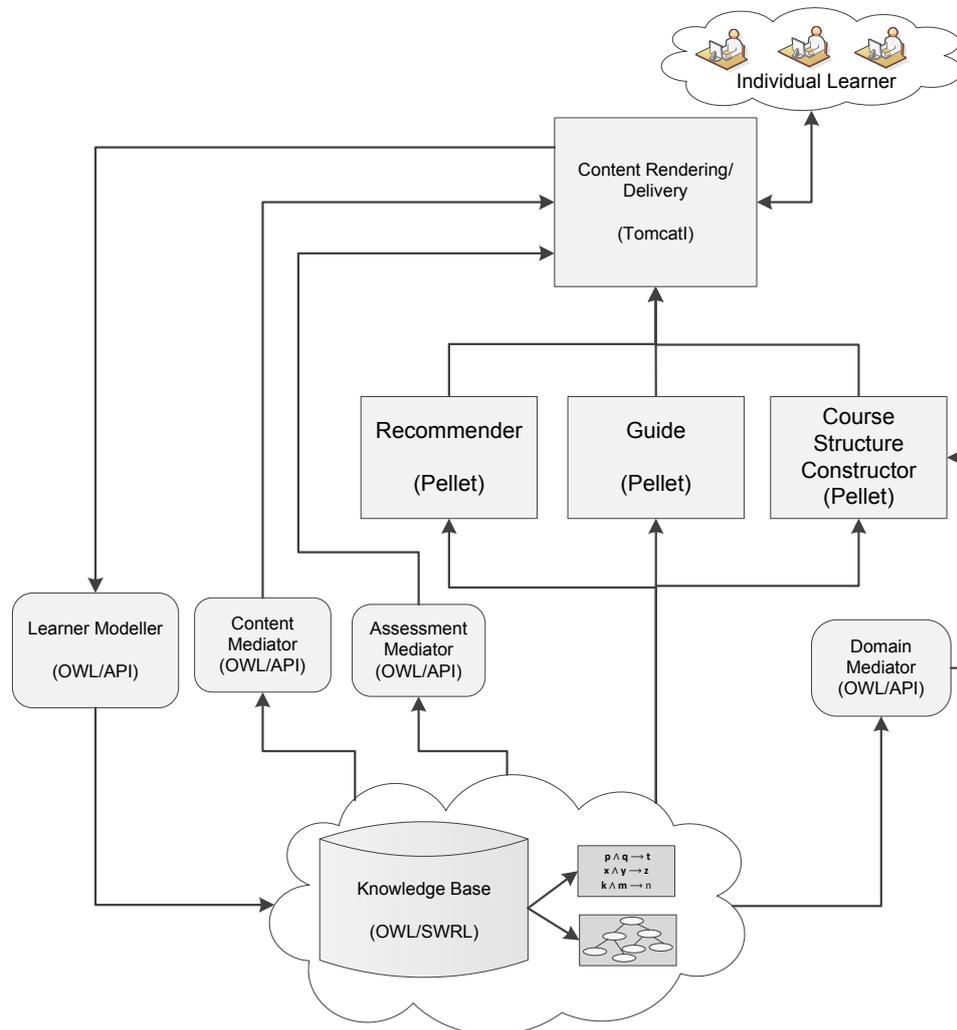


Figure 5.1: Technological Architecture of Rule-PAdel

The functionality of the components from a technological perspective is described below:

- **The Rule-PAdel Knowledge base** is an information repository in the form of ontologies and rules. After constructing the ontological knowledge model (e.g. domain, content, learner), OWL is employed for representing the knowledge base. OWL can define the structure of data by describing and categorising concepts within the domain and relations between pairs of concepts. It can be used to model the domain and support reasoning about the concepts. The adaptation model is established using SWRL rules in order to empower the knowledge base. After firing the rules, the inferred knowledge is used to update the knowledge base at runtime. The updated

knowledge base contains all the knowledge necessary for the adaptation process.

- **Learner Modeller** makes a copy of the learner model and keeps it in an accessible memory. It increases the performance of the system as the frequency of accessing the learner model in the knowledge base is considerably reduced. Learner modeller reads and writes information to/from the current learner model through OWL API facility. The OWL API supports loading and saving ontologies in a variety of syntaxes including OWL. It also facilitates accessing OWL reasoners such as Pellet (See Section 4.9).
- **Content Mediator** is responsible for handling requests for interacting with the knowledge base to retrieve the requested IOs. It uses the OWL API facility for accessing the content ontology. As the personalised learning content includes only the *id* for each IO, rather than the IOs themselves, the mediator is also responsible for finding appropriate IOs from the knowledge base and sending them to the *content rendering and delivery* component to be presented to the learner.
- **Assessment Mediator** is responsible for finding the appropriate assessment from the knowledge base based on some given criteria. The mediator connects to the ontology and retrieves the information through OWL API. It is also responsible for retrieving the assessment from the knowledge base and presenting it to the learner through *content rendering and delivery* component, as the selected assessment only contains the assessment *id*.
- **Domain Mediator** is responsible for making a copy of the domain model and keeping it in an accessible memory using the OWL API. Since the domain model does not change during the learning process, this gives the system quick access to the structure of the course and the relation between the topics, thus increasing the performance of the system.
- **Course Structure Constructor** performs the process of constructing the Annotated Course Structure (ACS) by using link annotation and link dimming techniques which helps the learner in navigating the domain space. This constructor gets the structure of the course from the domain mediator. It also gets the learner's current knowledge from the knowledge base using Pellet to build the ACS. Pellet is an open source Java reasoner for OWL ontologies. It can be used with the OWL API library to provide

reasoning over ontology and SWRL rules (See Section 4.9). Pellet makes inferences over the factual knowledge and this component then utilises that inferred knowledge which is related to the learner's knowledge level.

- **Recommender** is responsible for generating personalised learning content. When the SWRL rules are executed by Pellet after the factual knowledge is provided, this component gets the new inferred knowledge which is about the recommended IOs. After that, it selects and assembles appropriate IOs to generate the personalised learning content.
- **Guide** configures different settings of the system in order to prepare it for executing the guidance. For example, if the system guides the learner to read more examples, this component makes the required adjustments to the system so it can present the related example to the learner.
- **Content Rendering and Delivery** is responsible for delivering the output of the adaptation process to the learner which includes the adaptive course structure, adaptive content and adaptive guidance. The personalised content does not include actual IOs and assessments and only their ids. Consequently, the rendering process should attach the actual IOs and assessments in the learning content.

After transforming the personalised content into the JSPs (Java Server Pages), the IO contents and assessments are attached to it. Tomcat ¹ is used to deliver the personalised learning content to the learner. The course model returns information to Rule-PAdel from the user using JSP to capture user's interaction and input to the system. Following the initial delivery of the content and receiving user's input into the system, the system delivers its feedback and subsequent personalised content to the user using AJAX (Garrett, 2005) to avoid the need for a complete page refresh.

Ajax (Asynchronous JavaScript and XML)(Garrett, 2005) is a web development technique used on the client-side for asynchronous communication between servers and clients. With Ajax Rule-PAdel sends learning content and system's feedbacks to the learner's machine, and retrieves users input and interactions to the server asynchronously without interfering with the existing page.

¹<http://tomcat.apache.org/>

5.3 Implementation Issues of Content Modelling

Rule-PAdel is a generic personalised learning, that is, independent from the actual content. This makes the system applicable to different domains.

Two different ontological models are defined to represent the domain content namely domain and content model. Domain model presents domain structure by defining the topics covered in Rule-PAdel and semantic relation between domain topics. Content model represents the educational features of content specially, IOs as smallest pieces of content (See Chapter 4). In this section, the implementation issues of domain model and instructional objects are described in detail.

5.3.1 Domain Structure

The domain structure organises a set of topics covered in the system into a hierarchy and set semantic relations between these topics. The OWL language is used to design ontological modelling of domain structure. OWL provides constructs for defining the class and subclass which we use to describe the topic hierarchy. Accordingly, a topic can be defined as a class and its subtopics can be defined as its subclasses. These definitions enable the system to infer that when an individual learns all subtopics of a particular topic then she learns this topic. On the other hand, if she intends to learn a particular topic, the system infers that she must learn all subtopics of the desirable topic. For example, the adding fraction with same denominators and different denominators are to subtopics of add fraction topic. Following OWL code shows the definition of this example via *subClassOf* construct:

```
<owl:Class rdf:about="&DomainMath;AddDifferentDenominator">
  <rdfs:subClassOf rdf:resource="&DomainMath;AddingFraction"/>
</owl: Class>

<owl:Class rdf:about="&DomainMath;AddSameDenominator">
  <rdfs:subClassOf rdf:resource="&DomainMath;AddingFraction"/>
</owl: Class>
```

The semantic relations between various topics are defined through defining different properties including *isRelatedTo*, *hasPrerequisite*, *isPrerequisiteFor*, *isTaughtAfter* and *isTaughtBefore*. For example, *isRelatedTo* property links two topics which are semantically related to each other. As we mentioned previously, OWL vocabulary is rich in describing

relations among classes, properties, and individuals. We use these facilities to define the *emphasRelatedTo* property as a transitive and symmetric property. The transitive relation indicates that if topic A *isRelatedTo* topic B and topic B *isRelatedTo* topic C, thus the system can infer that topic A *isRelatedTo* topic C. Additionally, the symmetric relation indicates that if topic A *isRelatedTo* topic B, thus the system can infer that topic B *isRelatedTo* topic A as well. The following OWL fragment shows the definition of *isRelatedTo* property:

```
<owl:ObjectProperty rdf:about="&DomainModel;isRelatedTo">
  <rdf:type rdf:resource="&owl;SymmetricProperty"/>
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <rdfs:range rdf:resource="&DomainModel;Topic"/>
  <rdfs:domain rdf:resource="&DomainModel;Topic"/>
</owl:ObjectProperty>
```

Using OWL vocabulary we can also define that a property is *InverseOf* another one. For example, in Rule-PAdel *emphasPrerequisite* is defined *InverseOf* *emphasPrerequisiteFor* property and *versa visa*. The following OWL code shows these definitions:

```
<owl:ObjectProperty rdf:about="&DomainModel;hasPrerequisite">
  <rdfs:domain rdf:resource="&DomainModel;Topic"/>
  <rdfs:range rdf:resource="&DomainModel;Topic"/>
  <owl:inverseOf rdf:resource="&DomainModel;isPrerequisiteFor"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:about="&DomainModel;isPrerequisiteFor">
  <rdfs:domain rdf:resource="&DomainModel;Topic"/>
  <rdfs:range rdf:resource="&DomainModel;Topic"/>
  <owl:inverseOf rdf:resource="&DomainModel;hasPrerequisite"/>
</owl:ObjectProperty>
```

Two properties *isTaughtAfter* and its inverse *isTaughtBefore* are defined to represent the sequencing of topics in terms of the order in which the topics are to be presented to learners. These two properties are semantically different from *hasPrerequisite* and *isPrerequisiteFor* properties. If topic T2 *hasPrerequisite* relation with T1; it means that the learner must learn topic T1 before topic T2 (it is necessary to know topic T1 for learning topic T2). However, in the case where *isTaughtBefore* relation is defined between topic T1 and T2; this relation states that it is preferable to topic T1 presents to learner exactly before topic T2 but this is not mandatory (the sequences of topics). The inversion, symmetry and transitivity properties offer inference capabilities for the system.

5.3.2 Instructional Objects

In semantic rule-based approach instructional object (IO) is the finest-grained content which can be aggregated to generate learning content. As personalised learning content created by Rule-PAdel are delivered via the web, thus their IOs may have any format including XHTML, image files, Flash animations and audio content. Textual IOs are written in XHTML² format, thereby, they can be tagged by CSS (Cascade Style Sheet) identifiers for describing the presentation semantics. Therefore, the course authors or designers are enabled to create different styles for learners with different preferences. for example, a textual IO can be marked up as follows:

```
<DIV class=Heading>
  <B id=highlight>
    Example of Fraction
  </B>
</DIV>
<DIV class=Example>
  <p>If you divide a circle into eight equal pieces and select five of them,
  you have selected 5/8 a circle. </p>
  <DIV class=Image>
    <p> <img width=200 height=187
      src= "../Math/properFraction/p_emp_mod2.png" />
    </p>
  </DIV>
  <p>The <B id=highlight> denominator </B>
    8 tells us that the unit has 8 equal parts. Five of the parts is selected
    for a <B id=highlight> numerator of 5</B>.
    This fraction can also be written as five-eighth. 5/8 is example of
    fractions - parts of a whole.
  </p>
</DIV>
```

The HTML refers to the CSS by using the attributes *id* and *class* to change size, colour and other layout and styles of the presentation.

5.3.3 IO Annotation

In Rule-PAdel system, Ontology-based annotation is utilised to describe the uploaded IO with high-quality metadata. The abstract description of learning content through ontology-based annotation makes further information about the instructional objects, this enables the adaptive engine to choose appropriate IO for a particular learner between several candidates. In our approach, the content and domain ontology are used to semantically annotate each IO.

² <http://www.w3.org/TR/xhtml2/>

The content ontology is used to present an IO, whereas the domain ontology is used to semantically annotate an IO with concept from the domain ontology which shows the topic of the IO. Additionally, the concepts of Content Ontology formally define different kinds of instructional role such as definition, example, exercise, and so on. Furthermore, general metadata like *name*, *keyword*, *difficultyLevel*, *language*, *href* and *description* are attached to *InstructionalObject* class in content ontology through data properties to present general information about diverse IOs. The *ContentModel:SupportLS* is used to specify some features of a learner's learning style the IO is suitable for. Following OWL fragment shows an annotated instance of Instructional Object (IO):

```
<owl:NamedIndividual rdf:about="&ContentModel;IO_proper_example2">
  <rdf:type rdf:resource="&ContentModel;InstructionalObject"/>

  <ContentModel:hasDomainTopic rdf:resource="&DomainMath;ProperFraction"/>
  <ContentModel:hasIOType rdf:resource="&ContentModel;example"/>
  <ContentModel:hasLanguage rdf:resource="&ContentModel;English"/>
  <ContentModel:supportLS rdf:resource="&ContentModel;LearningStyle1"/>
  <ContentModel:supportLS rdf:resource="&ContentModel;LearningStyle2"/>

  <ContentModel:difficultyLevel rdf:datatype=&xsd:string>
    Moderate
  </ContentModel:difficultyLevel>

  <ContentModel:name rdf:datatype=&xsd:string>
    example_properFraction
  </ContentModel:name>

  <ContentModel:href rdf:datatype=&xsd:string>
    ./Math/ProperFraction/Example2.html
  </ContentModel:href>

  <ContentModel:keywords rdf:datatype=&xsd:string>
    fraction
  </ContentModel:keywords>

  <ContentModel:keywords rdf:datatype=&xsd:string>
    proper
  </ContentModel:keywords>

  <ContentModel:keywords rdf:datatype=&xsd:string>
    Denominator
  </ContentModel:keywords>

  <ContentModel:keywords rdf:datatype=&xsd:string>
    Numerator
  </ContentModel:keywords>

  <ContentModel:description rdf:datatype=&xsd:string>
    This example is about proper fraction.
  </ContentModel:description>

</owl:NamedIndividual>
```

Above code shows that *IO_proper_example2* supports two different learning styles which are defined as follows:

```
<owl:NamedIndividual rdf:about="&ContentModel;LearningStyle1">
  <rdf:type rdf:resource="&ContentModel;LearningStyle"/>
```

```

        <ContentModel:category rdf:datatype=&xsd:string>
            VARK
        </ContentModel:category>

        <ContentModel:hasType rdf:datatype=&xsd:string>
            ReadWrite
        </ContentModel:hasType>

        <ContentModel:order rdf:datatype=&xsd:integer>
            1
        </ContentModel:order>

    </owl:NamedIndividual>

    <owl:NamedIndividual rdf:about="&ContentModel;LearningStyle2">
        <rdf:type rdf:resource="&ContentModel;LearningStyle"/>

        <ContentModel:category rdf:datatype=&xsd:string>
            VARK
        </ContentModel:category>

        <ContentModel:hasType rdf:datatype=&xsd:string>
            Visual
        </ContentModel:hasType>

        <ContentModel:order rdf:datatype=&xsd:integer>
            2
        </ContentModel:order>

    </owl:NamedIndividual>

```

For instance, the above codes indicate that *IO_proper_example2* primarily supports read/write style and secondly visual style (first best fit with read/write style).

OWL enables further constraints on the relationships among classes, including cardinality. In Rule-PADEL, each IO should support minimum one learning style. The following OWL fragment shows the definition of this constraint:

```

<owl:Class rdf:about="&ContentModel;InstructionalObject">
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="&ContentModel;supportLS"/>
            <owl:onClass rdf:resource="&ContentModel;LearningStyle"/>

            <owl:minQualifiedCardinality
                rdf:datatype="&xsd;nonNegativeInteger">
                1
            </owl:minQualifiedCardinality>

        </owl:Restriction>
    </rdfs:subClassOf>
</owl:Class>

```

Above code indicates that all instances of *InstructionalObject* class must have minimum one value for *supportLS* property which is defined through *minQualifiedCardinality* property.

The ontology-based annotated IO in association with learner model enables system to dynamically generate personalised learning content for a specific learner by assembling

existing IOs.

5.4 Implementation Issues of Learner Modelling

As the learners can be described through annotations, they are represented by OWL only. After developing the learner model, it is continually updated as different aspects of learners change over the course of their learning. For example, learner's knowledge level and ability may change rapidly as the learner learns. The learner model is regularly accessed and updated during a learning session, as it is used very often by Rule-PAdel to generate personalised learning content, and personalisation decisions are made according to the information on the learner model.

As there exists a copy of the learner model in the accessible memory, the learner repository needs to be updated occasionally (e.g. when semantic rules need to be fired or when learner decides to log out of the system). This gives the system quick access to updated information, thus increasing the performance of the system considerably.

5.4.1 Ontology-based annotation of Learner

The learner model ontology presents personal preferences and learning characteristics of the learner which has interaction with the system (Yarandi et al., 2012d). The information is updated according to the learner's interactions with the content. The updated information is used by adaptation model to make adaptation decisions. Figure 5.2 depicts the graphical representation of the Learner model. The information is available for the system to adapt the learning content presentation and navigation for the learner (Hosseini et al., 2013).

The top class of learner model ontology is the User class, which points to *personalInformation*. The *personalInformation* class has some data properties to annotate learner in terms of the basic individual information such as user's name, date of birth, email, etc. so that the system can identify the user. The identification information of a particular user is presented as follows:

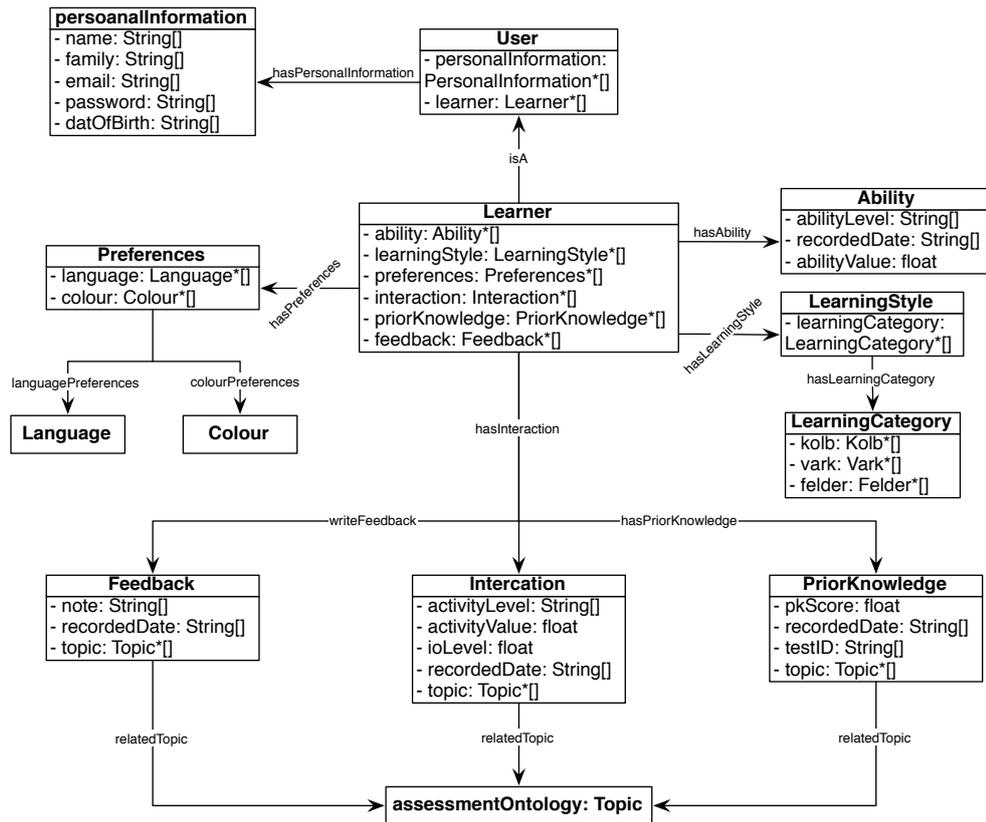


Figure 5.2: Graphical representation of the Learner model

```

<owl:NamedIndividual rdf:about="&LearnerModel;student2">
  <rdf:type rdf:resource="&LearnerModel;user"/>

  <LearnerModel:hasPersonalInformation
    rdf:resource="&LearnerModel;PI_student2"/>
</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&LearnerModel;PI_student2">
  <rdf:type rdf:resource="&LearnerModel;PersonalInformation"/>

  <LearnerModel:dateOfBirth rdf:datatype="&xsd:string">
    5-05-2005
  </LearnerModel:dateOfBirth>

  <LearnerModel:firstname rdf:datatype="&xsd:string">
    MARYAM
  </LearnerModel:firstname>

  <LearnerModel:lastname rdf:datatype="&xsd:string">
    YARANDI
  </LearnerModel:lastname>

  <LearnerModel:password rdf:datatype="&xsd:string">
    21-Mar-66
  </LearnerModel:password>

  <LearnerModel:email rdf:datatype="&xsd:string">
    yar@gmail.com
  </LearnerModel:email>

</owl:NamedIndividual>

```

OWL supports the use of some vocabularies to restrict the relation between different

concepts. For instance, a functional property states that there can be at most one individual that is related to another given individual, and the inverse functional property states that the inverse property is functional. For example, *hasPersonalInformation* is defined as a functional and inverse functional property. It means that any personal information is for exactly one user and each user is related to exactly one unit of personal information. This example is shown in the following OWL fragment:

```
<owl:ObjectProperty rdf:about="&LearnerModel;hasPersonalInformation">
  <rdf:type rdf:resource="&owl;FunctionalProperty"/>
  <rdf:type rdf:resource="&owl;InverseFunctionalProperty"/>
  <rdfs:range rdf:resource="&LearnerModel;PersonalInformation"/>
  <rdfs:domain rdf:resource="&LearnerModel;User"/>
</owl:ObjectProperty>
```

Learner class is subclass of the *User* to represent details about learners. The *Learner* class is a central concept as it includes all the properties of a learner (Yarandi et al., 2012c). It comprises learner's ability, preference, prior knowledge, feedback, learning style and Interaction. These five characteristics are defined through exactly five classes which are linked to the learner class through *hasAbility*, *hasPreferences*, *hasPriorKnowledge*, *writeFeedback*, *hasLearningStyle* and *hasIntarction* properties. These aspects are considered important to describe the learning characteristics of the learner (Hosseini et al., 2013). Following OWL fragment shows an instance of *Learner* class and its object properties:

```
<owl:NamedIndividual rdf:about="&LearnerModel;student2">
  <rdf:type rdf:resource="&LearnerModel;Learner"/>

  <LearnerModel:hasAbility rdf:resource="&LearnerModel;Ab_student2"/>

  <LearnerModel:hasPreferences rdf:resource="&LearnerModel;PF_student2"/>

  <LearnerModel:hasPriorKnowledge rdf:resource="&LearnerModel;PK_student2"/>

  <LearnerModel:writeFeedback rdf:resource="&LearnerModel;FB_student2"/>

  <LearnerModel:hasLearningStyle rdf:resource="&LearnerModel;LS_student2"/>

  <LearnerModel:hasIntract rdf:resource="&LearnerModel;IN_student2"/>

</owl:NamedIndividual>
```

Ab_student2, *PF_student2*, *PK_student2*, *FB_student2*, *LS_student2* and *IN_student2* are instances of *Ability*, *Preferences*, *PriorKnowledge*, *Feedback*, *LearningStyle* and *Interaction* class respectively. In what follows, we describe the details of these classes.

The *Ability* class presents the ability of the learner in each level of learning. In order to estimate the learner's ability, some regular exams are taken by the learner at different steps of the learning process. The results of these exams are analysed according to Item

response Theory to obtain the learner's ability. Explicitly, the ability level and the date when the ability is recorded are presented via *abilityLevel* and *recordedDate* properties. The learner's abilities in different stages of the learning process are kept in the instances of *Ability* class to obtain the fluctuation of ability during the learning process (Hosseini et al., 2013). The following examples show two instances of *Ability* class. It shows that the learner's ability was Moderate in 6th of April 2012 and is changed to High in 7th of June 2012.

```
<NamedIndividual rdf:about="&LearnerModel;Ab_student2">
  <rdf:type rdf:resource="LearnerModel\#Ability"/>

  <LearnerModel:abilityLevel rdf:datatype="xsd:string">
    Moderate
  </LearnerModel:abilityLevel>

  <LearnerModel:recordedDate rdf:datatype="xsd:string">
    2012-04-06
  </LearnerModel:recordedDate>

</owl:NamedIndividual>

<NamedIndividual rdf:about="&LearnerModel;Ab_student2">
  <rdf:type rdf:resource="LearnerModel\#Ability"/>

  <LearnerModel:abilityLevel rdf:datatype="xsd:string">
    High
  </LearnerModel:abilityLevel>

  <LearnerModel:recordedDate rdf:datatype="xsd:string">
    2012-06-07
  </LearnerModel:recordedDate>

</owl:NamedIndividual>
```

The *Preferences* class presents the preferences of learner regarding colour, and language. These two features are used to provide adaptive presentation. An instance of *Preferences* class is shown in the following example:

```
<owl:NamedIndividual rdf:about="&LearnerModel;Preferences_student2">
  <rdf:type rdf:resource="&LearnerModel;Preferences"/>

  <LearnerModel:LanguagePreference rdf:resource="&LearnerModel;English"/>

  <LearnerModel:ColourPreference rdf:resource="&LearnerModel;Green"/>

</owl:NamedIndividual>
```

The example shows that the language preference of student2 is “English” and also it shows that her colour preference is “Green”.

Furthermore, each learner has a set of prior knowledge related data which is presented in *PriorKnowledge* class via *hasPriorKnowledge* property. This class contains information about the learner's background knowledge and gained knowledge from previous steps of the

learning process via this system (Yarandi et al., 2012d). Gained knowledge can be obtained as a result of assessments which are taken from individual learners. *PriorKnowledge* class has following data properties for recording the learner's knowledge(Hosseini et al., 2013):

1. The *relatedTopic* property refers to the topic of the domain ontology that describes the topic of learner's acquired knowledge
2. The *PKScore* property represents the percentage score which is calculated based on the learner's response to the presented test.
3. The *recordedDate* property keeps the date when the learner completed the test.
4. The *testId* property refers to the identification of completed test by the learner. If the learner needs to repeat this topic, this property prevents presenting the same test repeatedly.

The instance of this class can be taken as measures of the learner's prior knowledge. The example of learner's prior knowledge can look as follows.

```
<owl:NamedIndividual rdf:about="&LearnerModel;student2">
  <rdf:type rdf:resource="&LearnerModel;Learner"/>

  <LearnerModel:hasPriorKnowledge rdf:resource=
    "&LearnerModel;PK_student2_AddFraction"/>
</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&LearnerModel;PK_student2_AddFraction">
  <rdf:type rdf:resource="&LearnerModel;PriorKnowledge"/>

  <LearnerModel:PKScore rdf:datatype="&xsd;float">
    68.0
  </LearnerModel:PKScore>

  <LearnerModel:recordedDate rdf:datatype="&xsd:string">
    2012-12-12
  </LearnerModel:recordedDate>

  <LearnerModel:testId rdf:datatype="&xsd:string">
    Test_125
  </LearnerModel:testId>

  <LearnerModel:relatedTopic rdf:resource="&DomainMath;AddFraction"/>
</owl:NamedIndividual>
```

The *PK_student2_AddFraction* is an instance of *PriorKnowledge* class which keeps the Prior Knowledge of learner student2 about “add fraction” topic.

The interactions with the Rule-PAdel at run time can be used to draw conclusions about possible learners' tasks and progress. These conclusions can be used later for updating prior knowledge and also provide personalisation. Therefore, *Interaction* class should

provide a structure of information about possible learner interaction. Interaction is based on actions taken by a particular learner, during learning process. It implies a topic learned from the experience, which is represented by *topicUsed* property. Interaction has a certain value for *activityLevel* (e.g. Easy, Moderate, Difficult). It also implies the level of IO recommended to the learner using *IOLevel* property. An instance of this class is shown as follows:

```
<owl:NamedIndividual rdf:about="&LearnerModel;IN_student2">
  <rdf:type rdf:resource="&LearnerModel;PriorKnowledge"/>

  <LearnerModel:activityLevel rdf:datatype="&xsd:string">
    difficult
  </LearnerModel:activityLevel>

  <LearnerModel:recordedDate rdf:datatype="&xsd:string">
    2012-12-15
  </LearnerModel:recordedDate>

  <LearnerModel:topicUsed rdf:resource="&DomainMath;SubtractFraction"/>

  <LearnerModel:IOLevel rdf:datatype="&xsd:string">
    Moderate
  </LearnerModel:IOLevel>

</owl:NamedIndividual>
```

Moreover, the learner's feedback is recorded in *Feedback* class via *writeFeedback* Property. This class is associated with the three following properties to represent the learner's feedback:

1. The *relatedTopic* property refers to the topic which the feedback is about.
2. The *note* property represents feedback of learner about the mentioned topic.
3. The *recordedDate* keeps the date when the feedback is recorded.

Finally, the *learningStyle* class holds information about the learner's learning style (*hasLearningStyle*) which is associated with the *LearningStyleCategory* class. There are different identifiable approaches to describe learning style preferences. *LearningStyleCategory* class represents three different learning models namely Kolb (Kolb, 1984), Felder-Silverman (Felder & Silverman, 1988) and VARK learning model (Fleming & Mills, 1992). The instructional designers can add new learning style models to this class.

Kolb's learning theory organises four distinct learning styles namely Activist, Theorist, Pragmatist and Reflector. Learners generally prefer one of the four styles above the others. The learner's learning preference is determined based on the result of a related questionnaire. It is represented in *Kolb* class, which is defined as a subclass of the

LearningStyleCategory class.

The Felder-Silverman model rates the learner's learning style in a scale of four dimensions: Active and Reflective, Sensing and Intuitive, Visual and Verbal, Sequential and Global. The *FelderSilverman* class, which is a subclass of the *LearningStyleCategory* class, presents these dimensions through their related subclasses (Yarandi et al., 2012c).

The VARK Learning model influences the nature and form of the delivered learning material. The acronym VARK stands for Visual, Aural, Read/write and Kinaesthetic sensory modalities that are used for learning information. The *Vark* class is defined as a subclass of the *LearningStyleCategory* class which represents these categories. Each of the learning style models has a related questionnaire which assesses variations in individual learning style preferences. The proposed ontology enables instructional designers to determine learner's learning style based on different learning models. Consequently, personalisation is achievable in different ways. For example, if the designer selects VARK model, the content will be personalised based on the type of delivered content. However, if she selects FelderSilverman model, this will affect the sequence or structure of learning content (Yarandi et al., 2012d).

Before initial learning process, and after learning style has been determined, current learning style category of the specific learner is stored in Learner ontology. For example, if instructional designer selects VARK model and the result of questionnaire should be stored in ontology. The following shows the learning preferences of a particular learner (student2):

```
<owl:NamedIndividual rdf:about="&LS_student2">
  <rdf:type rdf:resource="&LearnerModel;LearningStyle"/>
  <hasLearningCategory rdf:resource="&A_student2"/>
  <hasLearningCategory rdf:resource="&K_student2"/>
  <hasLearningCategory rdf:resource="&RW_student2"/>
  <hasLearningCategory rdf:resource="&V_student2"/>
</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&RW_student2">
  <rdf:type rdf:resource="&LearnerModel;ReadWrite"/>

  <learningCategoryValue rdf:datatype="&xsd;float">
    56.25
  </learningCategoryValue>

  <learningCategoryRanking rdf:datatype="&xsd;string">
    1
  </learningCategoryRanking>

</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&V_student2">
  <rdf:type rdf:resource="&LearnerModel;Visual"/>

  <learningCategoryValue rdf:datatype="&xsd;float">
```

```

        18.75
    </learningCategoryValue>

    <learningCategoryRanking rdf:datatype="&xsd:string">
        2
    </learningCategoryRanking>

</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&K_student2">
    <rdf:type rdf:resource="&LearnerModel;Kinaesthetic"/>

    <learningCategoryValue rdf:datatype="&xsd;float">
        15.25
    </learningCategoryValue>

    <learningCategoryRanking rdf:datatype="&xsd:string">
        3
    </learningCategoryRanking>

</owl:NamedIndividual>

<owl:NamedIndividual rdf:about="&A_student2">
    <rdf:type rdf:resource="&LearnerModel;Audio"/>

    <learningCategoryValue rdf:datatype="&xsd;float">
        9.75
    </learningCategoryValue>

    <learningCategoryRanking rdf:datatype="&xsd:string">
        4
    </learningCategoryRanking>

</owl:NamedIndividual>

```

The property *learningCategoryValue* is defined to represent the percentage score of a specific learner in each category and the *learningCategoryRanking* presents the order of different aspects of VARK for the learner. The *Visual*, *Audio*, *ReadWrite* and *Kinaesthetic* classes are defined as subclasses of *VARKLearningCategory* class.

5.4.2 Initialising Learner Model

Before Rule-PA_{del} can generate a personalised learning path for an individual learner it should have certain information about her. This information is acquired by presenting online instruments to the learner and getting learner's responses. Generally, the online instruments are a prior knowledge questionnaire to determine learners' prior knowledge and an online questionnaire to find out what their learning styles is (e.g a VARK questionnaire).

The learner mediator uses the learner's response to initialise the learner model. For example, if the system determines that a learner knows already about "Add Fraction" then an instance of *PriorKnowledge* class is created and initialised and also a link (property) is created between this instance and the specific learner.

In the first session, the learner completes the registration form which can also be accessed and modified later during their learning process. Through this process, the learners can change their profile and therefore new personalised learning content will be given to them.

5.4.3 Updating and Maintaining the Learner Model

During learning process, learners visit different learning content and perform various tasks. When the learner completes the sequence of learning contents, the Rule-PAdel system evaluates the learner's acquired knowledge. The learner modeller is responsible for maintaining and updating the learner model based on this new information. When the learner is participating in personalised learning, the OWL API may access the learner ontology. OWL API provides a number of facilities to offer updating, modifying and removing information in the model. In this way, we define the *insertLearnerToOWL*, *removeLearnerFromOWL*, *changingLearnerInOWL* and *findingLearnerInOWL* methods to provide this functionality. These methods are defined using the generalised methods of OWL API for accessing the ontology.

The learner modeller is also responsible for modifying and updating the copy of the learner model which is in the memory. Different methods such as *insertLearner*, *removeLearner* and *findingLearner* are defined to enable these.

5.5 Adaptation Model

Adaptation Model in the semantic rule-based approach contains a semantic rule set required for adaptation. The rule set offers personalised learning with respect to the learner model, and configures different parameters including the topic sequencing, the adaptive features of the learning content and adaptive guidance. This section describes the semantic rules that describe adaptation.

5.5.1 SWRL Rules for Adaptation

Our approach focuses on inference rules as a means of providing efficient and effective personalised learning. SWRL (Horrocks et al., 2004) was chosen to represent the adaptation decisions in Rule-PAdel for the following reasons:

1. SWRL can be integrated with OWL. SWRL rules work directly with the concepts and relationships defined in the OWL model. Therefore, adaptation would be explicitly represented in the ontology.
2. SWRL supports Hornlike rules expressed in terms of OWL concepts to reason about OWL individuals.
3. SWRL rules could be easily viewed and edited and reasoned by the instructional designer.
4. SWRL is a comprehensive, expressive yet simple language.
5. SWRL can be used in Java, thus it can be integrated with other Java-based components of Rule-PAdel.
6. SWRL is a declarative language, therefore bearing all the benefits of declarative paradigm. For example it allows the instructional designers to define adaptation decisions without going into much technical detail.

These features offer great flexibility to the instructional designer when creating adaptation models. In our approach, the SWRL rules describing the adaptation decisions are used to build adaptive structure of a course, to select appropriate IOs or to provide suitable guidance based on learner's interaction with the system. In what follows, different types of adaptation rules implemented in Rule-PAdel are presented. However, our approach enables instructional designers to add new rules, modify or remove existing rules in order to change the adaptation decisions.

5.5.2 Adaptive Course Structure

Adaptive course structure is a kind of adaptive navigational support which informs learners of the most suitable topics to learn. In other words, it helps learners to find an optimal path

through different topics based on their current knowledge. In our approach, this facility is provided by adaptively annotating and dimming the links to make easier the choice of the next link (more detailed in Section 3.3.2). Therefore, an Annotated Course Structure (ACS) is built to adapt the structure of a course to the learner's knowledge. Each item in the ACS represents a topic and is also linked to the learning content about this topic. The content behind each item in ACS, as we defined it, may have three educational states:

- Learned
- Ready to be learned
- Not ready to be learned

According to these three states, the item is shown in different colours to inform learners about the educational state of content behind the link. The primary information used to describe an annotated course structure is the learner's knowledge specified by learner ontology and the way topics in a course are structured from a pedagogical perspective available from the domain ontology. Thus, the following two SWRL rules are defined to recognise the learner's level of knowledge:

```
Topic(?t), Learner(?x), PriorKnowledge(?p), hasPriorKnowledge(?x, ?p),
relatedTopic(?p, ?t), pkScore(?p, ?v), greaterThanOrEqual(?v, 50.0) → knows(?x,
?t)

Topic(?t), Learner(?x), PriorKnowledge(?p), hasPriorKnowledge(?x, ?p),
relatedTopic(?p, ?t), pkScore(?p, ?v), lessThan(?v, 50.0) → notKnow (?x, ?t)
```

The informal meaning of the above rules is: if a learner passes the assessment about a specific topic (?t), the first rule implies that the learner knows this topic, otherwise the second rule implies that she does not know it. According to the above two rules and also the prerequisite relationship defined between different topics in domain ontology the three educational states can be implied with the following rules:

```
Topic(?t), Learner(?x), knows(?x, ?t) → learned (?x, ?t)

Topic(?t), Topic(?t1), Learner(?x), knows(?x, ?t), isPrerequisiteFor(?t, ?t1) →
readyToLearn(?x, ?t1)

Topic(?t), Topic(?t1), Learner(?x), notKnows(?x, ?t), isPrerequisiteFor(?t, ?t1)
→ notReadyToLearn(?x, ?t1)
```

Based on the results of the above rules, the system presents the ACS items in different colours to support learners to find their optimal learning path.

5.5.3 Adaptive presentation of Learning Content

The goal of adaptive presentation is to customise content to match with characteristics available in the learner model. Therefore, the learning content is not static, but adaptively generated or assembled from existing IOs for each learner. For example, with several adaptive presentation techniques, learners with high ability receive advanced IO (with more detailed and deep information), while learners with low ability receive a basic IO (with more additional explanation). However, the instructional designer can add new SWRL rules to adaptation model in order to adapt content to individual learners according to different criteria such as the learner's learning characteristics and needs. We define some example rules to adapt learning content to an individual learner based on learner's ability, learning style, language and also instructional plan.

Ability Level

Learning content that is too easy or difficult makes the learner dissatisfied and decreases the performance of learning. Therefore, considering learner ability and the difficulties of learning contents simultaneously can improve personalised learning performance. As we mentioned in Chapter 2, to obtain more precise estimation of learner's ability, the learner's responses to presented tests are analysed according to item response theory in order to estimate the learner's ability. The ability value usually is defined as a floating point number and should be between -3 and 3 with zero representing a moderate ability and the larger the number the higher the ability. Correspondingly, the difficulty value of each IO follows the same rules as the ability value, being a floating point number and being restricted between -3 and 3. The following rules find IOs with their difficulty level matched with the learner's ability level.

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,  
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqual(?s, -0.5),  
lessThan(?s, 0.5), subtract(?s, ?v1, ?v2) → hasAbilityToLearn(?x, ?y)
```

The above rule compares the difficulty value of each IO and the ability level of a learner. If the difference between these two values is less than 0.5 then it means that the learner's ability matches the difficulty level of the IO. As a result of firing the rule, the system determines that the learner has ability to learn this IO.

In the process of recommending an interactive IO, the system also considers the level of the previous interactive IO to which the learner has correctly responded. In order to be more accurate, a floating point number between -3 and 3 is defined to present the level of activity. This value is represented in the *activityValue* property. For example, if the learner correctly interacts with an activity with difficulty value v_2 within a specific topic, then following the successful completion of this activity the system recommends another interactive IO (activity), within the same topic, with the difficulty value being $v_2 + 1$ (i.e. one level harder than the previous one). The following rule formalises the above mentioned situation:

```
InstructionalObject(?y), InteractiveIO(?type), Learner(?x), Interaction(?p),
hasDomainTopic(?y, ?d), hasIOType(?y, ?type), hasInteraction(?x, ?p),
relatedTopic(?p, ?d), difficultyValue(?y, ?v1), activityValue(?p, ?v2),
greaterThanOrEqualTo(?s, -0.5), lessThan(?s, 0.5), subtract(?s, ?v1, ?v2) →
hasAbilityToDo (?x, ?y)
```

Several SWRL rules are defined, in the case that no IO with matched difficulty level is found; the system will attempt to find the IOs with difficulty level being one level lower or higher than the learner's ability. If no such IO is found, then the tolerance is increased to two and so forth. The following rule is one of these rules (see Appendix C for similar SWRL rules):

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqualTo(?s, 0.5),
lessThan(?s, 1.5), subtract(?s, ?v1, ?v2) → hasOneLevelLowerAbility(?x, ?y)
```

In the above rule, if the difference between the difficulty level of an IO and the learner's ability is between 0.5 and 1.5, the learner is one level under qualified to learn the specific IO.

Use of Non-numeric terminology

The ability value, difficulty value and activity value which are defined in the previous section are mapped to mathematical values in order to enable computation and therefore allow the system to perform mathematical calculations. However, the non-numeric terms are often used to facilitate the expression of facts and offer the students an intuitive understanding. The following rules represent the terms *very low*, *low*, *moderate*, *high*, *very high* which are the terms used to represent the ability value:

```
Ability(?a), abilityValue(?a, ?v1), lessThan(?v1, -1.5) → AbilityLevel(?a, VeryLow)

Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqualTo(?v1, -1.5), lessThan(?v1, -0.5) → AbilityLevel(?a, Low)

Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqualTo(?v1, -0.5), lessThan(?v1, 0.5) → AbilityLevel(?a, Moderate)

Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqualTo(?v1, 0.5), lessThan(?v1, 1.5) → AbilityLevel(?a, High)

Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqualTo(?v1, 1.5) → AbilityLevel(?a, VeryHigh)
```

The following rules represent the terms *basic*, *primary*, *intermediate*, *upper intermediate*, *advanced* which are the terms used to represent the difficulty value:

```
InstructionalObject(?y), difficultyValue(?y, ?v1), lessThan(?v1, -1.5) → DifficultyLevel(?y, Basic)

InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqualTo(?v1, -1.5), lessThan(?v1, -0.5) → DifficultyLevel(?y, Primary)

InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqualTo(?v1, -0.5), lessThan(?v1, 0.5) → DifficultyLevel(?y, Intermediate)

InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqualTo(?v1, 0.5), lessThan(?v1, 1.5) → DifficultyLevel(?y, UpperIntermediate)

InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqualTo(?v1, 1.5) → DifficultyLevel(?y, Advanced)
```

The following rules represent the terms *very easy*, *easy*, *moderate*, *difficult*, *very difficult* which are the terms used to represent the difficulty value of an activity:

```
Interaction(?y), activityValue(?y, ?v1), lessThan(?v1, -1.5) → activityLevel(?y, VeryEasy)

Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqualTo(?v1, -1.5), lessThan(?v1, -0.5) → activityLevel(?y, Easy)

Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqualTo(?v1, -0.5), lessThan(?v1, 0.5) → activityLevel(?y, Moderate)

Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqualTo(?v1, 0.5), lessThan(?v1, 1.5) → activityLevel(?y, Difficult)

Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqualTo(?v1, 1.5) → activityLevel(?y, VeryDifficult)
```

Learning Style

People learn differently, thus the idea of considering individual differences are valuable in order to improve learning outcome. There is a wide range of learning style models for classifying learners (see Section 5.4) and instructional designers are responsible to select

one of them. They can add appropriate rules to an adaptation model in order to implement different preferences and learning styles. This is significant for a personalised learning system as it enables the instructional designers to provide different adaptive effects based on different sets of models. In Rule-PAdel we defined several rules to adapt learning content based on VARK. For example, as a result of firing the following rules different IOs, which are suitable for a specific learner according to VARK learning style, are linked to the learner through *hasMatchedLS* property.

```
Learner(?x), InstructionalObject(?y), LearningStyle(?s), Visual(?k), supports(?y,
Visual), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s),
learningCategoryRanking(?k, "1") → LSIsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Audio(?k), supports(?y,
Audio), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s),
learningCategoryRanking(?k, "1") → LSIsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), ReadWrite(?k),
supports(?y,Read/Write), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s),
learningCategoryRanking(?k, "1") → LSIsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Kinaesthetic(?k),
supports(?y,Kinaesthetic), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s),
learningCategoryRanking(?k, "1") → LSIsSupportedWith(?x, ?y)
```

As we mentioned in Section 5.4.1, *learningCategoryRanking* represents the order of different aspects of VARK learning style for a specific learner. For example, the first rule informally means: if a specific IO supports *visual* learners and the first preference of learner *x* is *visual* then this IO is suitable for this learner.

Several rules were defined in Rule-PAdel to find IOs that support second, third and fourth learning preferences of a specific learner. The system utilises the result of these rules when an IO does not support the learner's first preference. The following rule represents one of these situations.

```
Learner(?x), InstructionalObject(?y), LearningStyle(?s), Visual(?k), supports(?y,
Visual), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s),
learningCategoryRanking(?k, "2") → LS2IsSupportedWith(?x, ?y)
```

As a result of the above rule, IOs which support visual learning style as learner's second preference are linked to the learner through *hasLS_MF2With* property. Similar SWRL rules for other styles and other order can be found in Appendix C.

Recommender Rule

The goal of adaptive presentation is adapting learning content to an individual. In our approach, while the instructional designer is able to change the aspects that the system may adapt towards, the system's rules adapt the content for a learner based on her learning style, ability, preferred language and instructional plan (i.e. Instructional plan decides on the tasks the learner should do and the order in which they should appear). The following SWRL rules provide the most suitable IOs, static and/or interactive, for a particular learner based on the points mentioned:

```
Learner(?x), InstructionalObject(?y), StaticIO(?t), Language(?g),
hasAbilityToLearn(?x, ?y), LSIsSupportedWith(?x, ?y), nextIOType(?x, ?t),
selectedTopic(?x, ?d), hasDomainTopic(?y, ?d), hasIOType(?y, ?t),
hasLanguagePreference(?x, ?g), isInLanguage(?y, ?g) → isRecommendedStaticIO(?x, ?y)

Learner(?x), InstructionalObject(?y), InteractiveIO(?t), Language(?g),
hasAbilityToDo(?x, ?y), LSIsSupportedWith(?x, ?y), nextIOType(?x, ?t),
selectedTopic(?x, ?d), hasDomainTopic(?y, ?d), hasIOType(?y, ?t),
hasLanguagePreference(?x, ?g), isInLanguage(?y, ?g) →
isRecommendedInteractiveIO(?x, ?y)
```

The above rules take into account the following criteria when selecting each IO for recommendation to a specific learner:

1. The next learning task (e.g. explanation, example, exercise, test) which the learner is recommended to do: *nextIOType* stores the instructional role of the next IO which should be presented to the learner based on the instructional plan and the current state of the learner. For example, if an instructional plan defines that the learner should be presented an *example* and then an *exercise*, and the *example* is presented to the learner, then the *nextIOType* for this learner would be *exercise*. Therefore, the system finds the IOs with *exercise* as their instructional role, by using *hasIOType* property.
2. The learner ability should be matched with the difficulties level of the IO: *hasAbilityToLearn* and *hasAbilityToDo* properties represent this match for static IOs and interactive IOs respectively. These two properties themselves are the result of other rules which were detailed in this section (in Subsection 5.5.3).
3. Learner's learning style: the recommended IO should support the learner's learning style which is represented with *LSIsSupportedWith* property. This property is also the result of other rules, described in this section (in Subsection 5.5.3).

4. Learner's language preferences: the IO should be in the language that the learner prefers.

Once these conditions, which the personalisation is based upon, are satisfied the rules will find the IOs which are suitable for the particular learner. The static and interactive IOs are recommended by *isRecommendedStaticIO* and *isRecommendedInteractiveIO* properties respectively. Similar rules are defined to find IOs which partially match the above criteria, in the cases where no fully matched IO can be found (see Appendix C)

5.5.4 Adaptive guidance

Adaptive guidance prevents learners from information overload by offering them personalised advices based on learner's interaction with the system. Ideally, they should help learners in finding a learning path that perfectly matches their profile and also enables them to complete their learning in an effective and efficient way. Moreover, when learners increase their knowledge after successfully completing a learning activity, a new matching process for the updated learner model and learner's current learning situation will be needed. In our approach, adaptive guidance uses 'pedagogical rules' which consider the learner's responses to interactive IOs (which can range from a simple exercise to a complex test) and the current situation of the learning process to offer personalised advice to the learners about which action to do or which content to study in the next step. They also provide the learners with supplementary content to remediate the learner's difficulties (e.g. more examples or more exercises). Rule-PAdel allows instructional designers to define different adaptive guidance based on diverse pedagogical rules. Currently, different SWRL rules are defined to infer adaptive guidance for the learner such as recommending them to learn a new topic, repeat the current topic with more detail, read more examples, do more activities with lower or higher difficulty levels and repeat prerequisite topics. These recommendations are defined according to the learner's progress and the relative status of the current content.

For instance, if a learner responds to an interactive IO (e.g. an exercise) with moderate difficulty level incorrectly, she is recommended to read a few examples to remediate her problems. If after reading the examples the learner again fails to respond correctly to the IO, the rule would then recommend the learner to read the learning content associated with

the IO in a lower difficulty level (i.e. with more additional explanations). If the learner has already read the learning content in its most basic form, she is recommended to repeat the prerequisite topic. The following rules implement the above situation in SWRL:

```
Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityLevel(?p, Moderate) →
isGuided(?x, moreExample)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityLevel(?p, Moderate),
isGuided(?x, moreExample), IOLevel(?p, ?v1), lessThan(?v1, 0) → isGuided(?x,
repeatPrerequisite)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityLevel(?p, Moderate),
isGuided(?x, moreExample), IOLevel(?p, ?v1), greaterThanOrEqual(?v1, 0) →
isGuided(?x, repeatWithLowerLevel)
```

The above rules are about a learner who fails in answering an interactive IO with moderate level of difficulty in different situations. The rules infer different paths of guidance for the learner based on her current situation and the learning content presented to her.

We also defined rules for when the learner fails in answering interactive IOs having difficulty level of more than moderate (e.g. difficult or very difficult). In this case, the learner is guided to do more interactive IOs with the same difficulty level. If she fails to respond correctly to them again, she is recommended to read a learning content with a higher difficulty level (with more detailed and deeper information). The following rules define these two situations:

```
Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
?v1), greaterThanOrEqual(?v1, 1) → isGuided(?x, moreActivity)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
?v1), greaterThanOrEqual(?v1, 1) isGuided(?x, moreActivity) → isGuided(?x,
learnWithHigherLevel)
```

All the above rules consider the situation where the learner fails to answer to an interactive IO correctly. If the learner responds correctly to an interactive IO with moderate difficulty or lower, then the system will advise her to do another interactive IO with one more level of difficulty. On the other hand, if she responds correctly and the level of the IO is higher than moderate (difficult or very difficult), she is recommended to participate in an assessment for evaluation of her current knowledge. The following SWRL rules represent these formally:

```
Learner(?x), responseToIO(?x, true), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
?v1), lessThanOrEqual(?v1, 0) → isGuided(?x, nextLevel)
```

```
Learner(?x), responseToIO(?x, true), Interaction(?p), hasInteraction(?x,
?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
?v1), greaterThanOrEqual(?v1, 1) → isGuided(?x, assessment)
```

There are more SWRL rules to guide learners based on their interaction with the system which are detailed in Appendix C.

5.6 Rule-based Adaptation Process

Adaptation process in Rule-PAdel provides adaptive navigation, adaptive presentation and adaptive guidance for learners using ontological modelling associated with rule-based reasoning mechanisms. This approach supports course authors and instructional designers to develop different personalised learning using diverse pedagogical approaches across different domains, while it prevents them from involving in technical tasks.

In Rule-PAdel, Pellet is the reasoner and also the rule engine that processes the ontological models and adaptation rule sets (Sirin et al., 2007). Pellet is an open source rule engine that offers sound and complete OWL reasoning. It has extensive support for reasoning with individuals and also SWRL rules. It is accessible through various interfaces including the OWL API interface which we have used for interacting with ontologies and has proven to be a reliable tool for working with OWL ontologies (Sirin et al., 2007).

After firing rules, the inferred knowledge is stored back to the ontology repository and therefore the knowledge base would be updated. It is based on this updated knowledge that Rule-PAdel provides adaptive learning for learners. By adaptation we mean adaptive course structure, adaptive content and adaptive guidance. These are provided through course structure constructor, recommender and guide components, which are core components of the architecture described in Section 5.2 The details of these components will be described in this section.

5.6.1 Course Structure Constructor

The responsibility of the course structure constructor is to provide annotated course structures for learners to help them in finding an optimal learning path based on their current knowledge. When SWRL rules are executed after providing the factual knowledge using Pellet, the course structure constructor gets the inferred information about the learner's knowledge and constructs the annotated course structure accordingly. The following syntax is used for loading adaptation ontology through OWL API and creating a reasoner using Pellet:

```
AdaptationOntology = manag.loadOntologyFromOntologyDocument(file);
reasoner = PelletReasonerFactory.getInstance()
.createReasoner(AdaptationOntology);
```

The component sets different colours for different topics based on the learner's current knowledge. The following algorithm shows this process:

```
courseStructureConstructor (Learner student1)
{
    learnedTopics = reasoner.getObjectPropertyValues(student1, learned);
    readyToLearnTopics = reasoner.getObjectPropertyValues(student1, readyToLearn);
    notReadyToLearnTopics =
    reasoner.getObjectPropertyValues(student1, notReadyToLearn);
    domainTopics = domainMediator.tree.traverse();

    foreach ( learnedTopic : learnedTopics )
    {
        Topic = Find( leanedTopic, domainTopics );
        Topic.setColor(Purple);
    }

    foreach ( readyToLearnTopic : readyToLearnTopics )
    {
        Topic=Find( readyToLearnTopic , domainTopics);
        Topic.setColor(Blue);
    }

    foreach ( notReadyToLearnTopic : notReadyToLearnTopics )
    {
        Topic = Find( notReadyToLearnTopic , domainTopics);
        Topic.setColor(Gray);
    }
}
```

The algorithm shows that the constructor receives the inferred information about the learner's current knowledge through *learned*, *readyToLearn* and *NotReadyToLearn* properties (more details in Section 5.5). It also gets the structure of a course in the form of a tree which shows the topics and the relations between them from the domain mediator (see Section 5.2 for more details). After that, it sets different colours for each topic based on the acquired information from the adaptation model.

5.6.2 Recommender

Using semantic rules, appropriate IOs may be selected for a particular learner at runtime based on her profile, available in the learner ontology. Recommender is responsible to assemble selected IOs to generate personalised learning content. The following algorithm shows this process:

```
Recommender(Learner student1)
{
  // Get the instructional plan from the content ontology and put it in the
  NextIOType stack
  Student1.stackNextIOType = getInstructionalPlan(Learner student1);

  // Delete the previous selected topic, get the new one from topic stack and add
  it to the adaptation ontology
  deleteAxiom (student1, selectedTopic);
  topic = student.popStackTopic();
  addAxiom (student1, selectedTopic, topic);

  while (!student1.isEmptyStackNextIOType())
  {
    // Get the current IO type from the stack, delete the previous one and add
    the new one to the adaptation ontology
    IOType = student1.popNextIOType();
    deleteAxiom (student1, nextIOType)
    addAxiom (student1, nextIOType, IOType);

    //refresh the reasoner to insert inferred knowledge to adaptation ontology
    reasoner.refresh();

    //Get th static and dynamic recommended IO from the adaptation ontology
    rcmdStaticIOs =
    reasoner.getObjectPropertyValues(Learner, isRecommendedStaticIO);
    rcmdInteractiveIOs = reasoner.getObjectPropertyValues(Learner,
    isRecommendedInteractiveIO);
    // Add the appropriate IO to the learning content
    assembler(rcmdStaticIOs , rcmdInteractiveIOs , IOType , student1);
  }
}
```

In the first step the *getInstructionalPlan* function gets the instructional plan from the content model and puts it in a stack called *nextIOType*. The instructional plan may include the tasks the learner should do and the order in which they should appear. This information is useful to teach the same topic in different instructional plans. In this case the instructional plan in the content model can be used for selecting and sequencing the IOs. In other words, it usually influences the way the learning content is structured. For instance, suppose there are two equivalent instructional plans: one shows basic definitions and then presents three examples and finally a puzzle; another might present an example followed by a definition and lastly an exercise. The two learning contents may teach the same topic, but in different ways. In this case the two instructional plans are defined in the content model.

After getting the instructional plan, selectedTopic is updated in the ontology using the two functions *deleteAxiom* and *addAxiom* to delete the previous axioms and add the new ones respectively. The same process is done with the IO types.

After updating the ontology, the reasoner (i.e. pellet) should be called to infer new information because of the changes in the knowledge base. Recommended static and interactive IOs are identified as the result of firing the rules. The assembler selects one IO which is not shown to the learner previously. It is called after the ontology is updated in iterations, each time adding the selected IO to the learning content. Therefore, once the process is completed, the learning content is constructed. The following code shows this process:

```
assembler(Individual rcmdStaticIOs ,Individual rcmdInteractiveIOs,string IOType ,
Learner student1)
{
    //Check whether the next recommendation is interactive or static and select
    the next IO from the related recommendations list
    if (IOType == "interactiveIO")
    {
        nextRecommendations = rcmdInteractiveIOs ;
    }
    else
    {
        nextRecommendations = rcmdStaticIOs ;
    }
    // Add an IO not shown to the learner previously to the learning content
    findIO=false;
    rcmdIO = GetFirstIO(nextRecommendations);
    while (!findIO)
    {
        if (!student1.findIO(rcmdIO,student1.IOList ))
        {
            // createIONode put rcmdIO and all its features on IONode (rcmdIO is
            only the IO's ID)
            IONode = createIONode(rcmdIO);
            student1.addIOList(IONode);
            student1.addLearningContent(IONode)
            findIO=true;
        }
        else
        {
            rcmdIO = GetNextIO(rcmdIO , nextRecommendations);
        }
    }
}
```

5.6.3 Guide

Guide in Rule-PAdel is a component which configures the system to prepare it for the execution of the guidance recommended by the adaptation model. The following algorithm partially shows this configuration:

```

guide(Student student)
{
    guide = reasoner.getObjectPropertyValues(Learner, isGuided);
    response=Rendering (guide);
    if (response)
    {
        if (guide.equal("NextLevel"))
        {
            IOL = student.getInteraction().getActivityValue();
            if (IOL < 2)
            {
                student.getInteraction().setActivityValue(IOL + 1);
                student.changeActivityValue();
            }
            student.pushNextIOType(IO.getIOType());
        }
        else if (guide.equal("MoreExample"))
        {
            student.pushNextIOType(IO.getIOType());
            student.pushNextIOType(ContentBase + "example");
        }
        else if (guide.equal("RepeatWithLowerLevel"))
        {
            Ab = student.getAbility().getAbilityValue() - 1;
            student.setAbility.setAbilityValue(Ab);
            student.changeAbility(t);
            student.clearStackNextIOType();
            getLearningPlan(student);
        }
        else if (guide.equal("assessment"))
        {
            student.pushNextIOType(ContentBase + "assessment");
        }
        .....
    }
}

```

The above algorithm shows that if the following guidance is recommended to the user, Guide configures the system's settings accordingly. These settings can be found in Table 5.1.

Recommended Guidance	Guide's reaction
Next level	The difficulty level of the interactive IO is incremented by one. Therefore, the system presents an interactive IO with the same type but one level more difficult than the current one in the next step.
More examples	The type of the current IO (e.g. exercise) is pushed to the nextIO-Type stack followed by 'example'. As a result, the system presents more examples and then the interactive IO (e.g. exercise) with the same type as the current IO to the learner.
Repeat with lower level	The ability of the learner is decremented by one level in order for the system to present learning content with lower difficulty level.
Assessment	The assessment as a type of IO is pushed to the stack in order to present an assessment to the learner in the next step.

Table 5.1: The reaction of Guide component to different guidance

5.6.4 Assessment Analyser

Assessment analyser analyses learner's responses to the assessments according to item response theory in order to calculate the ability of the learner. As we mentioned in Chapter 2, in order to estimate the ability of a learner, we must assume that all the items in the assessment are calibrated (see Section 4.8). It means, the numerical parameters of the items in the assessment are known. The direct result is that the scale of the measurement is the same as the scale of the parameters in the items. After attempting all the items in the assessment, the system receives learner's response and scores the items dichotomously. This means that the learner gets one for a correct answer and zero for an incorrect answer. Hence, we will have a response pattern $(U_1, U_2, U_3, \dots, U_j, \dots, U_n)$ which is called the assessment response vector, where $U_j = 1$ represents a correct answer given by the learner for the j^{th} item in the assessment. On the contrary, $U_j = 0$ represents an incorrect answer given by the learner to the j^{th} item in the assessment. After that, under item response theory, the Maximum Likelihood Estimator (MLE) is applied to effectively estimate the learner's abilities (Hambleton et al., 1991). MLE is an iterative procedure. It gets the items' parameters from assessment ontology and a priori value for the learner's ability. After that, the probability of the correct response to each item is calculated for the learner based on a three parameters logistic function (See Chapter 2). This formula is shown below:

$$P_i(\theta) = c_i + (1 + c_i) \frac{1}{1 + e^{-a_i(\theta - b_i)}} \quad (5.1)$$

Where:

θ is the ability level of learner

b_i is the difficulty parameter of item i ($-3 \leq b_i \leq +3$)

a_i is the discrimination parameter of item i ($0 \leq a_i \leq +1.7$)

$P_i(\theta)$ is the probability that learner with ability θ can response correctly to the item i

Then, according to the calculated probability, the MLE procedure estimates the new ability of the learner. The procedure is repeated until the change in the estimated ability is less than a threshold value (i.e. becomes stable). The estimation equation is as follows:

$$\theta_{s+1} = \theta_s + \frac{\sum_{i=1}^N -a[u_i - P_i(\theta_s)]}{\sum_{i=1}^N a_i^2 P_i(\theta_s) Q_i(\theta_s)} \quad (5.2)$$

Where:

θ_s is the estimated ability of the learner within iteration s

a_i is the discrimination degree of item i

u_i is the response made by the learner to item i :

$u_i = 1$ for a correct response

$u_i = 0$ for an incorrect response

$P_i(\theta_s)$ is the probability that learner with ability θ can response correctly to item i within iteration s .

$Q_i(\theta_s) = 1 - P_i(\theta_s)$ is the probability that learner with ability θ responses incorrectly to item i within iteration s .

As we know, in classical test theory, the learner gets a low score on a difficult exam and a high score on an easy one. Therefore, the result of the exam cannot be relied upon to prove the underlying ability of the learner. However, under item response theory, as the values of all the item parameters are in a common metric and they measure the same latent trait, the learner's ability is invariant in different exams. In other words, if a learner takes a hard or an easy exam, she obtains the same estimated ability. On the other hand, if a remedial learning is delivered to the learner, the learner's ability level would be changed in the exam. Accordingly, the learner's ability level may change. Consequently, based on the estimated ability, the learner model is updated and in the next step of the learning process an IO is recommended to the learner according to her new ability.

5.7 Creating a Personalised Learning Content in Rule-PADEL

Figure 5.3 shows the cycle Rule-PADEL follows when creating a personalised learning. A new learner signs up by completing the registration form to create her learner model. If there is a learner model already associated with the learner, then she could just use her credentials to log into the system.

From Figure 5.3 a learning session starts when a registered learner logs into the system. According to the information in the learner's model the adaptive structure of the course is presented to the learner in the form of an Annotated Course Structure (ACS). As the topic selection is being made, the adaptation model generates adaptive learning content based on the learner's profile. Rule-PADEL offers some interactive IOs during the learning process in order to make the learner's experience active. The learner's responses to interactive IOs inform the system about her difficulties. Adaptation model then provides guidance for the learner to remediate her difficulties. The system also provides feedback when receiving learner's responses to the IOs.

After the learner requests an assessment, Rule-PADEL selects and presents a suitable assessment to the learner. It evaluates the learner's answers on-the-fly based on IRT, provides feedback for the learner and updates the learner model. If the learner continues her learning process, the updated ACS is presented to her and from then on, the system adapts to this new updated information available in the learner's model.

5.7.1 Improve the Efficiency of System

In the first version of the system, the adaptation process was developed in such a way that after firing the rules, only one IO was presented to a student at any single time. Once this IO was dealt with by the student (read in the case of a written definition or attempted if an exercise), the next IO was decided as the result of the rules being re-fired. This means that the knowledge bases were reasoned over, after the student finished working with one single IO. Since the reasoning process is computationally expensive, this made the performance of the system very poor.

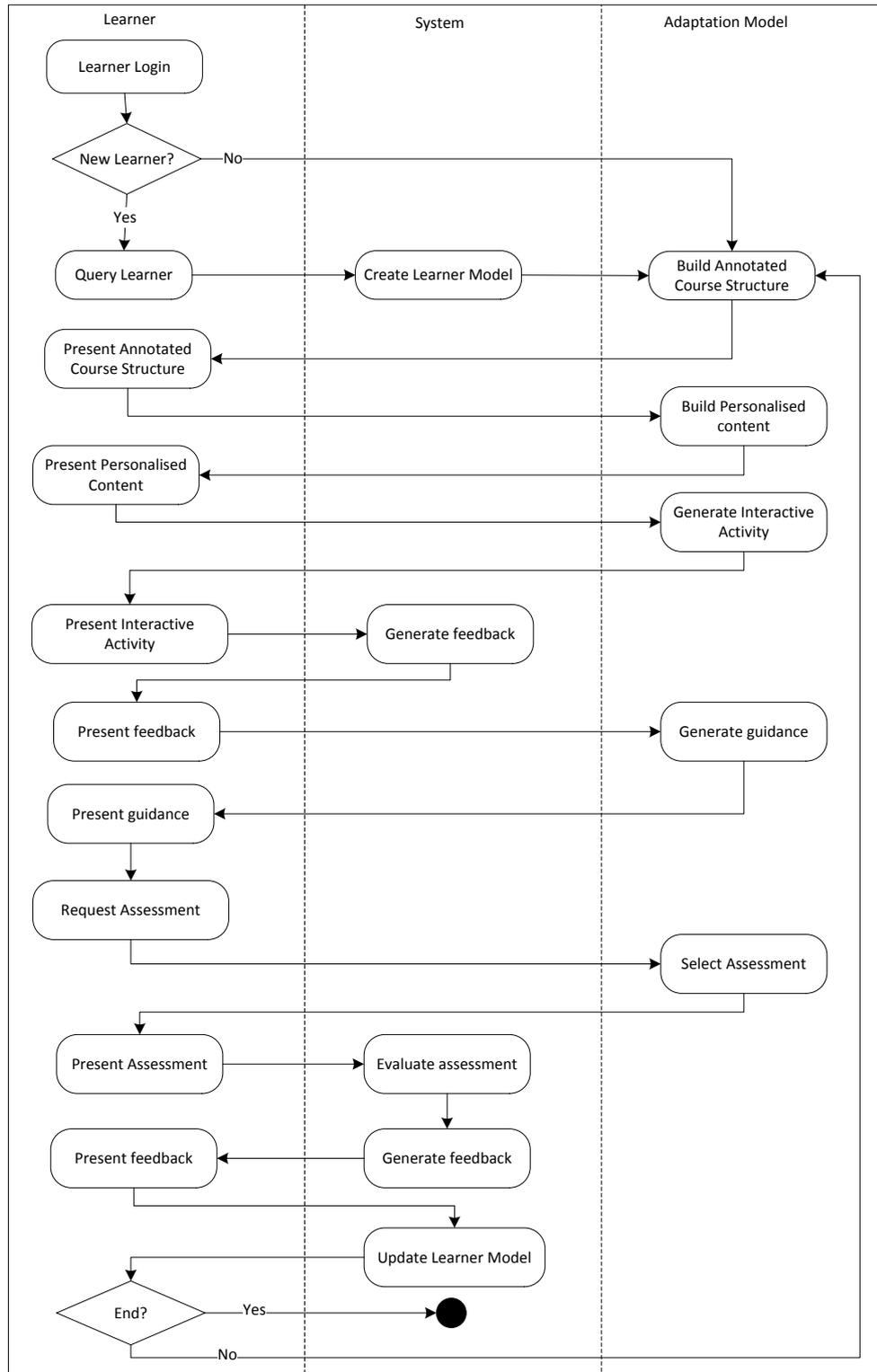


Figure 5.3: Rule-PAdel Personalised Learning Process

In the next version of the system, this issue was partially resolved. In this version, instead of generating one IO at a time, the adaptation mechanism was redesigned to recommend a series of IOs for the student based on the instructional plan. This series had the property of ending with an interactive IO (e.g. an exercise). Therefore, a student could attempt the IOs in the sequence until she gets to (an) interactive IO(s) where she sends her responses to the system. Upon receiving the response from the student, the system would evaluate it, update the knowledge bases, and reason over the updated knowledge bases to infer another sequence of IOs for the next step.

There were two important achievements in the second version of the system:

- The performance of the system was vastly improved; since the frequency of reasoning over knowledge bases was decreased, while still having the same precision for adaptability.
- The student was given more control over choosing to do and the order of attempting the given IOs, while preserving the order which the instructional designer had planned for the student based on various criteria.

5.8 Instance of Rule-PAdel Personalised Learning on Fraction

To complement the implementation of Rule-PAdel a personalised learning was developed on the topic of fractions in mathematics. The topic comprises of 26 sub topics and utilises learning content from a repository of several fine-grained IOs (See Appendix E domain math ontology). Over 100 pedagogical relationships are defined between these subtopics to find the optimal learning path. These relationships are all of different types defined in the domain ontology such as *hasPrerequisite*, *isTaughtAfter* and *isRelatedTo*.

At the start of the first session learners complete a registration form and the first version of their learner model is created. During the registration process, the system asks the users to fill in a form to capture their prior knowledge. They are also asked to fill in the VARK questionnaire that contains 16 questions in order to calculate their learning style.

In this implementation, there are a several IOs on the same topic but with different media

types to support different learning preferences based on the VARK model. In order to extend personalisation in this system, equivalent IOs covering the same topic are produced with different difficulty levels to cover learners with different abilities.

In *Fraction Learning System*, an instructional plan is defined in content ontology storing the tasks the learner should do and the order in which they should appear. In the case of this system, the instructional plan holds the following IO type about a selected topic in the first learner's encounter: one definition, three examples and three easy exercises. After that, based on learner's responses to the exercises, different supplementary contents and different exercises with different difficulty levels are delivered to her. Whenever the learner responds to three exercises with moderate difficulty level correctly, she can then take on an assessment to evaluate her about her newly acquired knowledge.

The instructional designer for the fraction learning system defined several guidance vocabularies to be used in the adaptation rules. These vocabularies include *moreExample*, *repeatPrerequisite*, *repeatWithLowerLevel*, *moreActivity*, *learnWithHigherLevel*, *nextLevel*, *assessment*, and *learnNewTopic*.

The screenshots of personalised fraction learning system in different stages are included in the Appendix D.

5.9 Summary

This chapter discussed the implementation of an adaptive e-learning system called Semantic Rule-based approach for Personalised Adaptive e-learning (Rule-PAdel), to validate that our approach is implementable and to enable the evaluation of the intended functionalities. First the technological architecture of Rule-PAdel has been presented. Then, the primary component models of Rule-PAdel which are learner, content and adaptation are described in depth. The rule-based adaptation process and assessment analyser issues were described. The cycle of creating personalised learning in Rule-PAdel is presented to demonstrate the operations of system. Finally, the chapter described a personalised learning system created to cover the topic of fraction in mathematics domain.

Chapter 6

Evaluation

6.1 Introduction

The aim of this chapter is to evaluate our semantic rule-based approach of adaptive personalised e-learning in order to examine whether it fulfils the objectives of this research. Throughout the evaluation process, we will use the Rule-PAdel system which is implemented based on our approach.

After this introduction, in Section 6.2 the methodology of evaluation was described. Then Section 6.3 and 6.4 describe the benefits of studying using a personalised e-learning system based on a semantic rule approach adapted to different students. These benefits can be recognised through examining learner's satisfaction and the learning effect of the system for learners. In order to evaluate users' satisfaction, learners are presented with a questionnaire to fill in, reflecting on their opinion about the different aspects of their experience using the system. Moreover, the effectiveness of the system was examined by determining the knowledge learners gained when working with Rule-PAdel.

After that, in Section 6.5, the satisfaction of both authors and teachers will be evaluated through interviewing them. The teachers involved in using the adaptive system as a part of their curriculum and the authors engaged with creating the personalised content were asked about their experiences with the system.

These three sections therefore evaluate the generated adaptive learning approach in terms of being pedagogically effective and satisfactory for learners and teachers who used it,

which is one of the key objectives of this research.

In the following two sections the advantages of taking a semantic rule-based approach for implementing a personalised e-learning system will be determined through examining the technical advantages of modelling the components of adaptivity (e.g. learner profile, content, domain and adaptation techniques) using semantic ontologies enriched with semantic rules. This is the second objective of our research.

In Section 6.6, the flexibility and extensibility of our developed system will be evaluated by adopting different adaptation techniques, domain and instructional plans, creating different versions of the system. These features are thoroughly examined as they are what differentiate our approach from the existing personalised e-learning systems as discussed in Chapter 3.

In Section 6.7, we examine the reusability of the components of adaptivity such as adaptation techniques (in adaptation model), concept sequencing (in domain model), instructional plans, and content (in content and assessment models) through representing each component in a separate specific ontology.

In Section 6.8, we compare flexibility, extensibility and reusability of Rul-PAdel with some other adaptive e-learning systems.

6.2 Methodology

In order to evaluate the semantic rule-based approach, Rule-PAdel which is an implementation of this approach was evaluated from two aspects:

- **Learner's satisfaction:** Examining whether the generated adaptive learning paths are pedagogically effective and satisfactory for learners and teachers
- **Flexibility, extensibility and reusability:** Examining the flexibility, extensibility and reusability of Rule-PAdel which is the result of using ontology along with semantic rules.

In the evaluation process, 50 students are selected to work with the system to learn several topics, each in a session. In each session two exams are taken from the students, one

before and one after working with the system, namely the pre and post-tests. At the end of the experiment, each learner filled-out a questionnaire to reflect his/her perceptions about different aspects of the system.

After this process is complete, the teachers who are involved in executing the evaluation procedure were interviewed and asked to comment on their experience.

The course authors and instructional designers are also were interviewed and asked to express their opinion about the flexibility, extensibility and reusability of the system.

As we explained in section 4.2, this research uses both quantitative and qualitative methods to validate its results. The qualitative aspect of the research does not impose any requirements on the size of the sample data as for these types of research, “size does not matter” but the feeling and the view point of the applicants are important. In our research, we have an open comments section in the questionnaire so that the students can state their opinion about the system in this section. In addition to that, the teachers, instructional designers and authors are also interviewed to express their opinion about the system.

Additionally, in this research we use t-tests to determine the pair-wise comparisons between the learner scores on the pre-test and the post-test. In t-test if the p-value is less than 0.005, it indicates that the observed result is unlikely to be random with 95% probability and the result is true for the population of the students.

6.3 Learner Satisfaction

The most important stakeholders in e-learning are the learners and any approach to adaptive e-learning must generate personalised learning that is satisfactory to the learners. Therefore, the goal of the experiment is to demonstrate:

1. Effectiveness of the personalisation provided by Rule-PAdel through examining learners’ performance, and
2. Learners’ satisfaction through eliciting learners’ opinion about different aspects of the system.

This section presents the results of the learner's survey about the proposed system. It starts with describing the details of the experiment design. After that, the analysis of the learners' evaluation about the system based on a questionnaire's result is reported. Then the results of the analysis of the effects of the learning process on the learners are summarised. Finally, a summary of results for evaluation is concluded.

6.3.1 Overview

In order to estimate the effectiveness of the proposed approach, two different versions of the system were designed. The first version is a system which adaptively supports the learner through the learning process. The other is a variation of a non-adaptive system, identical to the adaptive version with its user interface but without any adaptive features. In this version, learners are free to navigate to different topics and to attempt different exercises. In the adaptive system, learners are recommended using adaptive techniques based on each learner's profile, the result of the learner's interaction with different IOs and scores achieved from doing an assessment in the previous learning steps. Both systems were used in teaching *Fractions* in mathematics domain.

The topics were divided into two categories: easy and hard. Some topics are in both categories but with different difficulty levels and some only belong to one category. Table 6.1 shows the topics and their respective categories. A learner can start learning topics in the second group only if she has mastered the topics of the first one.

During the experiment, each learner experiences with both sets of topics. The learners explored the adaptive system for one set of topics and the non-adaptive system for the other set. The order in which the topics are organised stayed the same for all learners but the combination of the categories and the systems varied.

We selected 50 applicants with varied learning abilities who were interested in testing the Rule-PAdel learning system. Learners were randomly assigned into groups A and B. Group A used the adaptive version of the system while working with the topic in Set 1 and the non-adaptive version for working with the topic Set 2. While Group B did the inverse, as they used the non-adaptive version for Set 1 and the adaptive version for Set 2. There were 25 learners in each group with the average age being around 16 years old.

Introductory Topic Session 1	Topic Set 1	Topic Set 2
Explanation of the study	Proper Fraction	Equivalent Fraction
Familiarity with system	Improper Fraction	Simplifying fraction
Concept of fraction	Comparing Fraction with same denominator	Add Fraction with different denominator
-	Ordering Fraction with same denominator	Subtract Fraction with different denominator
-	Add Fraction with same denominator	Multiple Fraction
-	subtract Fraction with same denominator	Division Fraction

Table 6.1: Fraction topics used in the evaluation

The first session of the experiment was the same for all learners in both groups. This session included a brief explanation of the study, familiarisation with the system and familiarity with the concept of fractions (See column one of Table 6.1). Other sessions contained a 10-minutes pre-test, a 30-minutes interaction with the system and a 10-minutes post-test. Each of the tests contains 10 questions. Based on the results of the pre- and post-tests, knowledge gained values were calculated to determine the effectiveness of the adaptive learning techniques. At the end of the experiment each learner filled-out a questionnaire to reflect their perceptions about different aspects of the two systems.

6.3.2 Learner's Evaluation

At the end of the experiment, each learner was presented with a questionnaire to fill in so we can collect learners' opinions on the adaptive learning. The questionnaire comprised of twenty questions and an open comments section. In the questionnaire learners were asked to express their opinion about their experience, based on the five Likert-type scale (Likert, 1932) ranging from "strongly agree" to "strongly disagree". The questions were grouped into five general categories. By filling out the questionnaire, it is hoped that we find out whether the adaptive system has improved the learners' satisfaction or not. The categories of the questions, and the information they elicit, are as follows:

- Learner's opinion about adaptive and non-adaptive system; this category compares the learner's satisfaction with regards to both adaptive and the non-adaptive learning

experience.

- Satisfaction with adaptive navigation; examined in this category is the usability of the *Annotated Course Structure* (ACS) for navigating to different topics.
- Appropriateness of the personalised content; this category examines the learner's satisfaction with regards to the personalised content they were presented with.
- Usage of the adaptive guidance; this category examines how the learners used the adaptive guidance.
- Learner's interests; examined in this final category is the effect of the adaptive system on learner's learning motivations or interests.

The questions associated to each category are listed in Table 6.2. The complete questionnaire can be found in Appendix F. The itemised results of the analysis of learners' responses to the questions are going to be discussed in the following sections.

Comparing Adaptive and Non-adaptive System

The questions in this category intend to investigate general feelings towards the adaptive and the non-adaptive learning experience. As the learners have tried both for a topic set, we asked them about different aspects of the two systems implementing the approaches. Figure 6.1 visualises the learners' opinion about the adaptive and the non-adaptive systems.

Questions 1, 2, 3 and 4 compare the adaptive with the non-adaptive system. From Question 1, we realise that 74% of learners found the adaptive system more helpful for learning new topics and only 2% of the learners found it otherwise with the rest of them having no strong opinion.

Question 2 asked the learners whether the recommendations of the adaptive system helped them to solve their difficulties with 84% of the learners agreeing to that.

Question 3 suggests a possible reason as to why the adaptive system is more helpful by asking: "*I prefer adaptive system because it provides interactive features for the learners*". 74% of the learners agreed and only 12% disagreed.

Question 4 tries to determine the reason for the usefulness of adaptivity by asking: "*I prefer*

Category	Question Number	Questions
Comparing adaptive systems and non-adaptive systems	1	The adaptive system was more helpful for learning a new topic than the non-adaptive system.
	2	It was easier to solve my difficulties with the help of the recommendations of the adaptive system than in a non-adaptive system.
	3	I prefer adaptive system because it provides interactive features for learners.
	4	I prefer adaptive recommendations because it accurately brings helpful supplementary contents tailored for my needs, and I did not have to look for them in the entire course.
Satisfaction with adaptive navigation	5	The colours used in the course structure make the items more clear.
	6	The generated annotated course structure is easy to navigate.
	7	The adaptive annotations of the annotated course structure helped me to choose the most appropriate next topic to learn.
Appropriateness of the personalised content	8	The personalised services provided by the adaptive system satisfied my preferences.
	9	I am satisfied with the difficulty level of materials recommended by the adaptive system.
	10	I am satisfied with the difficulty level of the exercises and assessments recommended by the adaptive system.
	11	I am satisfied with the quality of the personalisation.
Usage of the adaptive guidance	12	I benefited from the materials recommended by the adaptive system.
	13	The adaptive guidance helps me to solve my difficulties.
	14	The adaptive guidance helps me to plan the next step of my learning.
	15	The adaptive guidance helps me to access sufficient material, examples and exercises to solve my difficulties.
Learner's interest	16	I tried to use the guidance offered in the adaptive system.
	17	I believe that the system is user-friendly.
	18	The adaptive system can promote my learning interests.
	19	I feel that the time passes very quickly when I use the adaptive system to learn mathematics.
	20	I would recommend this system to my classmates.

Table 6.2: The questions associated in different areas of questionnaire

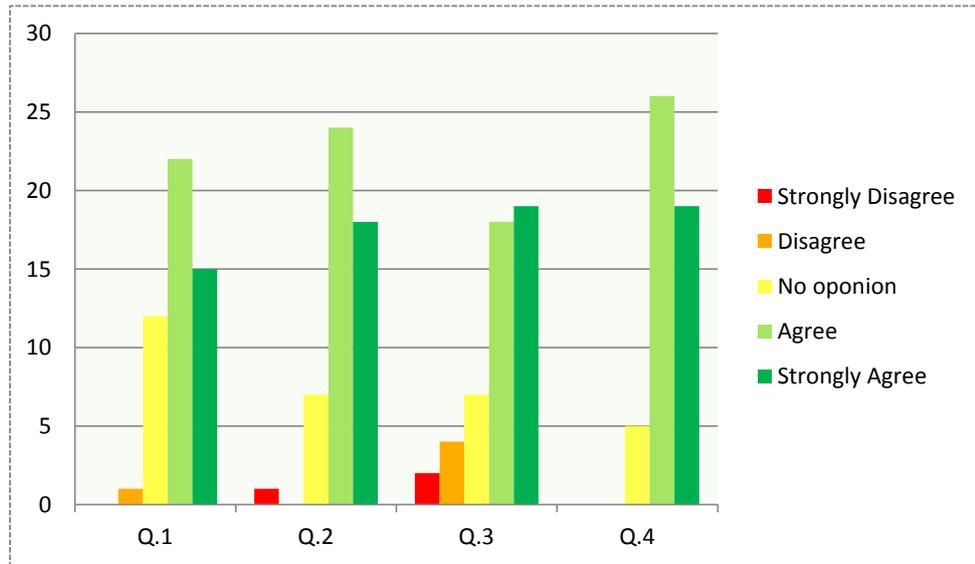


Figure 6.1: Learners' response to Comparing adaptive and non-adaptive system questions

adaptive recommendations because it accurately brings helpful supplementary contents tailored for my needs, and I did not have to look for them in the entire course.” 90% agreed that this is the reason for the usefulness of the adaptivity. 10% of the learners did not have any opinions and there was no disagreement regarding this question.

The result of the questions in this category show that the learners strongly prefer the adaptive system over the non-adaptive system one as it prepares personalised content according to their needs and it does it in a way that they do not need to search for suitable materials throughout the entire course.

Satisfaction with Adaptive Navigation

The questions in this category were used to determine the learners' usage of the navigation features in the adaptive system. Figure 6.2 visualises the level of learner's satisfaction with the adaptive navigation.

Question 5 investigates that *“The colours used in the course structure makes the items clearer.”* 56% of learners agreed with this statement. Only 4% of learners disagreed and the rest of them had no opinion about these colours. Questions 6 and 7 verified the helpfulness of adaptive annotation used for navigating to the most appropriate topic according to the learners' level of knowledge. More than 54% of learners thought that these annotations

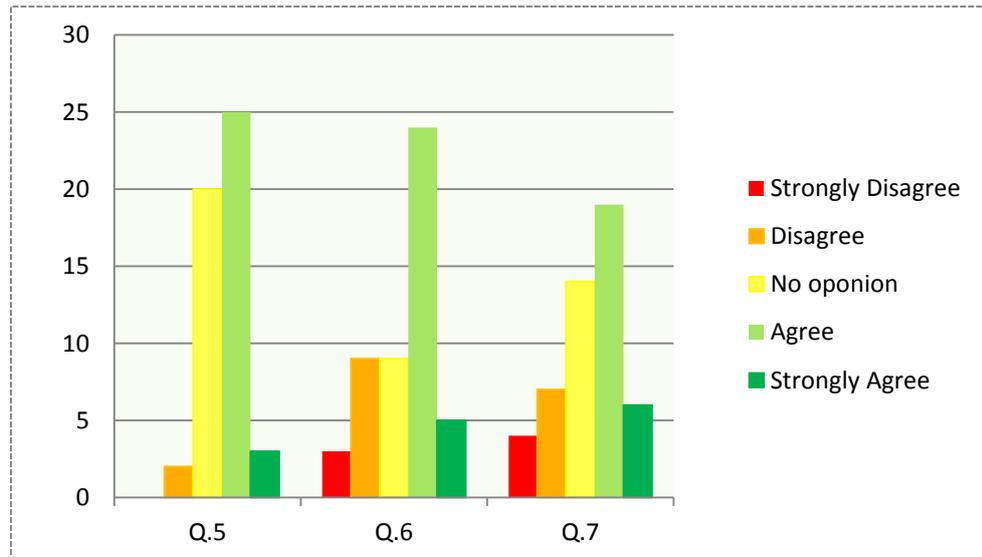


Figure 6.2: Learners' satisfaction level about adaptive navigation

helped them to choose the next topic while 23% of learners did not agree.

Overall learners were satisfied with the adaptive navigation of our system and only a minority of learners were not happy. Most of the unsatisfied learners were those who liked to have more control over selecting and visiting different topics. This makes sense since the adaptive navigation prevents learners from visiting the topics for which they have not passed the prerequisites. In the future implementations this option will be enabled for users. However, upon trying to access such content, the system would warn the user that he is about to access a content which is beyond his abilities.

Appropriateness of the Personalised Content

The reason behind getting this information is to evaluate the quality of personalisation and also to determine whether the generated personalised content meets the learner's requirements and expectations. Questions 8 to 12 of the questionnaire were designed in order to investigate the learners' feelings about personalised content. The summary of learners' responses to these questions are visualised in Figure 6.3.

Question 8 attempts to examine whether the personalised content delivered to the students reflected their preferences. As it can be seen in Figure 6.3, 60% of the learners felt that the personalised content generated by Rule-PAdel fulfilled their preferences and 10% did not have any opinion about it. Only 20% of the students said that the content did not reflect

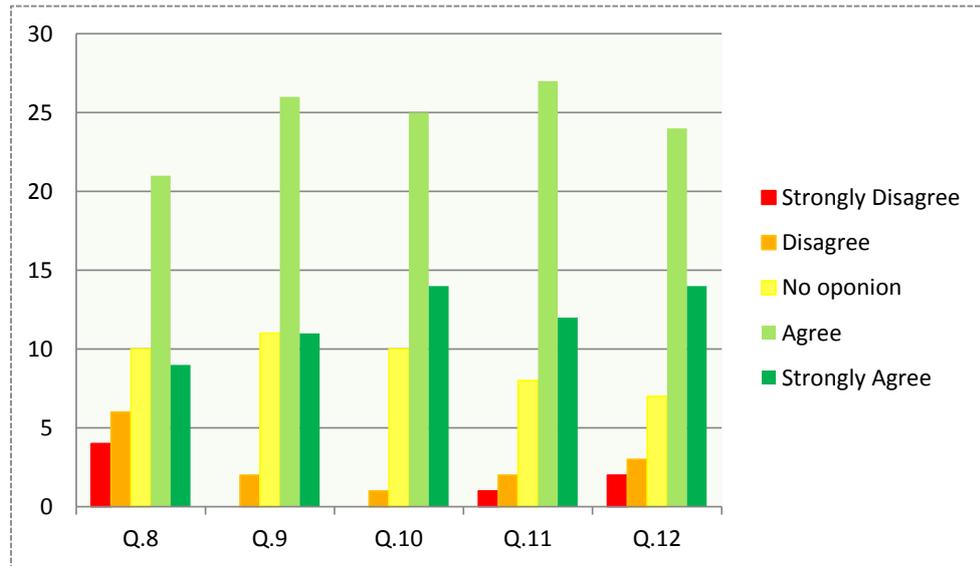


Figure 6.3: The suitability of personalised content

their preferences. A large proportion of these learners were those with prior knowledge about fractions. Again, these learners wished to have more control over content selection which they expressed by leaving comments in the open comments section.

Since materials that are too easy or too difficult to master frustrate students, Questions 9 and 10 are designed to observe the suitability of the material's difficulty level (e.g. content, exercises and assessments) for learners. More than 76% of learners are satisfied with the difficulty levels of the materials and responded that they understood the materials well. Only 3% of learners were not satisfied with the delivered content. These learners generally had fundamental difficulties with other related topics (e.g. Arithmetic operation) which are not covered in this learning system. The rest of the learners had no opinion on these questions.

Questions 11 and 12 attempt to examine learners' opinion about the quality and also usefulness of personalised content recommended by Rule-PAdel in general. As we can see in Figure 6.3, 77% of learners replied that overall they benefited from the personalised contents and they were also satisfied with their quality, 15% of the learners had no strong opinion and the other 8% of learners were not happy with the personalised content.

In general, the students found the experience of using personalised content suitable, while only a minority of learners (8% of them) did not have this feeling. After considering the other questions and also the open question we found that most of these learners had some

prior experience with the Fraction topic, therefore, they liked more control over the content and the minority of them had difficulties with the prerequisite topics which are not covered in this learning system (e.g. Arithmetic operation). Therefore they could not understand the learning content which is not related to the qualification of personalisation.

Usage of the Adaptive Guidance

In this category we attempt to capture Learners' experience of the adaptive guidance to remediate their difficulties in completing the exercises. These questions are designed to investigate learners' satisfaction when using this facility and the supplementary content. The learners' opinion about the adaptive guidance is visualised in Figure 6.4.

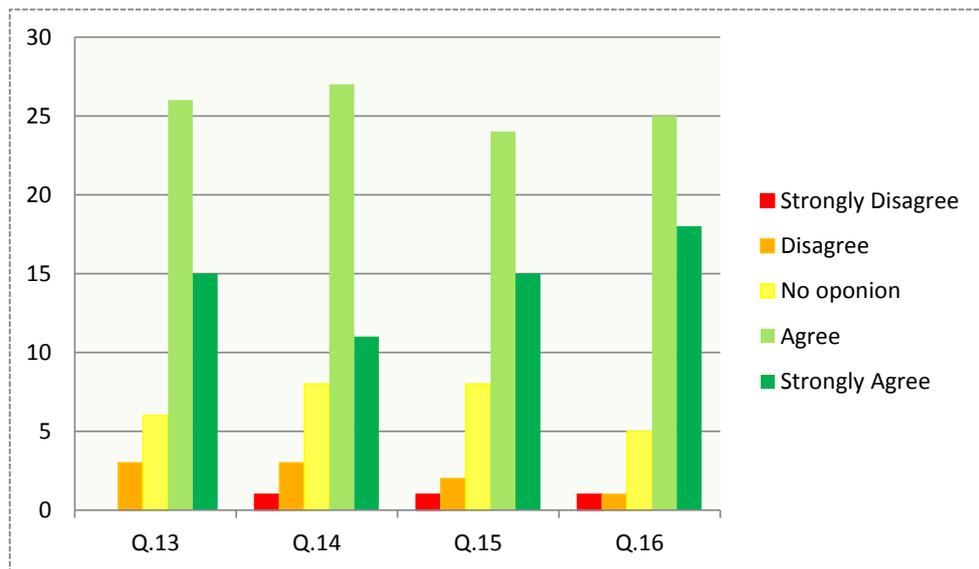


Figure 6.4: The usage of adaptive guidance

For examining the usability of adaptive recommendation, in Questions 13 and 14 we asked the learners whether adaptive guidance helped them to resolve their difficulties or to plan the next step of their learning. As it can be seen in Figure 6.4, 82% and 76% of learners responded positively to these two questions respectively. In both questions, less than 8% of learners replied with negative answers and other learners did not have any strong opinions.

According to answers for Question 15, learners liked the overall idea of supplementary contents (e.g. material, example and exercise) being recommended to them in order to remediate their difficulties (78% positive, 16% neutral and 6% negative).

Question 16 asked learners about the usability of recommended guidance in Rule-PAdel in general. Figure 6.4 shows that more than 86% answered affirmatively, while only 4% of learners responded dissentingly, leaving only 10% of the students with no strong opinion.

The results of the evaluation indicated that adaptive guidance based on learner’s progress successfully helped them to mentally organise their learning, and drew their attention to the topics and materials they needed to focus on the most. Only a minority of learners felt negative about the adaptive guidance with a large proportion of them being expert learners who answered correctly to most given exercises. Thus they did not feel the need for any adaptive guidance for resolving their problems. In other words, they did not feel to have any difficulties, thus no need for guidance to remediate it.

Learner’s Interest

The questions asked in this category try to examine learners’ desirability to work with the adaptive system. It also determines the effect of adaptive learning on promoting learners’ learning motivations and interests. Figure 6.5 visualises learners’ responses to these questions.

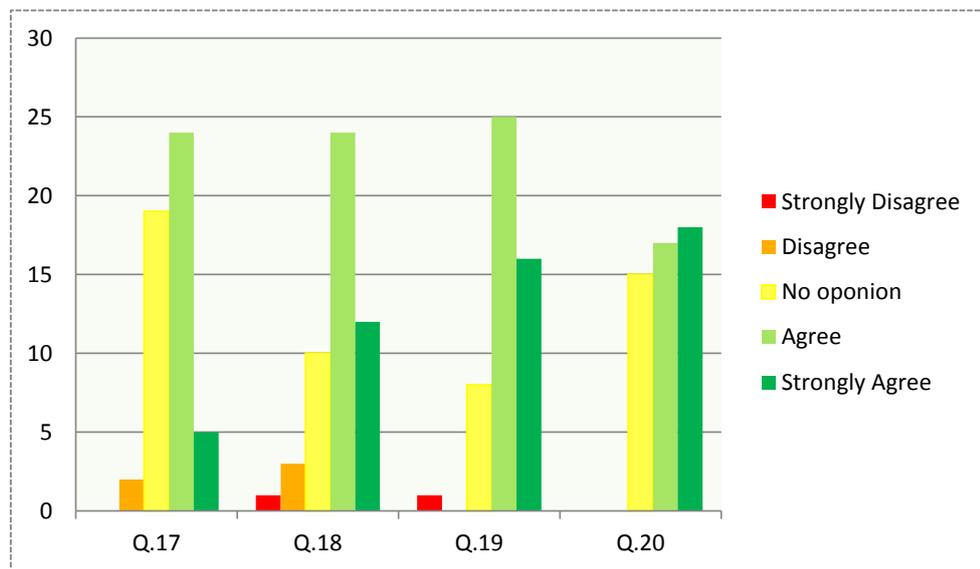


Figure 6.5: The effect of adaptive system in promoting learners interest

Firstly, learners were asked to evaluate whether the system was user friendly or not. As we can see in Figure 6.5, 58% of learners replied positively. This figure seems to be not as satisfying as the other ones. This is due to the fact that when asked about the user

friendliness of our system, the responders considered many issues with respect to the user interface. These issues, although valid in their own right, were not the focus of this research. Were this system to be commercialised, more effort on this side of the system would be required. However, as this system is only a prototype for demonstrating our hypothesis, the main focus here is on achieving personalisation and adaptability in learning new topics. Hence, overall we have more questions on these two main issues.

In Questions 18 and 19, we asked the learners about the effect of the adaptive system for promoting their engagement in learning. The result of these questions indicates that 77% of the learners' responded positively and only 5% of them responded negatively (other learners did not have any strong opinion).

In Question 20, we asked about whether learners would recommend the adaptive system to their classmates. 70% of the learners either strongly agreed or just agreed with no negative opinions given.

The main findings

In general, we found a number of key findings from the learners' opinions and experiences with using adaptive learning. The foremost is that the vast majority of learners were satisfied with all aspects of personalised learning, particularly with adaptive guidance and personalised content features. This is shown in Figure 6.6 by visualising learners' responses given in the questionnaire.

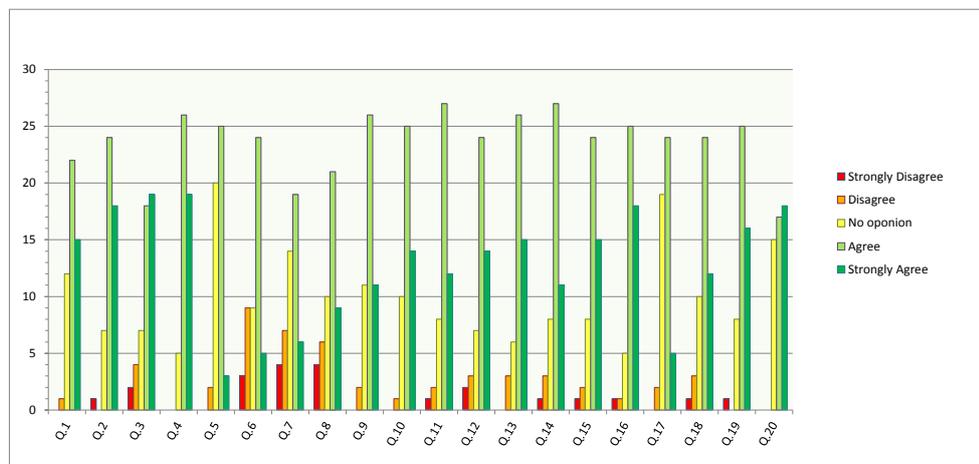


Figure 6.6: Summary of learners' evaluation

Additionally, we found that adaptive learning is more beneficial for novice learners who had low ability and less experience, as they need prescriptive learning experience, while expert learners who had more information about the subject (e.g. Fractions) wished to have more control over their learning experience.

The adaptive guidance promotes learners' motivation by supporting them in solving their difficulties. Also, the adaptive guidance prepares supplementary content which remediate learners' difficulties. Without it they would spend quite some time searching and finding suitable content which is monotonous for them, and it is sometimes nearly impossible.

The most significant finding out of learners' evaluation is the slow performance of the adaptation process. This issue is partly due to the constant connection to ontology knowledge bases, and because the reasoner (in this case Pellet) reasons over the whole knowledge base after any data is changed. For example, if a student incorrectly answers an exercise, pellet would reason over the whole knowledge base in order to find out what the next step would be. This performance issue was resolved in the second version of the Rule-PAdel (See Section 5.7.1 Improve the Efficiency of System) by recommending a series of IOs for the student instead of generating one IO at a time. However, the performance of system can be improved in future versions of the system by setting information dependencies in a way that avoids the whole knowledge base from being rereasoned on.

6.4 Learner Performance

One of the most important parts when evaluating an educational system is to analyse how much the learners have learned from the topic(s). To determine the effectiveness of our approach, we need to prove that learners' interactions with the system actually resulted in them gaining new knowledge. Therefore, in this subsection, we consider the results of pre- and post-test scores over the period of this experiment, to assess the effectiveness of the system. In order to be more accurate in our data, the Knowledge Gain (KG) and the Normalised Knowledge Gain (NKG) scores are calculated based on work from (Sosnovsky, 2011). KG is the difference between the learner's score in post and pre-test. Equation 6.1 was used to calculate adjusted knowledge gain. This equation omits the negative values as it is considered invalid data.

$$KG = \text{Max}(0, \text{Score}_{\text{Post-test}} - \text{Score}_{\text{Pre-test}}) \quad (6.1)$$

However, this formula does not take into account the differences in the learners' initial knowledge levels. Therefore, normalised knowledge gain (NKG) was used to calculate the effectiveness of the system. NKG is defined as the ratio of the actual knowledge gain to the maximum knowledge gain possible. This Equation is as follows:

$$NKG = \frac{\text{Max}(0, \text{Score}_{\text{Post-test}} - \text{Score}_{\text{Pre-test}})}{\text{Score}_{\text{Max}} - \text{Score}_{\text{Pre-test}}} \quad (6.2)$$

The following subsections present the effectiveness of Rule-PAdel on different aspects of students' learning.

Learning Effect for Complex Materials

Tables 6.3 and 6.4 demonstrate the mean, standard deviation for pre-test, post-test, knowledge gain and normalised knowledge gain for our analysis of the topic Set 1 (easy topics) and 2 (difficult topics) respectively.

Group	Pre-test Score		Post-test Score		Knowledge Gain		NKG	
	Mean	Std. Deviation	Mean	Std. Deviation	Mean	Std. Deviation	Mean	Std. Deviation
A	11.24	3.49	14.76	2.36	3.53	1.70	0.42	0.16
B	10.53	3.86	13.24	4.16	2.88	1.73	0.31	0.20

Table 6.3: Test and knowledge gain result (Topic Set 1)

Group	Pre-test Score		Post-test Score		Knowledge Gain		NKG	
	Mean	Std. Deviation	Mean	Std. Deviation	Mean	Std. Deviation	Mean	Std. Deviation
A	9.76	4.15	10.47	4.33	1.24	1.25	0.12	0.13
B	9.71	4.04	14.53	3.50	4.82	1.98	0.5	0.20

Table 6.4: Test and knowledge gain result (Topic Set 2)

During learning topic Set 1 (the easy learning material), there is no significant difference between KG and NKG, although Group A, who worked with the adaptive system, obtained better KG and NKG. While, once the learning material became more complex (topic Set 2), the adaptive system significantly outperformed the non-adaptive system.

In order to verify that having interactions with the system actually leads to learning the materials, pair-wise comparisons of scores on the pre-test and the post-test have been undertaken (per group, per topic set). Table 6.5 summarises the results of four paired samples t-tests¹ for groups A, B and two topic sets. During learning topic Set 1, both groups have significantly learned the new topic (the result of post-tests were significantly higher than the result of pre-tests). However, Group A, who worked with adaptive system, showed more progress in comparison with Group B, who worked with the non-adaptive system. During learning topic Set 2, the adaptive system (grey cells in the table) resulted in significant learning. At the same time, the non-adaptive system (white cells in the table) showed no learning (P-value is greater than 0.005)¹.

Group	Topic Set 1				Topic Set 2			
	Paired Differences		T	P-value	Paired Differences		T	P-value
	Mean	Std. Deviation			Mean	Std. Deviation		
A	3.53	1.70	8.56	<0.001	0.70	1.90	1.53	0.07
B	2.71	2.08	5.35	<0.001	4.82	1.98	10.06	<0.001

Table 6.5: The result of the Matched-Pairs T-Tests between Score_{pre-test} and Score_{post-test}

The main difference between Topic Sets 1 and 2 is the complexity of the material. Topic Set 2 uses the knowledge introduced in topic Set 1 and extends it with more advanced topics such as fractions with different dominators. The analysis of learning demonstrates that personalised learning have higher impact on the learning of complex material.

This is an important finding as when learners deal with easy materials, they need less support from the adaptive guidance. They usually improve their understanding of the topic by just practising with the interactive IOs (e.g. exercises) and in most cases they never even require supplementary content or personalised learning. On the other hand, when they engage with more complex materials, personalisation and adaptation becomes more beneficial for them. When learners respond incorrectly to interactive IOs (e.g. exercises), they can benefit from adaptive guidance and supplementary content to remediate their difficulties.

¹When the P-value is less than 0.005, with 95% probability, the assumptions are valid for the tests; this is not the case for the non-adaptive systems for topic set 2.

Learning Effect on Weaker Learners

The analysis of the effect of learning is presented here in the same way it was presented in the previous section with the main focus now being on weak learners. The correlation between knowledge gain and pre-test scores of learners in the adaptive and the non-adaptive system are calculated. Figure 6.7 presents two scatter plots showing this correlation. The left-hand side plot shows the collected data from the learning topic Set 1, and the right-hand plot presents the data from the other set. In both plots, there is a clear negative correlation between knowledge gain and the pre-test scores. Learners with lower pre-test scores gained more knowledge using the adaptive system than ones with higher pre-test score. However, in the non-adaptive system, there is no such correlation. These plots show that the weaker a student, the more benefits she gets from adaptation. Table 6.6 shows the statistical data about weak students when learning both topic sets.

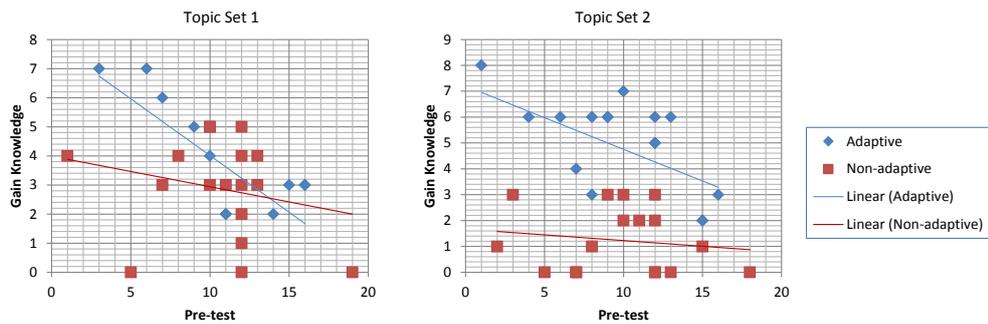


Figure 6.7: Knowledge gain vs. pre-test scores: Left Topic Set 1; Right Topic Set 2

	Topic Set 1		Topic Set 2	
	Adaptive system	Non-adaptive system	Adaptive system	Non-adaptive system
Score _{pre-test}	8.5	7.75	7.0	7.2
Score _{post-test}	13.0	10.88	12.3	8.2
KG	4.5	3.37	5.3	1.5
NKG	0.38	0.30	0.42	0.13

Table 6.6: Statistics about weaker learners for learning

The results of the analysis reveal that weak students’ need more personalised content and guidance. They usually have problems in understanding different materials with various difficulty levels. Therefore, personalised contents and adaptive guidance would benefit them a lot especially when they respond incorrectly to an interactive IO (e.g. an exercise).

On the other hand, more advanced learners benefit less from adaptivity, for they already grasp a large percentage of the materials they read/attempt.

Overall Key Findings

Overall, we realised that by working with adaptive system the learners' performance is improved. This could be confirmed by comparing the scores of the learners who studied with the adaptive system with those who studied with the non-adaptive system. Learners consistently performed better when using the adaptive system. It may be related to the good match between the difficulty level of contents, the learners' abilities and preferences. It could also be due to improved learners' engagement with the system, as a result of solving their difficulties with adaptive guidance. From these findings it is clear that Rule-PAdel and semantic rule-based approach are capable of generating personalised e-learning systems which are more effective and appropriate.

Furthermore, the performance results indicate that in the following two situations the adaptive system was more beneficial than non-adaptive:

- Using for learning complex topics: According to the performance results, we find out that the learners who engaged with advanced materials needed more support from the adaptive system. It is clear that when the learners deal with complex material and exercises, they need more advice and supplementary content to solve their problems.
- Used by weaker learners: As shown in the performance results, personalised content and adaptive guidance was more helpful for weaker students than intellectually stronger ones. It is because they get informed on what their weaknesses are, receive supplementary contents and actively get involved in different stages for solving their weakness. They were also presented with appropriate materials based on their abilities.

6.5 Teacher Satisfaction

Teachers are responsible for controlling e-learning courses and integrating them as a part of the instructional curriculum they are responsible for. Two teachers involved in teaching

fractions along with their students have used the personalised learning system and were asked to comment on their experience. In general, the teachers were satisfied with the progress of their students when using the system. They mentioned in their comments that the system has increased students' motivation; students have spent more time in solving difficult learning problems which allowed them to complete their homework with more accuracy.

The teachers also felt that different students progressed differently when using the system. This is because the learners who worked with the adaptive system showed more progress than the ones who worked with the non-adaptive system. This problem would naturally get solved when all learners are engaged with the adaptive system.

Additionally, the teachers were pleased that the system supported different instructional plans across teaching a specific learning domain (i.e. Fraction). However, they had to author SWRL rules and work with ontologies to create different instructional plans. This was easy for some of the teachers who were familiar with this or similar technologies. However, some of the teachers needed an authoring tool for a simplified process of defining new instructional plans. This issue was also reflected when they wanted to modify adaptation techniques. Therefore, having appropriate authoring tools is one of the ways in which this system can be further improved in the future. However, the scope of developing such tools itself can be the topic of new research in the field of technologically inspired methods for teaching.

6.6 Flexibility and Extensibility of System

One of the main objectives of this research was to develop a flexible and extensible personalised e-learning system. In Rule-PAdel, the flexibility and extensibility have been considered from the following aspects:

1. The system can produce dynamically personalised learning experience for particular learners at run time based on learner's interaction with the system.
2. The system can generate different adaptive effects by using different components of adaptivity without impacting the implementation of the system.

This section evaluates the flexibility and extensibility of the system by examining both the ability of generating on-the-fly personalisation and generating different adaptive effects through modifying discrete models for components of adaptivity.

6.6.1 Flexibility through Generating On-the-fly Personalisation

Ontologically modelling knowledge, representing the domain, content, assessment and learner models using OWL ontology and establishing adaptation rules using SWRL, provide the basis for both description logic reasoning and rule-based inference power to generate runtime personalised learning. In order to examine whether the system generates suitable guidance based on learners' interaction with the system, the transactional log data of all learners' interactions with the systems were collected in the following two situations during the experiment:

- Learners' answers to exercises
- Recommendations generated by the system

Table 6.7 lists different adaptive guidance recommended by the system based on learners' responses to the exercises.

The adaptive guidance illustrated in Table 6.7, indicates that the system has sufficient flexibility, as it generates appropriate learning paths for learners at runtime based on their responses to the system.

6.6.2 Flexibility and Extensibility in Using Different Components of Personalisation

The semantic rule-based approach was designed to develop personalised e-learning systems which are flexible and extensible in generating different adaptive effects. These features make it easy to update and extend without affecting the system as a whole. This section presents the evaluation of flexibility and extensibility with regards to different extensions of Rule-PAdel.

Level of Exercises	Result of Exercises	level of previous learning content	Adaptive guidance
Easy	Pass	–	<i>Please do more exercises with higher difficulty level</i>
Easy	Fail	Moderate / High	<i>Please read a learning content with a lower difficulty (more additional explanation)</i>
Easy	Fail	Low	<i>Please repeat Prerequisite of this topic</i>
Moderate	Pass	–	<i>Please do more exercises with higher difficulty or participate in related assessment</i>
Moderate	Fail first time	–	<i>Please read More examples and do more exercises for better understanding</i>
Moderate	Fail second time	Moderate / High	<i>Please read a learning content with a lower difficulty (more additional explanation)</i>
Moderate	Fail second time	Low	<i>Please repeat the prerequisite of this topic</i>
Difficult	Pass	–	<i>Please continue to do more exercises or take the related assessment</i>
Difficult	Fail first time	–	<i>Please do more exercises for passing this level or take the related assessment</i>
Difficult	Fail second time	–	<i>Please read a learning content with a higher difficulty (more advanced and deeper information)</i>

Table 6.7: Different adaptive guidance

Flexibility and Extensibility in Using Different Adaptation Models

The key factor enabling the flexibility of the system in using different adaptation techniques is the abstraction provided by the ontological modelling enriched by SWRL rules. Through abstraction, adaptation techniques describe how concepts, not instances, are adapted. Rules do not describe any relationships in a particular domain, but embody a pure generic approach when selecting the most appropriate IO. Adaptation techniques do not embody any information about contents. This enables the modification of the adaptation techniques independently of the content. Through the separation of the content and adaptation techniques, instructional designers could manage the adaptive features of the e-learning system. They can modify only the adaptation rules without authoring or modifying the content.

In addition to knowledge abstraction, representing the adaptation decisions using SWRL rules provides explicit definition for all personalisation features in the system. These adaptation rules use data from the domain, content, assessment and learner model ontologies to perform personalisation. With this approach many of the functionalities behind the rules are hidden. Therefore, it simplifies modification of adaptation rules for instructional designers.

In order to examine whether our system satisfies this flexibility, first an instructional designer was asked to author the adaptation rules for the original version of the personalised Fraction learning system, which formed the basis of the learners' evaluations. In this system, the designer used VARK to model the recommendation component. Next, an extension of this system was built and another instructional designer was asked to replace the VARK learning style to Kolb by modifying the adaptation rules. This modification in the learning style model is successfully integrated with the rest of the system where both systems continued to work by using two different models of recommendations. This evaluation shows the flexibility, and extensibility of Rule-PAdel in using different adaptation techniques and the simplicity of updating based on the adaptation rules.

At the end of this evaluation, we interviewed the two instructional designers and asked their level of satisfaction with creating and modifying adaptation techniques. They were satisfied with this facility. However, one of the designers who was not familiar with the technologies felt that an authoring tool which abstracts away from the underlying logical

rules can simplify the authoring process, as mentioned earlier in Section 6.5.

Flexibility of System in Using Different Domain Models

The semantic rule-based approach supports flexibility in using different domain knowledge in the system as well as the adaptation techniques. This is facilitated by using appropriate domain ontologies as the sole source of knowledge about a topic. In other words, the system's adaptive engine does not contain any knowledge about a particular domain. In the case of our original system, it was using an ontology for the fraction domain. However, the system could be equally well applied in any other domain provided that an appropriate domain ontology model is made available.

To evaluate the domain flexibility of Rule-PAdel another extension of the system was implemented for teaching *Exponents and roots* domain. It should be mentioned that the domain which Rule-PAdel teaches could be different such as physics or chemistry. We have evaluated our system using maths as we worked closely with authors in this domain. Domain ontology was designed to formally describe subtopics of *Exponents and roots* and their pedagogical relations (e.g. prerequisite relation). The domain ontology can be found in APPENDIX E. Figures 6.8 and 6.9 show the annotated course structure (ACS) of the original system and the ACS of the system's extension respectively. Only a few IOs were created for the purpose of testing the system's extension. This shows that the system is flexible in using different domains across an adaptation model and instructional plans.

Flexibility of the System in Using Different Subjects

The semantic rule-based approach is fully domain independent. This is achieved by using appropriate domain ontologies as the system's only source of knowledge. The original implementation of Rule-PAdel is based on the ontology covering the *Fraction* topic in the *mathematics* domain. However, the system can be equally well applied in any other domain provided that an appropriate domain ontology is made available. In order to investigate the flexibility of Rule-PAdel in using other subjects, another extension of the system was implemented for teaching *Physical Processes* topic in the *science* domain. Domain ontology was designed to formally describe the subtopics of *Physical Processes*

The screenshot shows a web interface for a fraction topic. On the left, under 'Structure of Content', there is a tree of topics:

- Arithmetic Operation
- Percentage
- What is a fraction?
 - Concept of Fraction
 - Proper Fraction
 - Improper Fraction
 - Comparing Fraction
 - Comparing Two Fraction
 - With same denominator
 - Different denominator
 - Comparing More Than Two Fraction
 - Ordering Fraction
 - Equivalent Fraction
 - Concept of Equivalent Fraction
 - Simplifying Fraction
 - Adding Fraction
 - With same denominator
 - With different denominator
 - Subtracting Fraction
 - With same Denominator
 - With different Denominator
 - Multiplying Fractions

On the right, under 'Instruction', there is a text box:

Now you can select the subject you want to study.
- Note the colour of the links:
Purple: You already know the subject. You can Select these subjects to study them again.
Blue: These are the subjects which are suitable for you to learn right now.
Gray: You are not ready yet to study these.

Below this is a 'Go Ahead' button.

Figure 6.8: Annotated Course Structure of Fraction topic (version 1)

and the pedagogical relationships between them. A segment of the domain ontology is presented in Figure 6.10. Only by replacing the domain ontology without changing the implementation code the system properly works with the new subject.

Flexibility of System in Using Different Instructional Plan

In the previous sections we have discussed the flexibility of our system in using different adaptation techniques and the domain of knowledge. In addition to those, the system is also flexible in adopting different instructional plans in the content ontology across the same domain and adaptation model.

To evaluate the flexibility of the instructional plan two versions of the personalised fraction learning system were developed, each with the same materials but in teaching them in different ways. This facilitates the use of different types of instructional objects (IOs) with different sequencing IOs, using various instructional plans. In the original system, the instructional plan defined in the content ontology states that the system should always present a definition, three examples, three exercises and finally an assessment. Figure 6.11 shows a screenshot of the system which illustrates the sequence of types of IOs. In the second version, the instructional designer designed an alternative instructional plan. This new plan specifies that the system must present an example, an explanation about the

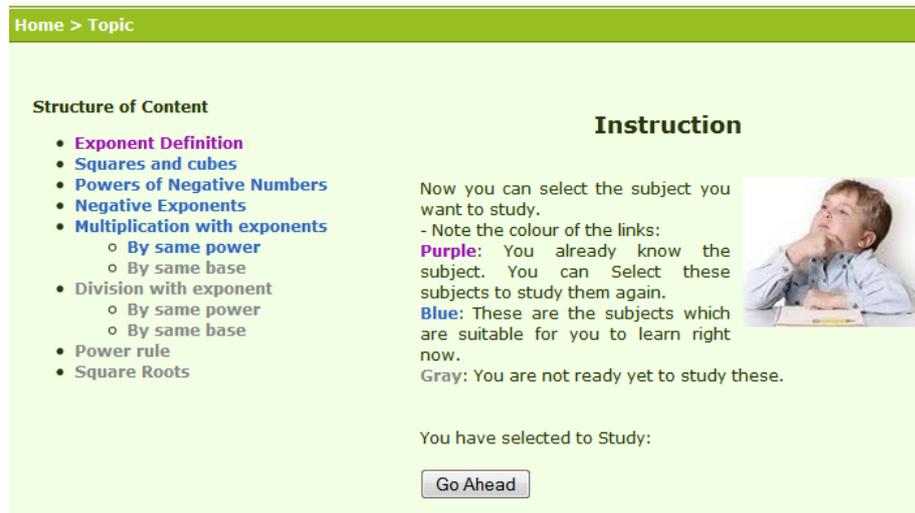


Figure 6.9: Annotated Course Structure of Exponents topic (Version 2)

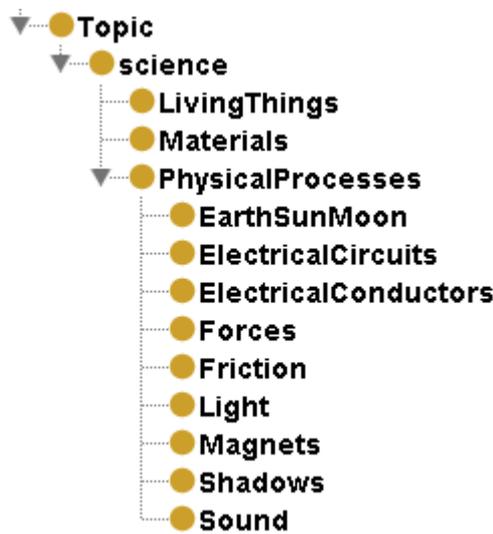


Figure 6.10: A segment of domain ontology for science subject

topic, an activity, an exercise and finally an assessment to the learners in the same order. A screenshot of the second version of the system is shown in Figure 6.12.

As it can be seen from the Figures 6.11 and 6.12 the system presented different IOs in different orders. These two perspectives of the same material are achieved only by changing the IO types and their order in the content ontology without the need to change anything from the internal implementation of the system.

Figure 6.11: Screenshot of the system with first instructional plan (version 1)

6.7 Reusability of the Components of Adaptivity

The reusability of adaptation techniques, content and instructional plans have been evaluated in different extensions of the personalised fraction learning system developed using the semantic rule-based approach. In the proposed approach, presenting the components of adaptivity by discrete ontologies is the primary driver in the reuse of its components. The explicit conceptualisation of the components of adaptivity in the form of ontologies facilitates knowledge reuse within a personalised learning system. Additionally, the representation of each of the components of adaptivity by a specific ontology results in a clear separation of these components (adaptation, content, assessment and domain). Adaptation model uses data from the aforementioned ontologies to perform personalisation. The separation of adaptation techniques, instructional plans and concept sequencing from a particular content enables the reusability for each of these components in future updates of the system or in other systems. This reuse is demonstrated in different extensions of Rule-PAdel.

The size of a piece of content has a significant impact on its reusability. A large piece of content is less possible to be reused than one that is smaller. This is due to the fact that the course authors are capable of including a discrete and small IO within a new personalised

Figure 6.12: Screenshot of the system with second instructional plan (version 1)

learning system.

Reuse of fine-grained content, concept sequencing and adaptation techniques were demonstrated through evolving different extensions of the *Fraction* learning system (See Section 6.6.2). The variations between different extensions were mainly differences in preferred instructional plans for teaching *Fraction*, expressed by the instructional designers. The fine granularity of the IOs supported easy reuse across each of these extensions. In the original system, the instructional plans defined that the learners should be presented by a *definition, three examples, three exercises and an assessment*, while the second extension *two examples, an explanation, an activity, an exercise and an assessment* were presented. Some of the created IOs in the original system such as examples, exercises and assessments were reused to implement the second extension, while the authors should have created new IOs such as activities for the new extension. The adaptation rules (defined in the adaptation model) and concepts sequencing (defined in the domain ontology) used in the original version were also reused in the second version. The new extension of the system worked perfectly with the newly created content. It is clear that the capability of content reusability is strongly dependent on the size of the content and its scope. Finer-grained pieces of content have more potential for being reused in different learning systems. This evaluation indicates that the content, concept sequencing and adaptation techniques were

reused properly in the second version of the system.

To evaluate the reusability of the instructional plans, the domain or the adaptation techniques should be changed, while reusing the same instructional plan in different versions of the system. Both of these evaluations were considered in Section 6.6.2 (where the adaptation model and the domain model are changed).

In general, from evaluating the reusability of the models of Rule-PAdel, some key points were found. The first is that ontological modelling is the primary factor for enabling the reuse of different components of adaptivity. Defining each component by a well-defined ontology facilitates a clear definition of discrete models which makes it possible to reuse any of them. SWRL Rules provide explicit definition of all adaptive activities in the system to encourage the reusability and modifiability of the adaptation rules. Moreover, ontology-based modelling supports describing the content in a fine-grain level which facilitates the reusability of the content.

6.8 Comparing Rule-PAdel with other similar Adaptive System

The next generation of e-learning systems need to provide greater flexibility, extensibility and reusability to support today's learning requirements. However, most of the current e-learning systems have performed poorly in these areas. In the approach proposed in this research, ontologies have been associated with reasoning mechanisms and rules to represent adaptation components in educational systems. In this section the flexibility, extensibility and reusability of Rule-PAdel are compared with other similar adaptive e-learning systems to validate the results presented for Rule-PAdel. This comparison is presented in Table 6.8.

As it can be seen in this table, some of the surveyed systems presented in this section use only ontologies to represent knowledge and others apply rule based reasoning as well. However, each of them use ontology and rule to represent a different section of all adaptation components. For example, curriculum content sequencing system applies rules to represent content sequencing. However, Protus2 applies rules to represent only the

Felder and Silverman learning style. The table shows that each system supports flexibility, extensibility and reusability partially.

Additionally, there are many other adaptive systems such as Topolor (Shi et al., 2013) which have their adaptive rules encoded in their implementation code. Therefore, given the effort in attaining this information, it is not reused.

System's Name	system	Using ontology	Using Rule	Flexibility , Extensibility and Reusability			
				Content ontology	Domain ontology	Adaptation ontology	Instructional ontology
Interbook	(Brusilovsky, 1996)	No	No	No	No	No	No
AHA	(De-Bra & Calvi, 1998)	No	No	No	Yes	No	No
TANGOW	(Carro et al., 2002)	No	No	Yes	No	No	No
ADAPT	(Brusilovsky et al., 2005)	Yes	No	No	Yes	No	No
TANGRAM	(Jovanović et al., 2009)	Yes	No	Yes	Yes	No	No
Curriculum Content Sequencing	(Chi, 2009)	Yes	Only for Curriculum Content	Yes	Yes	No	No
Modelling instructional design theories with ontologies	(Sicilia et al., 2011)	Yes	Only for instructional design	No	No	No	Yes
Protus 2.0	(Vesin et al., 2012)	Yes	Only for learning style	Yes	Yes	Only for learning style	No
Rule-PAdel	(Yarandi, 2013)	Yes	Yes	Yes	Yes	Yes	Yes

Table 6.8: The result of comparison Rule-PAdel with some other adaptive e-learning systems

6.9 Summary

This chapter has presented the result of evaluating the Rule-PAdel system. After an introduction, in Section 6.2 the methodology of evaluation was described. Then Section 6.3 and 6.4 discussed the evaluation of learners' satisfaction with the adaptive system and also the impact the system made on the student's learning process. The learner's evaluation shows that the learners were satisfied with the experience of using a personalised learning system which implements Rule-PAdel. It has been shown that learners were able to learn

significantly when working with the adaptive system. The results have indicated that this system significantly improves learners' knowledge gain when working with more complex learning content. It has also indicated that weaker students benefited the most from using an adaptive system. Section 6.5 presented the teachers' satisfaction results with using the system. Overall they were satisfied with using an adaptive system. They especially stated that the system promoted the learners' motivations and interests. The results of these two sections have indicated that the implemented adaptive personalised learning system was pedagogically effective and satisfactory for both learners and teachers which was one of the primary objectives of this research. The satisfaction of learners and teachers is a key result which shows the feasibility of the main posed research question in this thesis.

Section 6.6 has demonstrated the flexibility and extensibility of a personalised learning system generated using the semantic rule-based approach. The flexibility and extensibility of the system were considered from two perspectives. Firstly, we found out that the system is flexible as it dynamically generates appropriate learning paths for the learners based on their interaction with the system. Next, defining the components of adaptivity using separated ontologies and implementing adaptation techniques using SWRL rules enables the authors to add, delete or modify these components (e.g. adaptation model) without the need to change the implementation of the system. Through this facility, the system is flexible and extensible in using various adaptation techniques, domain knowledge and instructional plans.

In section 6.7 we have examined the reusability of the adaptation techniques, content, instructional plan and concept sequencing which are facilitated by the separation of the components of adaptivity. The result of this examination revealed that the discrete components of adaptivity could be reused in various personalised e-learning systems. The findings of these sections recognise some features that facilitate the reusability. The size of content has a key feature in reusability of content. Finer-grained content defined by ontologies is more reusable. Moreover, the explicit conceptualisation of components of adaptivity in the form of ontologies facilitates reusability of these components.

The outcomes of the last two sections have shown that the generated adaptive system was flexible and extensible and also demonstrated that the components of adaptivity are reusable across different personalised e-learning systems. These findings correspond to

another objective of this thesis.

In last section we compared Rule-PAdel with other similar adaptive e-learning systems.

Chapter 7

Conclusion and Future work

7.1 Introduction

In this thesis, a semantic rule-based approach is introduced for adaptive learning systems to offer a personalised learning experience for its users. In this approach, ontological modelling makes possible a clear separation of adaptivity components. Additionally, enriching ontologies with semantic rules increases the reasoning power and helps to represent adaptation decisions. This novel approach aims to improve flexibility, extensibility and reusability of learning systems, while offering pedagogically effective and satisfactory learning experiences for learners. Moreover, semantic rules facilitate runtime incorporation of discrete adaptivity components to generate flexible personalisation during the learning process.

In this chapter, we will revisit the objectives of this thesis and identify how these objectives were achieved. After that, we present contributions this research makes in the field of technology inspired learning and we conclude by highlighting potential directions for future works.

7.2 Achieved Objectives

The objective of this thesis was to support dynamic personalised learning which provides its users with sufficient flexibility from two aspects. Firstly, the system dynamically generates

flexible personalised learning experiences for different learners based on their progress. Secondly, it facilitates the development of personalised e-learning systems which support different adaptation techniques, instructional plans, concept sequencing and domain of knowledge without having to impact on the existing adaptive engine implementation. As it was stated in Chapter 1 in order to achieve this objective an innovative approach was developed for designing and implementing adaptive learning systems which separates out the modelling of the main adaptivity elements to offer a domain independent architecture for adaptive e-learning systems. Additionally we mentioned that the proposed approach enables the reusability and maintainability of the components for adaptivity.

At the heart of this thesis, the semantic rule-based approach is proposed. The proposed approach defines the components of adaptivity into separated models which are used by an independent adaptive engine. Accordingly, ontologies and rules are used as the basis of the proposed approach. Ontologies are used to represent a shared conceptualisation of the components of adaptivity (e.g. domain, learner, content, assessment and adaptation models). They allow specifying formally and explicitly the related concepts, their properties and their relationships. In addition to having sets of basic implicit reasoning mechanisms derived from the associated underlying description logic, ontologies are also enriched with rules to make further inferences and to express adaptation logic. Using rules also allows for additional expressivity for the representation formalism and reasoning on instances. For example, if a learning concept is selected to be learned an adaptation rule may be fired to choose instructional objects that are appropriate for teaching the concepts, from a pool of existing instructional objects. Typically, this decision is based on learner's abilities and preferences, implemented as ontological models.

The models and components of adaptivity which this thesis is focusing on are Instructional Objects (IOs), adaptation model rules (in the form of semantic rules), domain models (in the form of course structures and concepts sequencing), content models (in the form of structure of content and instructional plans), learner's and assessment models.

The most important module in our semantic rule-based approach is the Adaptation Model as it includes the adaptation techniques employed throughout the system. The adaptation model contains a set of rules that when using rule-based reasoning can produce personalised learning contents and recommendation for adaptive guidance tailored to the learner's

progress. Different conditions are modelled in the body of the rules. Through a review of existing related works we found out that many adaptive e-learning systems had their adaptation rules either entwined in their content model or in the system's business logic. The adaptation model is extensible in the sense that it enables instructional designers to easily modify and expand the adaptation techniques of the courses they author. Adaptation strategies are not included inside the adaptive engine. It is rather implemented as a separate model using ontologies enriched with semantic rules. Adaptation model, and in fact any other model, is reusable across diverse adaptive e-learning systems. The ontology-based modelling approach which we use facilitates this reusability through referring to the defined concepts in the ontologies rather than the instances.

The domain model describes how topics of a learning domain are structured. It presents all topics covered in the learning system and their pedagogical relations. It describes concept sequencing developed by domain experts to realise the order in which the topics are to be learned. The aim of a domain model is to represent the sequence of topics without referring to the content model. Separating domain and content models in their definition promotes system's flexibility and the reusability of its components as it enables course author(s) to modify the navigation and presentation adaptivity separately. For instance, if an author attempts to modify only the adaptive navigation of a course she only needs to change the domain model that implements it.

The content model ontology provides a common representation of the content. It specifies content classifications along with their semantic relationship in order to allow adaptive engine to select and compose appropriate IOs at runtime. In order to improve the flexibility of the system in generating personalised learning content, which is one of the objectives of this thesis, two factors have been considered in designing the content model. Firstly, learning content has been separated from the adaptation logic which results in the learning content being no longer specific to any given adaptation rule, or instructional plan. Secondly, the learning contents have been shaped in fine-granularity levels which we call Instructional Objects (IOs). Because of the small size of IOs, there is a higher chance for them to easily fit to different applications, making it easier to dynamically generate personalised learning content for different learners and therefore increase the reusability of existing IOs.

Assessment and evaluation of an individual's knowledge is a major pillar on which personalisation stands. Assessment is used to identify how much knowledge a learner has acquired during her interaction with the system and is used to personalise the learning curriculum and assign remedial problems in areas where the learner is weak. In our approach, an ontology-based assessment model is developed to formally represent relevant information about tests, specially the parameters used to calculate learners' ability based on the item response theory. It also enables annotating tests and their components semantically which facilitate its reuse in different assessments.

The learner model ontology presents personal information and learning characteristics of different learners who interact with the system and helps in deciding the correct teaching strategies for them. The main purpose of this ontology is to identify a user and to determine the best adaptation strategy and guidance based on her captured knowledge which is accessible from the user's profile.

The independent adaptive engine at the heart of the system provides the functionality for generating personalised learning experiences. Although it performs the necessary computations for personalisation, it does not contain the strategies or knowledge for any particular learning domain, concept sequencing or instructional plan. The related knowledge is all modelled and kept in their respective ontologies. This separation results in a system architecture that is highly modular thus having higher levels of abstraction.

The main achieved objectives of the Rule-PAdel system grounded on the proposed semantic rule-based approach are as follows:

- The ability to generate satisfactory and pedagogically suitable learning paths.
- To have separated components of adaptation.
- To be flexible and extensible in supporting different subject domains, instructional plans and adaptation techniques.
- To be accurate to adapt to the learner's current knowledge and abilities.
- To be flexible in dynamically recommending adaptive guidance for learners based on their needs and progress.

This thesis has achieved the initial objective of developing a novel approach for generating

a personalised e-learning system witnessed to what has been shown in Chapter 6. This chapter demonstrated that the generated personalised e-learning system is satisfactory and pedagogically sound for learners. It also showed that students were able to learn effectively when working with this system. This chapter also showed how separating knowledge models increased our system's flexibility and extensibility and allowed for the reusability of all of its adaptivity components. In this chapter, flexibility and extensibility of the system were evaluated from two perspectives. First, the trialling showed that the system is flexible in generating appropriate learning paths for learners based on their interactions with the system. Secondly, it showed that defining the components of adaptivity, using separated ontologies and implementing adaptation techniques using semantic rules, facilitates the generation of different adaptive effects based on different sets of models (e.g. adaptation model) without having an impact on the implementation of the system.

7.3 Contribution to Knowledge

By introducing our semantic rule-based approach for developing adaptive and personalised e-learning, this thesis made several contributions to the state of the art. Firstly, as part of our approach, different components of personalisation are presented using ontologies enriched with semantic rules. The ontology-based knowledge representation facilitates a clear separation of the components required for adaptivity. It also enables e-learning systems to carry out automated reasoning.

Through adding ontologies enriched with semantic rules we were able to provide explicit definition for all adaptation decisions. Semantic rules express relations that cannot be presented by ontologies in addition to enabling personalisation at runtime based on updated information. Moreover, semantic rules promote maintainability of the adaptation process as they separate out the adaptation logic from the core engine which allows authors (non-professional programmers) to develop adaptation techniques. This innovative approach provides sufficient flexibility and extensibility for adaptive systems over the adaptation process.

As part of our approach, an adaptive engine is designed to allow for fine-grained pieces of content to be dynamically assembled into self-contained units of personalised content

at runtime. The engine also recommends an adaptive guidance for learners based on analysing their responses to the delivered activities.

One of the key contributions of this work, which was to have a level of independence between the adaptive engine and the components of adaptivity, is achieved by having an ontological architecture. This independence allows the engine to work with different adaptation models and knowledge domains. Additionally the explicit conceptualisation of the components of adaptivity in the form of ontologies encourages its maintainability and reusability. The ontology-based definition of content in a fine-grained level promotes the flexibility of the system in producing effective personalisation. In this thesis the Item Response Theory is applied for calculating learner's abilities in order to provide accurate personalisation (Yarandi et al., 2011a).

7.4 Future Work

There are several directions towards which this research can be further extended and improved. The following areas are potentially worthwhile pursuing in the future:

- **Authoring Tools:** There are already tools for authors to create learning objects, but only a few of them allow the authors to create personalised courses. The semantic rule-based approach, proposed in this research, enables authors to reuse the existing components of adaptivity and create new adaptive learning experiences. However, they have to write SWRL rules and work with ontologies to create different instructional plans. This although seems easy for some of the teachers familiar with similar technologies, the rest needed an authoring tool for a simplified process for creating a new adaptive learning experience. Therefore, having appropriate authoring tools facilitates the process of reusing and creating components of adaptivity into a new personalised learning experience. The scope of developing such tools is beyond this research as there are many issues to consider when developing them. Moreover, representing knowledge with ontologies and semantic rules as well as supporting authors in searching and choosing appropriate adaptation strategies for a given domain allows for populating these strategies through an appropriate authoring tool.

- **Improving information dependency:** One of the significant contributions of a personalised e-learning system implementing the semantic rule-based approach is the personalisation it offers by rule-based reasoning at runtime. During the learning process, when a student interacts with the system and responds to an interactive IO, the reasoner reasons over the whole knowledge base in order to find out what should be the next step. In the future versions, the efficiency of the system can be improved through setting the information dependencies in a way that avoids the whole knowledge base from being re-reasoned on. Therefore, the system would work with large-scale knowledge bases in a reasonable time. For speeding up the reasoning process, it is necessary to change how knowledge is structured and represented properly. Moreover, the system can use incremental reasoning; reasoning over only the updated data in the ontology without having to perform all the reasoning steps from scratch.
- **Collaborated learning:** Another area of possible future work is facilitating adaptive learning that involves communication between students who collaborate towards reaching common objectives. Adaptive collaborated learning is one of the most meaningful ways to support individual learning. In our approach the learners have interactions with the system which in future works can enable them to engage in rich interactions with each other. Through these facilities, we can guide learners to perform either individual or collaborative learning. In semantic rule-based approach, all components of personalisation such as learner, content and domain were designed through ontological modelling thus the system knows about the learners' needs. For example, the learner model could help the system select learners with similar learning styles in a group so they could better understand each other. Additionally, the proposed approach enables the system to diagnose the difficulties of the learners and guide them to remediate their problems. Therefore, it will be possible to put the learners who can support one another in one group and to propose personalised collaborative activities that help its members to remediate their difficulties. In collaborative sessions, learners can interact by reflecting their opinions, articulating their reasonings and explaining their knowledge. Particularly active learners based on the felder-silverman learning style model (which is part of the learner model in our approach) learn better when performing group works and engaging in discussions.

- Further improving the current implementation: Although the learners were satisfied with the system in the evaluation process, there were some questions which they indicated that they are not as satisfied with as the other ones. From the learners' evaluation, we recognised that there are two main important parts of the system that can be improved: the personalised navigation (annotated course structure) and the user interface. On the personalised navigation, we have utilised different colours to annotate course structure. However, in future works we will be able to apply different visual cues such as different icons, font sizes and special characters for annotation to clarify the educational status of the content behind the links. Additionally, we can use other strategies of adaptive navigation (see Chapter 3) to help learners to find appropriate paths in their learning. On the user interface issue, the current system was only a prototype for *Onto-PADEL* which its main focus was on achieving personalisation and adaptability in learning new topics. Hence, in the future, there should be more effort on this side of the system. For instance, the system could enable learners to customise the user interface based on their preferences (e.g. colour, font, size and background). Furthermore, the interface could allow them to perform most of the possible actions using only the keyboard for the additional comfort. Moreover, the interface can visualise the progress of learners during the learning process in order to increase their awareness of how they are progressing in their learning process and improve their motivation and engagement in learning.

In addition to the above listed future works, the ultimate task in the future would be a long-term evaluation of the proposed approach by using it in different educational environments.

References

- ADL (2004), Sharable content object reference model (scorm), Technical report, Advanced Distributed Learning (ADL).
- Antoniou, G. & Harmelen, F. V. (2004), *Web Ontology Language: OWL*, Springer, pp. 67–92.
- Antoniou, G. & Harmelen, F. v. (2008), *A Semantic Web Primer*, Massachusetts Institute of Technology.
- Antoniou, G., Damásio, C. V., Grosz, B., Horrocks, I., Kifer, M., Maluszynski, J. & Patel-Schneider, P. F. (2005), Combining rules and ontologies. a survey, Deliverable I3-D3, REWERSE.
- Aroyo, L. & Mizoguchi, R. (2003), Process-aware authoring of web-based educational systems, in ‘international Workshop on Semantic Web and Web-based Education (SWWL03), in conjunction with CAISE03 Conference’, Velden, Austria, pp. 212–221.
- Aroyo, L., Dicheva, D. & Cristea, A. I. (2002), Ontological support for web courseware authoring, in ‘Proceedings of the 6th International Conference on Intelligent Tutoring Systems’, ITS ’02, Springer-Verlag, London, UK, UK, pp. 270–280.
- Ausubel, D., Novak, J. & Hanesian, H. (1978), *Educational Psychology: A Cognitive View*, Holt, Rinehart and Winston.
- Ausubel, D. P. (1960), ‘The use of advance organizers in the learning and retention of meaningful verbal material’, *Journal of Educational Psychology*. **51**(5),pp. 267–272.
- Baker, F. B. (1992), *Item Response theory: Parameter estimation techniques*, Marcel Dekker, New York.

- Baker, F. B. (2001), *The basics of item response theory*, second edn, ERIC clearing house on assessment and evaluation, Washington.
- Barker, P., Campbell, L. M. & Roberts, A. (2004), 'Ims meta-data best practice guide for ieee 1484.12.1-2002 standard for learning object metadata - version 1.3 final specification'.
- Baylari, A. & Montazer, G. (2009), 'Design a personalized e-learning system based on item response theory and artificial neural network approach', *Expert Systems with Applications* **36**(4),pp. 8013–8021.
- Beaumont, I. H. (1994), 'User modelling in the interactive anatomy tutoring system anatom-tutor', *User Modeling and User-Adapted Interaction* **4**,pp. 21–45.
- Bechhofer, S., Harmelen, F. v., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F. & Stein, L. A. (2004), Owl web ontology language reference, Technical report, The World Wide Web Consortium.
- Bechhofer, S., Horrocks, I., Goble, C. A. & Stevens, R. (2001), Oiled: A reasonable ontology editor for the semantic web, in 'Proceedings of the Joint German/Austrian Conference on AI: Advances in Artificial Intelligence', KI '01, Springer-Verlag, London, UK, UK, pp. 396–408.
- Bechhofer, S., Volz, R. & Lord, P. (2003), Cooking the semantic web with the owl api, in D. Fensel, K. Sycara & J. Mylopoulos, eds, 'The Semantic Web - ISWC 2003', Vol. 2870 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 659–675.
- Bechtel, W. & Graham, G. (1999), *A Companion to Cognitive Science*, Blackwell Companions to Philosophy, Wiley.
- Berlanga, A. J. & nalvo, F. J. G.-P. (2008), 'Learning design in adaptive educational hypermedia systems', *Journal of Universal Computer Science* **14**(22),pp. 3627–3647.
- Berners-Lee, T. (2006), 'Artificial intelligence and the semantic web: Aaai2006 keynote', PowerPoint slides.
- Berners-Lee, T., Hendler, J. & Lassila, O. (2001), 'The semantic web', *Scientific American* **284**(5),pp. 34–43.
- Böcker, H.-D. & Hohl, H. andSchwab, T. (1990), Hypadapter: Individualizing hypertext, in

- 'International Conference on Human-Computer Interaction (INTERACT)', Cambridge, U.K.
- Boyle, C. & Encarnacion, A. O. (1994), 'Metadoc: An adaptive hypertext reading system', *User Modeling and User-Adapted Interaction* **4**,pp. 1–19.
- Brewster, C. & O'Hara, K. (2007), 'Knowledge representation with ontologies: Present challengesfuture possibilities', *International Journal of Human-Computer Studies* **65**(7),pp. 563 – 568.
- Brooks, C., McCalla, G. & Winter, M. (2006), Flexible learning object metadata, in 'SW-EL05: Applications of Semantic Web Technologies for E-Learning Workshop'.
- Brusilovsky, P. (1996), 'Methods and techniques of adaptive hypermedia', *User Modeling and User Adapted Interaction* **6**(2-3),pp. 87–129.
- Brusilovsky, P. (1998), Adaptive educational systems on the world wide web: A review of available technologies, in '4th International Conference on Intelligent Tutoring Systems (ITS'98)', Vol. 1452, Springer.
- Brusilovsky, P. (1999), 'Adaptive and intelligent technologies for web-based education', *Special Issue on Intelligent Systems and Teleteaching, Knstliche Intelligenz* **4**,pp. 19–25.
- Brusilovsky, P. (2001a), Adaptive educational hypermedia, in 'Proceedings of 10th International PEG conference', Tampere, Finland, pp. 8–12.
- Brusilovsky, P. (2001b), 'Adaptive hypermedia', *User Modeling and User Adapted Interaction* **11**(1-2),pp. 87–110.
- Brusilovsky, P. (2007), *Adaptive Navigational Support*, Vol. 4321, Springer-Verlag, Berlin, Heidelberg, chapter 8, pp. 263–290.
- Brusilovsky, P. & Pesin, L. (1998), 'Adaptive navigation support in educational hypermedia: An evaluation of the isis-tutor', *Journal of Computing and Information Technology* **6**,pp. 27–38.
- Brusilovsky, P. & Peylo, C. (2003), 'Adaptive and intelligent web-based educational systems', *International Journal of Artificial Intelligence in Education* **13**,pp. 156–169.

- Brusilovsky, P., Durlach, P. J. & Lesgold, A. M. (2012), *Adaptive Hypermedia for Education and Training*, Cambridge University Press, chapter 3, pp. 46–66.
- Brusilovsky, P., Eklund, J. & Schwarz, E. (1998), Web-based education for all: A tool for developing adaptive courseware, in ‘Proceedings of the Seventh International World Wide Web Conference’, Vol. 30, pp. 291–300.
- Brusilovsky, P., Schwarz, E. & Weber, G. (1996), Elm-art: An intelligent tutoring system on the world wide web, in ‘In Proceedings of the Third International Conference, ITS’.
- Brusilovsky, P., Sosnovsky, S. & Yudelson, M. (2005), Ontology-based framework for user model interoperability in distributed learning environments, in ‘Proceedings of World Conference on ELearning (E-Learn 2005)’, Vancouver, Canada, pp. 2851–2855.
- Bunt, A., Carenini, G. & Conati, C. (2007), *The adaptive web*, Springer-Verlag, Berlin, Heidelberg, chapter Adaptive content presentation for the web, pp. 409–432.
- Burstein, M. H., Hobbs, J. R., Lassila, O., Martin, D., McDermott, D. V., McIlraith, S. A., Narayanan, S., Paolucci, M., Payne, T. R. & Sycara, K. P. (2002), Daml-s: Web service description for the semantic web, in ‘Proceedings of the First International Semantic Web Conference on The Semantic Web’, ISWC ’02, Springer-Verlag, London, UK, UK, pp. 348–363.
- Carro, R., Pulido, E. & Rodriguez, P. (2002), Adaptive internet-based learning with the tangow system, in M. Ortega & J. Bravo, eds, ‘Computers and Education in the 21st Century’, Springer Netherlands, pp. 127–135.
- Chandrasekaran, B., Josephson, J. R. & Benjamins, V. R. (1999), ‘What are ontologies, and why do we need them?’, *IEEE Intelligent Systems* **14**(1),pp. 20–26.
- Chellatamilan, T. & Suresh, R. M. (2011), ‘Intelligent agents for the semantic adaptive e-learning system’, *International Journal of Computer Applications ICACT*(1),pp. 1–5. Published by Foundation of Computer Science, New York, USA.
- Chen, C.-M. (2008), ‘Intelligent web-based learning system with personalized learning path guidance’, *Computers & Education* **51**(2),pp. 787–814.
- Chen, C.-M. & Duh, L.-J. (2008), ‘Personalized web-based tutoring system based on fuzzy item response theory’, *Expert Systems with Applications* **34**(4),pp. 2298–2315.

- Chen, C.-M., Lee, H.-M. & Chen, Y.-H. (2005), 'Personalised e-learning system using item response theory', *Computers & Education* **44**(3),pp. 237 – 255.
- Chen, C.-M., Liu, C.-Y. & Chang, M.-H. (2006), 'Personalized curriculum sequencing utilizing modified item response theory for web-based instruction', *Expert Systems with Applications* **30**(2),pp. 1378–396.
- Chen, I. (2009), *Behaviorism and Developments in Instructional Design and Technology*, IGI Global, Hershey, PA, USA, chapter 23, pp. 153–172.
- Chi, Y.-L. (2009), 'Ontology-based curriculum content sequencing system with semantic rules', *Expert Systems with Applications* **36**(4),pp. 7838 – 7847.
- Chinnici, R., Moreau, J.-J., Ryman, A. & Weerawarana, S. (2007), Web services description language (wsdl) version 2.0 part 1: Core language, Technical report, W3C.
- Conlan, O. (2000), Novel components for supporting adaptivity in education systems - model-based integration approach, in 'Proceedings of the eighth ACM international conference on Multimedia', New York, NY, USA.
- Conlan, O. (2006), The multi-model, metadata driven approach to personalised elearning services, PhD thesis, Department of Computer Science, University of Dublin, Trinity College.
- Conlan, O., Dagger, D. & Wade, V. (2002), Towards a standards-based approach to e-learning personalization using reusable learning objects, in 'Proc. of World Conference on E-Learning, E-Learn 2002', pp. 210–217.
- Craik, F. I. & Lockhart, R. S. (1972), 'Levels of processing: A framework for memory research', *Journal of Verbal Learning and Verbal Behavior* **11**(6),pp. 671 – 684.
- DCMI (2012), 'Dublin core metadata element set, version 1.1'.
- De-Bra, P. & Calvi, L. (1998), Aha: a generic adaptive hypermedia system, in 'Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia', pp. 5–22.
- De Bra, P., Aerts, A., Berden, B., de Lange, B., Rousseau, B., Santic, T., Smits, D. & Stash, N. (2003), Aha! the adaptive hypermedia architecture, in 'Proceedings of the fourteenth ACM conference on Hypertext and hypermedia', HyperText '03, ACM, New York, NY, USA, pp. 81–84.

- Desai, N., Mallya, A. U., Chopra, A. K. & Singh, M. P. (2005), Owl-p: A methodology for business process development, *in* 'Agent-Oriented Information Systems 2005', pp. 79–94.
- Dolog, P. (2006), *Knowledge Representation and Reasoning in Personalized Web-Based e-Learning Applications*, VSB-Technical University of Ostrava.
- Dolog, P. & Nejdl, W. (2007), The adaptive web, Springer-Verlag, Berlin, Heidelberg, chapter Semantic web technologies for the adaptive web, pp. 697–719.
- Dolog, P., Henze, N., Nejdl, W. & Sintek, M. (2004), Personalization in distributed e-learning environments, *in* 'WWW Alt. '04 Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters', ACM Press, pp. 170–179.
- Duval, E. (2002), 'Ieee standard for learning object metadata (1484.12.1)'.
- Dwyer, K. K. (1998), 'Communication apprehension and learning style preference: Correlations and implications for teaching', *Communication Education* **47**(2),pp. 137–150.
- Farance, f. (2001), Draft standard for learning technology public and private information (papi) for learners (papi learner), Technical report, IEEE Learning Technology Standardization Committee (LTSC).
- Felder, R. M. & Silverman, L. K. (1988), 'Learning and teaching styles in engineering education', *English Education* **78**(7),pp. 674–681.
- Felder, R. M. & Soloman, B. A. (1991), 'The index of learning styles (ils)'.
- Fensel, D., Horrocks, I., Harmelen, F. v., Decker, S., Erdmann, M. & Klein, M. C. A. (2000), Oil in a nutshell, *in* 'Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management', EKAW '00, Springer-Verlag, London, UK, UK, pp. 1–16.
- Fink, J., Kobsa, A. & Nill, A. (1996), User-oriented adaptivity and adaptability in the avanti project, *in* 'In Designing For The Web: Emprical Studies'.
- Fleming, N. (1987), 'Teaching and learning styles, vark strategies'.
- Fleming, N. D. & Mills, C. (1992), 'Not another inventory, rather a catalyst for reflection', *To Improve the Academy* **11**,pp. 137–146.

- Garrett, J. J. (2005), 'Ajax: A new approach to web applications'.
- Gašević, D., Jovanović, J. & Devedžić, V. (2007), 'Ontology-based annotation of learning object content', *Interactive Learning Environments* **15**(1),pp. 1–26.
- Gašević, D., Jovanović, J., Knight, C. & Richards, G. (2008), 'Ontologies for effective use of context in e-learning settings', *The International Review of Research in Open and Distance Learning*.
- Ghazal, M. A., Yusof, M. M. & Zin, N. A. M. (2011), Adaptive educational hypermedia system using cognitive style approach: Challenges and opportunities, in '2011 International Conference on Electrical Engineering and Informatics', Bandung, Indonesia, pp. 1–6.
- Glaserfeld, v. E. (1995), *A constructivist approach to teaching*, Erlbaum, Hillsdale, NJ, pp. 3–15.
- Golemati, M., Katifori, A., Vassilakis, C., Lepouras, G. & Halatsis, C. (2007), Creating an ontology for the user profile: Method and applications, in 'In Proceedings of the First International Conference on Research Challenges in Information Science (RCIS)'.
- Grigoriadou, M. & Papanikolaou, K. (2000), 'Learning environments on the web: The pedagogical role of the educational material', *Themes in Education* **1**(2),pp. 145–161.
- Gruber, T. R. (1993), 'A translation approach to portable ontology specification', *Knowledge Acquisition* **5**(2),pp. 199–220.
- Guzman, E. & Conejo, R. (2005), 'Self-assessment in a feasible, adaptive web-based testing system', *Education, IEEE Transactions on* **48**(4),pp. 688–695.
- Haarslev, V. & Möller, R. (2001), Racer system description, in 'Proceedings of the First International Joint Conference on Automated Reasoning', IJCAR '01, Springer-Verlag, London, UK, UK, pp. 701–706.
- Hambleton, R. K., Swaminathan, H. & Rogers, H. J. (1991), *Fundamentals of Item Response Theory*, Sage publications, the International Professional Publishers, London.
- Hamdi, M. S. (2007), 'Masacad: A multi-agent approach to information customization for the purpose of academic advising of students', *Applied Soft Computing* **7**(3),pp. 746 – 771.

- Henze, N. & Nejd, W. (2000), Extendible adaptive hypermedia courseware: Integrating different courses and web material, *in* 'Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems', AH '00, Springer-Verlag, London, UK, UK, pp. 109–120.
- Henze, N., Dolog, P. & Nejd, W. (2004), 'Reasoning and ontologies for personalized e-learning in the semantic web', *Educational Technology & Society* **7**(4),pp. 82–97.
- Henze, N., Nejd, W. & Wolpers, M. (1999), Modeling constructivist teaching functionality and structure in the kbs hyperbook system, *in* 'Proceedings of the 1999 conference on Computer support for collaborative learning', CSCL '99, International Society of the Learning Sciences.
- Horridge, M. & Bechhofer, S. (2011), 'The owl api: A java api for owl ontologies', *Semantic Web* **2**(1),pp. 11–21.
- Horrocks, I. (2002), 'Daml+oil: a description logic for the semantic web', *IEEE Data Eng. Bull.* **25**(1),pp. 4–9.
- Horrocks, I. (2008), 'Ontologies and the semantic web', *Communications of the ACM* **51**(12),pp. 58–67.
- Horrocks, I., Patel-Schneider, P. F. & van Harmelen, F. (2003), 'From shiq and rdf to owl: The making of a web ontology language', *Journal of Web Semantics* **1**(1),pp. 7–26.
- Horrocks, I., Patel-Schneider, P. F., Bechhofer, S. & Tsarkov, D. (2005), 'Owl rules: A proposal and prototype implementation', *Web Semant.* **3**(1),pp. 23–40.
- Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosz, B. & Dean, M. (2004), Swrl: A semantic web rule language combining owl and ruleml, W3c member submission, World Wide Web Consortium.
- Horrocks, I., Patel-Schneider, P. F., McGuinness, D. L. & Welty, C. A. (2007), Owl: a description logic based ontology language for the semantic web, *in* F. Baader, D. Calvanese, D. McGuinness, D. Nardi & P. F. Patel-Schneider, eds, 'The Description Logic Handbook: Theory, Implementation, and Applications (2nd Edition)', Cambridge University Press, chapter 14, pp. 458–486.
- Hosseini, S. A., Tawil, A.-R. H., Jahankhani, H. & Yarandi, M. (2013), 'Towards an

- ontological learners modelling approach for personalised e-learning’, *International Journal of Emerging Technologies in Learning (iJET)* **8**(2),pp. 4–10.
- Huang, M.-J., Huang, H.-S. & Chen, M.-Y. (2007), ‘Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach’, *Expert Systems with Applications* **33**(3),pp. 551 – 564.
- Idris, N., Yusof, N. & Saad, P. (2009), ‘Adaptive course sequencing for personalization of learning path using neural network’, *International Journal of Advances in Soft Computing and Its Applications* **1**(1),pp. 49–61.
- Illeris, K. (2004), *The three dimensions of learning*, Krieger Pub. Co.
- Jeong, H.-Y., Choi, C.-R. & Song, Y.-J. (2012), ‘Personalized learning course planner with e-learning dss using user profile’, *Expert Systems with Applications* **39**(3),pp. 2567–2577.
- Jia, H., Wang, M., Ran, W., Yang, S. J. H., Liao, J. & Chiu, D. K. W. (2011), ‘Design of a performance-oriented workplace e-learning system using ontology’, *Expert Syst. Appl.* **38**(4),pp. 3372–3382.
- Jovanović, J., Gašević, D. & Devedžić, V. (2006), ‘Ontology-based automatic annotation of learning content’, *Int. J. Semantic Web Informatin System* **2**(2),pp. 91–119.
- Jovanović, J., Gašević, D. & Devedžić, V. (2009), ‘Tangram for personalized learning using the semantic web technologies’, *Journal of Emerging Technologies in Web Intelligence* **1**(1),pp. 6–21.
- Kabassi, K. & Virvou, M. (2004), ‘Personalised adult e-training on computer use based on multiple attribute decision making’, *Interacting with Computers* **16**(1),pp. 115 – 132.
- Kaplan, C., Fenwick, J. & Chen, J. (1993), ‘Adaptive hypertext navigation based on user goals and context’, *User Modeling and User-Adapted Interaction* **3**(3),pp. 193–220.
- Karp, P. D., Chaudhri, V. K. & Thomere, J. (1999), Xol: An xml-based ontology exchange language, Technical report, SRI International.
- Kay, J. & R.J.Kummerfeld (1994), Adaptive hypertext for individualised instruction, in ‘Workshop on Adaptive Hypertext and Hypermedia at Fourth International Conference on User Modeling’.

- Keefe, J. W. (1979), *Learning Style: An Overview*, National Association of Secondary School Principals.
- Kifer, M., Lausen, G. & Wu, J. (1995), 'Logical foundations of object-oriented and frame-based languages', *J. ACM* **42**(4), pp. 741–843.
- Klašnja-Milićević, A., Vesin, B., Ivanović, M. & Budimac, Z. (2010), 'E-learning personalization based on hybrid recommendation strategy and learning style identification', *Computers & Education* **56**(3), pp. 885–899.
- Kolb, D. A. (1984), *Experiential learning: experience as the source of learning and development*, Prentice Hall, Inc., United State of America.
- Krötzsch, M., Simančík, F. & Horrocks, I. (2012), 'A description logic primer', *CoRR*.
- Likert, R. (1932), *A technique for the measurement of attitudes*, Archives of Psychology, New York.
- Liu, Z., Liu, L., Kang, H., Zhong, S. & Jia, B. (2009), An ontology-based method of adaptive learning, in 'Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC', NCM '09, IEEE Computer Society, Washington, DC, USA, pp. 1586–1591.
- Math, N. & Chen, J. (1996), 'User-centered indexing for adaptive information access', *User Modeling and User-Adapted Interaction* **6**, pp. 225–261.
- McBride, B. (2002), 'Jena: A semantic web toolkit', *IEEE Internet Computing* **6**(6), pp. 55–59.
- mcbride, B. (2004), The resource description framework (RDF) and its vocabulary description language RDFS, in S. Staab & R. Studer, eds, 'Handbook on Ontologies', International Handbooks on Information Systems, Springer, pp. 51–66.
- Meccawy, M. (2009), A service-orientated architecture for adaptive and collaborative e-learning systems, PhD thesis, School of Computer Science, University of Nottingham.
- Medić, A. & Golubović, A. (2010), 'Making secure semantic web', *Universal Journal of Computer Science and Engineering Technology* **1**(2), pp. 99–104.
- Milosevic, D., Brkovic, M. & Sendelj, R. (2006), Ontology-based learner modeling system

- for web-based education, *in* 'Proceedings of the 5th WSEAS International Conference on E-ACTIVITIES', Venice, Italy, pp. 394–398.
- Mitra, N. & Lafon, Y. (2007), Soap version 1.2 part 0: Primer, Technical report, W3C.
- Mödritscher, F., García, V. M. B. & Gütl, C. (2004), Enhancement of scorm to support adaptive e-learning within the scope of the research project adele, *in* J. Nall & R. Robson, eds, 'Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2004', AACE, Washington, DC, USA, pp. 2499–2505.
- Mohan, P. & Brooks, C. (2003), Learning objects on the semantic web, *in* 'Advanced Learning Technologies, 2003. Proceedings. The 3rd IEEE International Conference on', pp. 195–199.
- Morimoto, Y., Ueno, M., Kikukawa, I., Yokoyama, S. & Miyadera, Y. (2007), Salms: Scorm-compliant adaptive lms, *in* T. B. S. Carliner, ed., 'Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education', AACE, Chesapeake, VA, pp. 7287–7296.
- Muijs, D. & Reynolds, D. (2005), *Effective Teaching: Evidence and Practice*, SAGE Publications.
- Müller, H.-M., Kenny, E. E. & Sternberg, P. W. (2004), 'Textpresso: An ontology-based information retrieval and extraction system for biological literature', *PLoS Biol* 2(11),pp. e309.
- Mulwa, C., Lawless, S., Sharp, M., Arnedillo-Sanchez, I. & Wade, V. (2010), Adaptive educational hypermedia systems in technology enhanced learning: a literature review, *in* 'Proceedings of the 2010 ACM conference on Information technology education', SIGITE '10, ACM, New York, NY, USA, pp. 73–84.
- Necib, C. & Freytag, J.-C. (2003), Ontology based query processing in database management systems, *in* R. Meersman, Z. Tari & D. Schmidt, eds, 'On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE', Vol. 2888 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 839–857.
- Noy, N. F. & McGuinness, D. L. (2001), *Ontology development 101: A guide to creat-*

- ing your first ontology, Technical Report KSL-01-05, Stanford Knowledge Systems Laboratory.
- Noy, N. F. & Musen, M. A. (2000), Prompt: Algorithm and tool for automated ontology merging and alignment, *in* 'Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence', AAAI Press, pp. 450–455.
- Ormrod, J. E. (1995), *Human Learning*, Merrill.
- Ouksel, A. M. & Sheth, A. (1999), 'Semantic interoperability in global information systems', *SIGMOD Rec.* **28**(1),pp. 5–12.
- Pahl, C. & Holohan, E. (2009), 'Applications of semantic web technology to support learning content development', *Interdisciplinary Journal of E-Learning and Learning Objects* **5**,pp. 1–25.
- Patel-Schneider, P. F., Hayes, P. & Horrocks, I. (2004), Owl web ontology language semantics and abstract syntax, Technical report, W3C Recommendation.
- Piaget, J. (1985), *The Equilibration of Cognitive Structures: The Central Problem of Intellectual Development*, University of Chicago Press.
- Popescu, E., Badica, C. & Trigano, P. (2007), Rules for learner modeling and adaptation provisioning in an educational hypermedia system, *in* 'Symbolic and Numeric Algorithms for Scientific Computing, 2007. SYNASC. International Symposium on', pp. 492–499.
- RDF (2004), 'Rdf vocabulary description language 1.0: Rdf schema', W3C.
- Rey-lópez, M., Brusilovsky, P., Meccawy, M., Díaz-redondo, R., Fernández-vilas, A. & Ashman, H. (2008), 'Resolving the problem of intelligent learning content in learning management systems', *International Journal on E-Learning* **7**(3),pp. 363–381.
- RuleML (2012), 'The rule markup initiative'.
- Sanginetto, E. (2008), *An Adaptive E-Learning Platform for Personalized Course Generation*, Idea Group Inc, Hershey, USA, pp. 262–281.
- Schlupe, S., Bettoni, M. & Schär, S. G. (2006), 'Modularization and structured markup

- for learning content in an academic environment’, *International Journal on E-Learning* **5**(1),pp. 35–44.
- Seeberg, C., Reichenberger, K., Fischer, S., Steinmetz, R. & Steinacker, A. (1999), Dynamically generated tables of contents as guided tours in adaptive hypermedia systems, in B. Collis & R. Oliver, eds, ‘Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 1999’, AACE, Chesapeake, VA, pp. 640–645.
- Seng, J.-L. & Kong, I. (2009), ‘A schema and ontology-aided intelligent information integration’, *Expert Systems with Applications* **36**(7),pp. 10538 – 10550.
- Sheth, A., Bertram, C., Avant, D., Hammond, B., Kochut, K. & Warke, Y. (2002), ‘Managing semantic content for the web’, *IEEE Internet Computing* **6**(4),pp. 80–87.
- Shi, L., Gkotsis, G., Stepanyan, K., Al Qudah, D. & Cristea, A. (2013), Social personalized adaptive e-learning environment: Topolor - implementation and evaluation, in H. Lane, K. Yacef, J. Mostow & P. Pavlik, eds, ‘Artificial Intelligence in Education’, Vol. 7926 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 708–711.
- Sicilia, M.-A., Lytras, M. D., Sánchez-Alonso, S., García-Barriocanal, E. & Zapata-Ros, M. (2011), ‘Modeling instructional-design theories with ontologies: Using methods to check, generate and search learning designs’, *Computers in Human Behavior* **27**(4),pp. 1389 – 1398.
- Sintek, M. & Decker, S. (2002), Triple-a query, inference, and transformation language for the semantic web, in I. Horrocks & J. Hendler, eds, ‘The Semantic Web ISWC 2002’, Vol. 2342 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 364–378.
- Sirin, E. & Parsia, B. (2004), Pellet: An owl dl reasoner, in ‘Proceedings of the 2004 International Workshop on Description Logics (DL2004)’, Vol. 104, CEUR-WS.org, Whistler, British Columbia, Canada.
- Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A. & Katz, Y. (2007), ‘Pellet: A practical owl-dl reasoner’, *Web Semant.* **5**(2),pp. 51–53.
- Smith, B. (2003), ontology, in L. Floridi, ed., ‘The Blackwell Guide to the Philosophy of Computing and Information’, Oxford: Blackwell, pp. 155–166.

- Smythe, C. (2005), 'Ims learner information package summary of changes'.
- Smythe, C. & Jackl, A. (2004), 'Ims content packaging information model - version 1.1.4 final specification'.
- Snae, C. & Brüeckner, M. (2007), 'Ontology-driven e-learning system based on roles and activities for thai learning environment', *Interdisciplinary Journal of E-Learning and Learning Objects* **3**,pp. 1–17.
- Sosnovsky, S. (2011), *Ontology-Based Open-Corpus Personalization for E-Learning*, PhD thesis, Informmation Science University of Pittsburgh.
- Stern, M. & Woolf, B. P. (2000), Adaptive content in an online lecture system, *in* 'Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems', AH '00, Springer-Verlag, London, UK, pp. 227–238.
- Studer, R., Benjamins, V. R. & Fensel, D. (1998), 'Knowledge engineering: Principles and methods', *Data Knowl. Eng.* **25**(1-2),pp. 161–197.
- Sure, Y., Erdmann, M., Angele, J., Staab, S., Studer, R. & Wenke, D. (2002), *Ontoedit: Collaborative ontology development for the semantic web*, *in* 'Proceedings of the First International Semantic Web Conference on The Semantic Web', ISWC '02, Springer-Verlag, London, UK, pp. 221–235.
- Trella, M., Carmona, C. & Conejo, R. (2005), *Medea: an open service-based learning platform for developing intelligent educational systems for the web*, *in* '12th International Conference on Artificial Intelligence in Education (AIED 2005)', pp. 27–34.
- Tsarkov, D. & Horrocks, I. (2006), *Fact++ description logic reasoner: system description*, *in* 'Proceedings of the Third international joint conference on Automated Reasoning', IJCAR'06, Springer-Verlag, Berlin, Heidelberg, pp. 292–297.
- Tserendorj, T. (2010), *Approximate Assertional Reasoning Over Expressive Ontologies*, PhD thesis, Fakultät für Wirtschaftswissenschaften, Karlsruher Institut für Technologie (KIT).
- Ullrich, C. (2005), 'The learning-resource-type is dead, long live the learning-resource-type!', *Learning Objects and Learning Designs* **1**(1),pp. 7–15.

- Vassileva, D. & Bontchev, B. (2009), Adaptation engine construction based on formal rules, *in* 'CSEDU (1)'09', pp. 327–332.
- Vavoula, G., Pachler, N. & Kukulska-Hulme, A. (2009), *Researching mobile learning: Frameworks, tools and research designs*, Peter Lang, Oxford.
- Verbert, K. & Duval, E. (2008), 'Alocom: a generic content model for learning objects', *International Journal on Digital Libraries* **9**(1), pp. 41–63.
- Vesin, B., Ivanović, M., Klačnja-Milićević, A. & Budimac, Z. (2011), Rule-based reasoning for altering pattern navigation in programming tutoring system, *in* 'System Theory, Control, and Computing (ICSTCC), 2011 15th International Conference on', Sinaia, pp. 1–6.
- Vesin, B., Ivanović, M., Milićević, A. K. & Budimac, Z. (2012), 'Protus 2.0: Ontology-based semantic recommendation in programming tutoring system', *Expert Syst. Appl.* **39**(15), pp. 12229–12246.
- Wainer, H., Dorans, N. J., Flaugher, R., Green, B. F. & Mislevy, R. J. (2000), *Computerized Adaptive Testing: A Primer*, second edn, Routledge Taylor & Francis Group.
- Wang, E. & Kim, Y. S. (2006), A teaching strategies engine using translation from swrl to jess, *in* 'Proceedings of the 8th international conference on Intelligent Tutoring Systems', ITS'06, Springer-Verlag, Berlin, Heidelberg, pp. 51–60.
- Wang, F.-H. (2006), Application of componential irt model for diagnostic test in a standard conformant elearning system, *in* 'ICALT', Kerkrade, pp. 237–241.
- Watson, W. R., Lee, S. & Reigeluth, C. M. (2007), Learning management systems: An overview and roadmap of the systematic application of computers in education, *in* F. M. Neto & F. V. Brasileiro, eds, 'Advances in Computer-Supported Learning', IGI Global, Hershey, PA, USA, chapter four, pp. 66–96.
- Yarandi, M. (2013), Semantic Rule-based Approach for Supporting Personalised Adaptive E-Learning, PhD thesis, School of ARCHITECTURE, COMPUTING & ENGINEERING.
- Yarandi, M., Jahankhani, H. & Tawil, A.-R. (2012a), An ontology-based adaptive mo-

- bile learning system based on learners' abilities, in 'Global Engineering Education Conference (EDUCON), 2012 IEEE', IEEE, Marrakesh, pp. 1–8.
- Yarandi, M., Jahankhani, H. & Tawil, A.-R. H. (2012b), An adaptive e-learning decision support system, in 'Interactive Collaborative Learning (ICL), 2012 15th International Conference on', Villach, pp. 1–5.
- Yarandi, M., Jahankhani, H. & Tawil, A.-R. H. (2013), 'Towards adaptive e-learning using decision support systems', *International Journal of Emerging Technologies in Learning (iJET)* **8**(S1),pp. 44–51.
- Yarandi, M., Jahankhani, H., Dastbaz, M. & Tawil, A. R. (2011a), Personalised mobile learning system based on item response theory, in '6th Annual Conference on Advances in Computing and Technology', London, UK, pp. 108–115.
- Yarandi, M., Tawil, A. R. & Jahankhani, H. (2012c), Ontologies for personalised adaptive learning, in 'Advances in Computing & Technology 2012', University of East London, London.
- Yarandi, M., Tawil, A.-R. H. & Jahankhani, H. (2011b), Adaptive e-learning system using ontology, in 'DEXA Workshops', Toulouse, pp. 511–516.
- Yarandi, M., Tawil, A.-R. H., Jahankhani, H. & Hosseini, S. A. (2012d), Ontology-based learner modelling for supporting personalised e-learning, in 'Interactive Mobile and Computer Aided Learning (IMCL), 2012 International Conference on', Amman, pp. 113–118.
- Yu, C. H. (2007), 'A simple guide to the item response theory'.

Appendix A

Publications

A.1 Journal and Conference Publications

Yarandi, M., Jahankhani, H., and Tawil, A.-R. H. (2013). Towards adaptive e-learning using decision support systems. *International Journal of Emerging Technologies in Learning (iJET)*, 8(S1):44-51.

Hosseini, S. A., Tawil, A.-R. H., Jahankhani, H., and Yarandi, M. (2013). Towards an ontological learners modelling approach for personalised e-learning. *International Journal of Emerging Technologies in Learning (iJET)*, 8(2):4-10.

Yarandi, M., Jahankhani, H., and Tawil, A.-R. H. (2012b). An adaptive e- learning decision support system. In *Interactive Collaborative Learning (ICL), 2012 15th International Conference on*, pages 1-5, Villach.

Yarandi, M., Jahankhani, H., and Tawil, A.-R. (2012a). An ontology-based adaptive mobile learning system based on learners' abilities. In *Global Engineering Education Conference (EDUCON), 2012 IEEE*, pages 1-8, Marrakesh. IEEE.

Yarandi, M., Tawil, A.-R. H., Jahankhani, H., and Hosseini, S. A. (2012e). Ontology-based learner modelling for supporting personalised e-learning. In *Interactive Mobile and Computer Aided Learning (IMCL), 2012 International Conference on*, pages 113-118, Amman.

Yarandi, M., Tawil, A.-R. H., and Jahankhani, H. (2011b). Adaptive e-learning system

using ontology. In DEXA Workshops, pages 511-516, Toulouse.

Yarandi, M., Tawil, A. R., and Jahankhani, H. (2012c). Ontologies for personalised adaptive learning. In *Advances in Computing and Technology 2012*, London. University of East London.

Yarandi, M., Jahankhani, H., Dastbaz, M., and Tawil, A. R. (2011a). Personalised mobile learning system based on item response theory. In *6th Annual Conference on Advances in Computing and Technology*, pages 108-115, London, UK. *Awarded the best Paper Presentation.*

Yarandi, M., Tawil, A.-R. H., and Jahankhani, H. (2013). An Ontological Account of Adaptive E-Learning. *Journal of Systems and Software. (In process).*

Yarandi, M., Jahankhani, H., and Tawil, A.-R. H., Hosseini, S. A. (2013). A Semantic Rule-based Approach for supporting Personalised Adaptive E-learning. *International Journal of Expert Systems with Applications. (In process).*

A.2 Academic Posters

Yarandi, M. Ontological approach to support a personalised adaptive e-learning system (onto-PADEL). *The winner of the Poster Competition at BCS e-learning SG's INSPIRE 2012 Conference, 23 Aug 2012, University of Tampere, Finland.*

Yarandi, M., Tawil, A. R., and Jahankhani, H. Ontologies for personalised adaptive learning. *Winner of the Perryman award for the best poster presentation in Advances in Computing & Technology Conference, 19 Jan 2012, London, UK.*

Appendix B

Academic Posters

Ontologies for Personalised Adaptive Learning

Maryam Yarandi
Abdel-Rahman H. Tawil
Hossein Jahankhani

School of Architecture, Computing and Engineering (ACE)

1. Introduction
Personalised services are nowadays an important research issue in the field of e-learning because no fixed learning paths are appropriate for all learners. Typically, traditional learning systems deliver the same content to all learners, irrespective of their learning styles, abilities, knowledge levels and preferences. This problem may be addressed by adapting the learning content toward the characteristics of particular learner. This paper proposes an innovative Ontological approach to support a Personalised Adaptive e-learning system (Onto-PAdel system) which assembles dynamically instructional objects to generate tailored learning objects for individual learners. The learning content associated with sequencing logic provides a clear separation between the domain and content models to increase the reusability and flexibility of the system. Additionally, learner's profiles are modelled to describe learner's characteristics. The proposed system monitor learner at different stages of learning process to ensure that specific targets have been made prior to the next level of learning.

2. Objectives
Adaptive systems provide personalised content in accordance with the knowledge and characteristics specific to individual learners. The aim of this work is to generate personalised learning content based on three axes of adaptivity:
• Pedagogical approaches: Using a combination of sequenced instructional objects such as conceptual definitions, examples, practices and explanations
• Learner's characteristics: using learner's characteristics such as prior knowledge, learning style and preferences
• Learner's progress: learners' progress at each step of the learning process is based on result of regular tests and learners' abilities calculated according to the Item Response Theory (IRT).

3. Onto-PAdel System
Ontology driven Personalised Adaptive e-learning (Onto-PAdel) system is an adaptive e-learning system which adapts learning content to learner's current level of knowledge, abilities, learning style, and other personal preferences. According to information in the learner's model the structure of the course is presented to the learner in the form of annotated course structure by using link annotations and link hiding of adaptive navigation support techniques.

4. Semantic Models For Onto-PAdel
Using ontologies in our proposed approach has the benefit of building reusable modular systems capable to reflect individual learner's needs. Other important benefit is the ability to automatically compose instructional objects into a new Lesson adapted with specific instructional design and needs for individual learners. The Onto-PAdel system is domain independent and it is a fully ontology based system. A segment of the domain ontology is presented in the following figure.

5. Content Model For Onto-PAdel
In order to facilitate direct access to course components a semantic representation of course structures is defined in our system. Consequently, the same Instructional object can be used in different ways to tailor learning content according to preferences, abilities, learning style, and other specific features of a learner that is relevant for the learning process. Therefore, we are able to create on the fly personalised learning contents out of those components. The content and user model ontologies used as part of our adaptive system is shown in the following figures.

6. Conclusion
This paper presents a personalised e-learning system which creates adaptive content for learner based on learner's ability, learning style, level of knowledge and preferences. In addition, Ontology-based approach is used to design domain, content and user model. The domain model consists of some classes and properties to define the topics of a domain and semantic relation between them. The content model describes the structure of courses and their components. The learner model describes learner's characteristics to deliver the tailor content. This modelling is used to annotate instructional objects and generate adaptive content based on individual learner's needs. The response of the learner to some regular test during the learning process is analysed by the item response theory and the ability of learner is evaluated. The progress from one stage of learning process to the next stage is determined based on the updated learner's profile. The prototype system is still being constructed.

Figure B.1: Academic poster by Maryam Yarandi; won the Perryman award for the best poster presentation at Advances in Computing & Technology Conference, 19 Jan 2012



School of Architecture, Computing and Engineering (ACE)

Ontological approach to support a Personalised Adaptive e-learning system (Onto-PAdel)

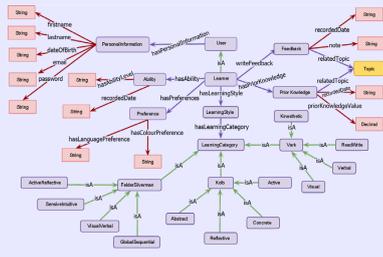
Maryam Yarandi*

* Thanks to Dr. Hossein Jahankhani and Dr. Abdel-Rahman H. Tawil

Personalised services are nowadays an important research issue in the field of e-learning as no fixed learning paths are appropriate for all learners. Typically, traditional learning systems deliver the same content to all learners, irrespective of their characteristics. This problem may be addressed by adapting the learning content toward the characteristics of particular learner. This study proposes an innovative **Ontological** approach to support a **Personalised Adaptive e-learning system (Onto-PAdel system)** which assembles dynamically instructional objects to generate tailored learning content for individual learners based on learner's characteristics and analysis of previous learning steps.



Using ontologies in our proposed approach has the benefit of building reusable modular systems capable of reflecting individual learner's needs. Other important benefit is the ability to automatically compose Instructional Objects into new lessons adapted with specific instructional design and needs for individual learners. The Onto-PAdel system is domain independent and it is a fully ontology based system.



Four ontological knowledge models namely domain, user, content and test are incorporated as part of the system to enable adaptation. Learners are engaged in learning topics, complete activities and take tests, while the system continuously updates learners' profiles and provide learning recommendation based on the analysis of learners' progress during the learning process.

Figure B.2: Academic poster by Maryam Yarandi; the winner of the poster competition at BCS e-Learning SGs INSPIRE 2012 Conference, 23 Aug 2012

Appendix C

SWRL rules

C.1 Annotated course structure

```
Topic(?t), Learner(?x), PriorKnowledge(?p), hasPriorKnowledge(?x, ?p),  
relatedTopic(?p, ?t), pkScore(?p, ?v), greaterThanOrEqualTo(?v, 50.0) → knows(?x,  
?t)
```

```
Topic(?t), Learner(?x), PriorKnowledge(?p), hasPriorKnowledge(?x, ?p),  
relatedTopic(?p, ?t), pkScore(?p, ?v), lessThan(?v, 50.0) → notKnow (?x, ?t)
```

```
Topic(?t), Learner(?x), knows(?x, ?t) → learned (?x, ?t)
```

```
Topic(?t), Topic(?t1), Learner(?x), knows(?x, ?t), isPrerequisiteFor(?t, ?t1) →  
readyToLearn(?x, ?t1)
```

```
Topic(?t), Topic(?t1), Learner(?x), notKnows(?x, ?t), isPrerequisiteFor(?t, ?t1)  
→ notReadyToLearn(?x, ?t1)
```

C.2 Ability Level

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,  
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqualTo(?s, -0.5),  
lessThan(?s, 0.5), subtract(?s, ?v1, ?v2) → hasAbilityToLearn(?x, ?y)
```

```
InstructionalObject(?y), InteractiveIO(?type), Learner(?x), Interaction(?p),  
hasDomainTopic(?y, ?d), hasIOType(?y, ?type), hasInteraction(?x, ?p),  
relatedTopic(?p, ?d), difficultyValue(?y, ?v1), activityValue(?p, ?v2),  
greaterThanOrEqualTo(?s, -0.5), lessThan(?s, 0.5), subtract(?s, ?v1, ?v2) →  
hasAbilityToDo (?x, ?y)
```

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,  
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqualTo(?s, 0.5),  
lessThan(?s, 1.5), subtract(?s, ?v1, ?v2) → hasOneLevelLowerAbility(?x, ?y)
```

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,  
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqualTo(?s, 1.5),  
lessThan(?s, 2.5), subtract(?s, ?v1, ?v2) → hasTwoLevelLowerAbility(?x, ?y)
```

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,  
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqualTo(?s, 2.5),  
lessThan(?s, 3.5), subtract(?s, ?v1, ?v2) → hasThreeLevelLowerAbility(?x, ?y)
```

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqual(?s, -1.5),
lessThan(?s, -0.5), subtract(?s, ?v1, ?v2) → hasOneLevelHigherAbility(?x, ?y)
```

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqual(?s, -2.5),
lessThan(?s, -1.5), subtract(?s, ?v1, ?v2) → hasTwoLevelHigherAbility(?x, ?y)
```

```
InstructionalObject(?y), Learner(?x), hasAbility(?x, ?a), difficultyValue(?y,
?v1), abilityValue(?a, ?v2), lastAbility(?a, true), greaterThanOrEqual(?s, -3.5),
lessThan(?s, -2.5), subtract(?s, ?v1, ?v2) → hasThreeLevelHigherAbility(?x, ?y)
```

C.3 Non-numeric terminology

```
Ability(?a), abilityValue(?a, ?v1), lessThan(?v1, -1.5) → AbilityLevel(?a,
VeryLow)
```

```
Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqual(?v1, -1.5), lessThan(?v1,
-0.5) → AbilityLevel(?a,Low)
```

```
Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqual(?v1, -0.5), lessThan(?v1,
0.5) → AbilityLevel(?a,Moderate)
```

```
Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqual(?v1, 0.5), lessThan(?v1,
1.5) → AbilityLevel(?a,High)
```

```
Ability(?a), abilityValue(?a, ?v1), greaterThanOrEqual(?v1, 1.5) →
AbilityLevel(?a, VeryHigh)
```

```
InstructionalObject(?y), difficultyValue(?y, ?v1), lessThan(?v1, -1.5) →
DifficultyLevel(?y,Basic)
```

```
InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqual(?v1, -1.5),
lessThan(?v1, -0.5) → DifficultyLevel(?y,Primary)
```

```
InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqual(?v1, -0.5),
lessThan(?v1, 0.5) → DifficultyLevel(?y,Intermediate)
```

```
InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqual(?v1, 0.5),
lessThan(?v1, 1.5) → DifficultyLevel(?y, UpperIntermediate)
```

```
InstructionalObject(?y), difficultyValue(?y, ?v1), greaterThanOrEqual(?v1, 1.5)
→ DifficultyLevel(?y,Advanced)
```

```
Interaction(?y), activityValue(?y, ?v1), lessThan(?v1, -1.5) → activityLevel(?y,
VeryEasy)
```

```
Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqual(?v1, -1.5),
lessThan(?v1, -0.5) → activityLevel(?y, Easy)
```

```
Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqual(?v1, -0.5),
lessThan(?v1, 0.5) → activityLevel(?y, Moderate)
```

```
Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqual(?v1, 0.5),
lessThan(?v1, 1.5) → activityLevel(?y,Difficult)
```

```
Interaction(?y), activityValue(?y, ?v1), greaterThanOrEqual(?v1, 1.5) →
activityLevel(?y, VeryDifficult)
```

C.4 Learning Style

```
Learner(?x), InstructionalObject(?y), LearningStyle(?s), Visual(?k), supports(?y,
Visual), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s),
learningCategoryRanking(?k, "1") → LSIsSupportedWith(?x, ?y)
```

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Audio(?k), supports(?y, Audio), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "1") → LS1IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), ReadWrite(?k), supports(?y, Read/Write), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "1") → LS1IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Kinesthetic(?k), supports(?y, Kinesthetic), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "1") → LS1IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Visual(?k), supports(?y, Visual), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "2") → LS2IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Visual(?k), supports(?y, Visual), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "3") → LS3IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Visual(?k), supports(?y, Visual), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "4") → LS4IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Audio(?k), supports(?y, Audio), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "2") → LS2IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Audio(?k), supports(?y, Audio), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "3") → LS3IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Audio(?k), supports(?y, Audio), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "4") → LS4IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), ReadWrite(?k), supports(?y, Read/Write), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "2") → LS2IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), ReadWrite(?k), supports(?y, Read/Write), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "3") → LS3IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), ReadWrite(?k), supports(?y, Read/Write), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "4") → LS4IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Kinesthetic(?k), supports(?y, Kinesthetic), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "2") → LS2IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Kinesthetic(?k), supports(?y, Kinesthetic), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "3") → LS3IsSupportedWith(?x, ?y)

Learner(?x), InstructionalObject(?y), LearningStyle(?s), Kinesthetic(?k), supports(?y, Kinesthetic), hasLearningCategory(?s, ?k), hasLearningStyle(?x, ?s), learningCategoryRanking(?k, "4") → LS4IsSupportedWith(?x, ?y)

C.5 Recommended IO

Learner(?x), InstructionalObject(?y), StaticIO(?t), Language(?g), hasAbilityToLearn(?x, ?y), LS1IsSupportedWith(?x, ?y), nextIOType(?x, ?t), selectedTopic(?x, ?d), hasDomainTopic(?y, ?d), hasIOType(?y, ?t), hasLanguagePreference(?x, ?g), isInLanguage(?y, ?g) → isRecommendedStaticIO(?x, ?y)

Learner(?x), InstructionalObject(?y), InteractiveIO(?t), Language(?g), hasAbilityToDo(?x, ?y), LS1IsSupportedWith(?x, ?y), nextIOType(?x, ?t), selectedTopic(?x, ?d), hasDomainTopic(?y, ?d), hasIOType(?y, ?t), hasLanguagePreference(?x, ?g), isInLanguage(?y, ?g) →

isRecommendedInteractiveIO(?x, ?y)

Learner(?x), InstructionalObject(?y), StaticIO(?t), Language(?g),
 hasAbilityToLearn(?x, ?y), LSIsSupportedWith(?x, ?y), selectedTopic(?x, ?d),
 hasDomainTopic(?y, ?d), hasIOType(?y, ?t), hasLanguagePreference(?x, ?g),
 isInLanguage(?y, ?g) → isRecommendedStaticIOs(?x, ?y)

Learner(?x), InstructionalObject(?y), InteractiveIO(?t), Language(?g),
 hasAbilityToDo(?x, ?y), LSIsSupportedWith(?x, ?y), selectedTopic(?x, ?d),
 hasDomainTopic(?y, ?d), hasIOType(?y, ?t), hasLanguagePreference(?x, ?g),
 isInLanguage(?y, ?g) → isRecommendedInteractiveIOs(?x, ?y)

C.6 Guidance

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityLevel(?p, Moderate) →
 isGuided(?x, moreExample)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityLevel(?p, Moderate),
 isGuided(?x, moreExample), IOLevel(?y, ?v1), lessThan(?v1, 0) → isGuided(?x,
 repeatPrerequisite)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityLevel(?p, Moderate),
 isGuided(?x, moreExample), IOLevel(?p, ?v1), greaterThanOrEqual(?v1, 0) →
 isGuided(?x, repeatWithLowerLevel)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v1), greaterThanOrEqual(?v1, 1) → isGuided(?x, moreActivityOrTest)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v1), greaterThanOrEqual(?v1, 1) isGuided(?x, moreActivity) → isGuided(?x,
 teachWithHigherLevel)

Learner(?x), responseToIO(?x, true), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v1), lessThanOrEqual(?v1, 0) → isGuided(?x, nextLevel)

Learner(?x), responseToIO(?x, true), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v1), greaterThanOrEqual(?v1, 1) → isGuided(?x, assessment)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v2), lessThanOrEqual(?v2, -1), IOLevel(?p, ?v1), lessThan(?v1, 0) → isGuided(?x,
 repeatPrerequisite)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v2), lessThanOrEqual(?v2, -1), IOLevel(?p, ?v1), greaterThanOrEqual(?v1, 0) →
 isGuided(?x, repeatWithLowerLevel)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v1), greaterThanOrEqual(?v1, 1) isGuided(?x, teachWithHigherLevel) → isGuided(?x,
 moreExample)

Learner(?x), responseToIO(?x, false), Interaction(?p), hasInteraction(?x,
 ?p), relatedTopic(?p, ?d), selectedTopic(?x, ?d), activityValue(?p,
 ?v1), greaterThanOrEqual(?v1, 1) isGuided(?x, moreExample) → isGuided(?x,
 assessment)

Appendix D

Screenshots of the Programme

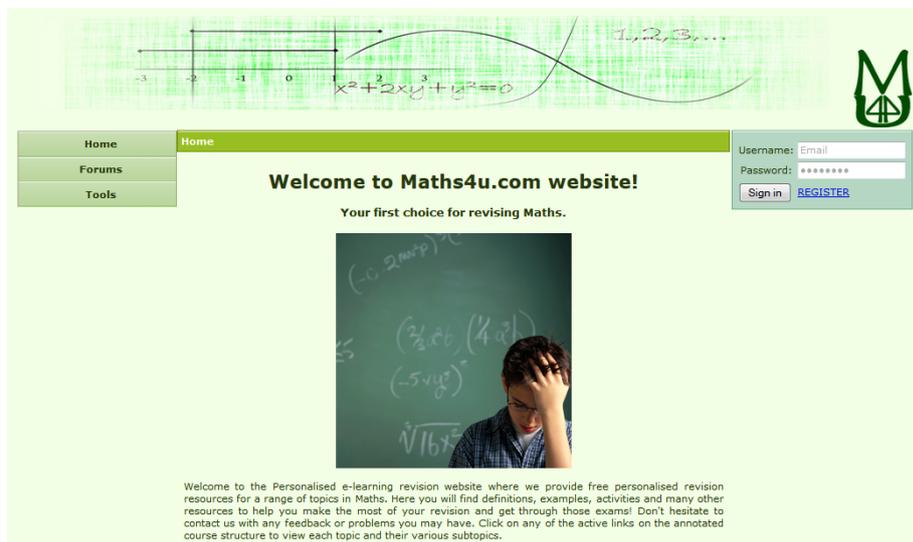


Figure D.1: Login Page

First name: *

Last name: *

Date of birth: *

Email address: *

Confirm email address: *

Color Preference: *

Language Preference: *

Post Code:

Telephone:

Mobile:

Password: *

Confirm password: *

I accept the [Terms and Condition](#) of the website.

*Please fill in all required fields (*) in this form.*

Figure D.2: Registration form

Home > Learning Style

Choose the answer which best explains your preference and check the box next to it. Please check more than one if a single answer does not match your perception. Leave blank any question that does not apply. you should answer min 12 question.

1. You are helping someone who wants to go to your airport, town centre or railway station. You would:

- go with her.
- tell her the directions.
- write down the directions.
- draw, or give her a map.

2. You are not sure whether a word should be spelled `dependent' or `dependant'. You would:

- see the words in your mind and choose by the way they look.
- think about how each word sounds and choose one.
- find it in a dictionary.
- write both words on paper and choose one.

3. You are planning a holiday for a group. You want some feedback from them about the plan. You would:

- describe some of the highlights.
- use a map or website to show them the places.
- give them a copy of the printed itinerary.
- phone, text or email them.

4. You are going to cook something as a special treat for your family. You would:

Figure D.3: Learning style Test

home

Choose the answer which best explains your knowledge level of the respected topics and select the appropriate option. If you do not make any selection the default value which is "Never Heard of the topic" will be used.

Arithmetic Operation	Know well
Percentage	Never heard of the topic
What is a fraction?	Have a basic idea
Ratio	Never heard of the topic
Approximation	Never heard of the topic
Decimal Number	Never heard of the topic
Whole Number	Have a basic idea
Mixed Number	Never heard of the topic

Next

Figure D.4: Prior knowledge form Test

Home > Topic

Structure of Content

- **Arithmetic Operation**
- Percentage
- **What is a fraction?**
 - **Concept of Fraction**
 - **Proper Fraction**
 - **Improper Fraction**
 - **Comparing Fraction**
 - **Comparing Two Fraction**
 - **With same denominator**
 - Different denominator
 - Comparing More Than Two Fraction
 - Ordering Fraction
 - Equivalent Fraction
 - Concept of Equivalent Fraction
 - Simplifying Fraction
 - **Adding Fraction**
 - **With same denominator**
 - With different denominator
 - **Subtracting Fraction**
 - **With same Denominator**
 - With different Denominator
 - Multiplying Fractions

Instruction

Now you can select the subject you want to study.

- Note the colour of the links:

Purple: You already know the subject. You can Select these subjects to study them again.

Blue: These are the subjects which are suitable for you to learn right now.

Gray: You are not ready yet to study these.

You have selected to Study:

Go Ahead



Figure D.5: Annotated course structure

The screenshot shows the 'ProperFraction' page. On the left, there is a navigation menu with 'Home', 'Forums', and 'Tools'. Below it, the user's name 'MARYAM' and family 'YARANDI' are listed, along with learning style percentages: Read/Write (56.25%), Visual (18.75%), Audio (12.5%), and Kinesthetic (12.5%). The main content area is titled 'ProperFraction' and has tabs for 'intro', 'definition', 'example', 'exercise', and 'assessment'. The 'definition' tab is active. The text asks 'What is a fraction?' and states 'A fraction is a part of a whole. There are two numbers to every fraction:'. Below this is a diagram showing the fraction $\frac{2}{7}$ with arrows pointing to the '2' labeled 'Numerator' and the '7' labeled 'Denominator'. To the right of the fraction is a bar model with 7 equal segments, the first two of which are shaded green. Below the diagram, it says 'The top number of the fraction is called the numerator. The bottom number is called the denominator.' At the bottom of the content area are 'Previous' and 'Next' buttons. On the right side, there is a login section with 'Username: Email' and 'Password: *****' fields, and 'Sign in' and 'REGISTER' buttons. Below that, a recommendation says 'Read More examples and do more exercise for better understanding the lesson.' and the difficulty level is 'Easy'.

Figure D.6: definition of proper fraction

The screenshot shows the 'ProperFraction' page with the 'example' tab active. The main content area displays a number line from 0 to 1, divided into 6 equal parts. The first 4 parts are shaded red. Below the number line, the fraction $\frac{4}{6}$ is shown, with '4' labeled 'Numerator' and '6' labeled 'Denominator'. Below the fraction, it says 'The denominator 6 shows that the distance from 0 to 1 is divided into 6 equal parts. The numerator 4 shows that 4 of the parts is selected. This fraction can also be written as four-sixth.' At the bottom of the content area are 'Previous' and 'Next' buttons. The right side of the page is identical to Figure D.6, but the difficulty level is 'Moderate'.

Figure D.7: Example of proper fraction

The screenshot shows the 'ProperFraction' page with the 'exercise' tab active. The main content area asks 'How much is shaded?' and shows a circle divided into 8 equal sectors, with 3 sectors shaded red. Below the circle, there are three radio button options: $\frac{2}{8}$, $\frac{2}{7}$, and $\frac{6}{8}$. Below the options is a 'check Answer' button. At the bottom of the content area are 'Previous' and 'Next' buttons. The right side of the page is identical to Figure D.6, but the difficulty level is 'Moderate'.

Figure D.8: Exercise about proper fraction

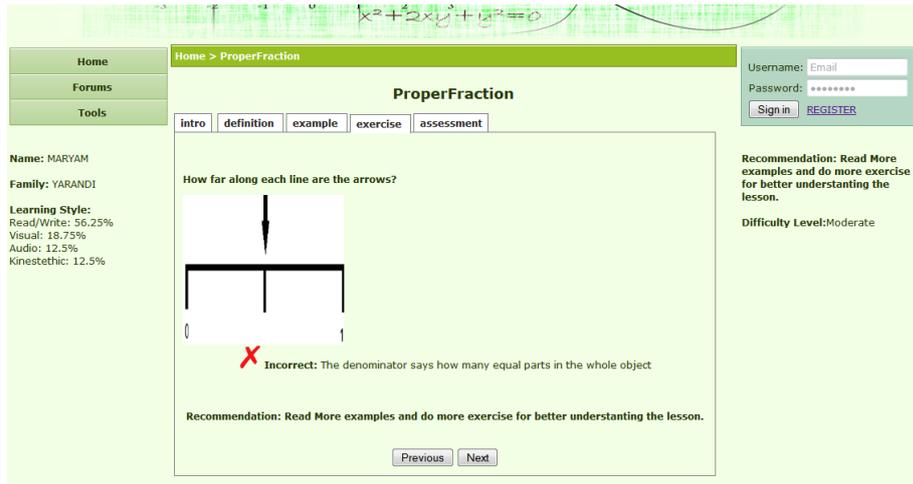


Figure D.9: Recommended adaptive guide

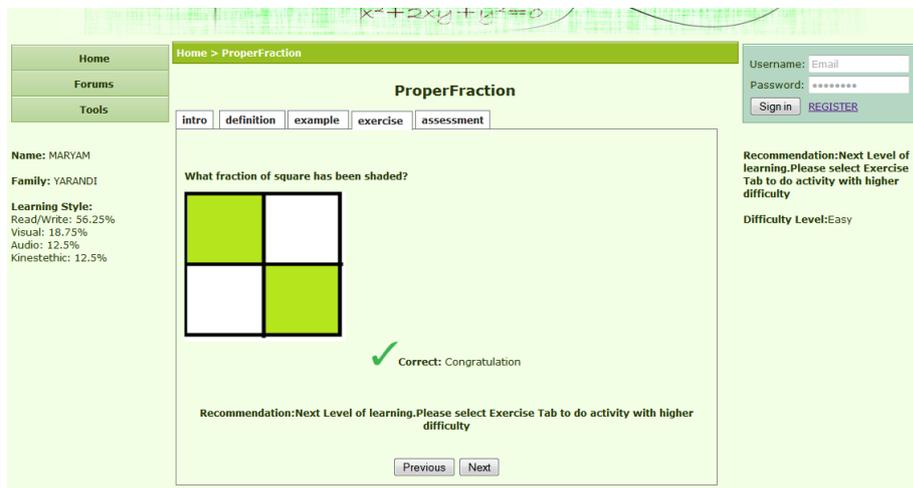


Figure D.10: Recommended adaptive guide

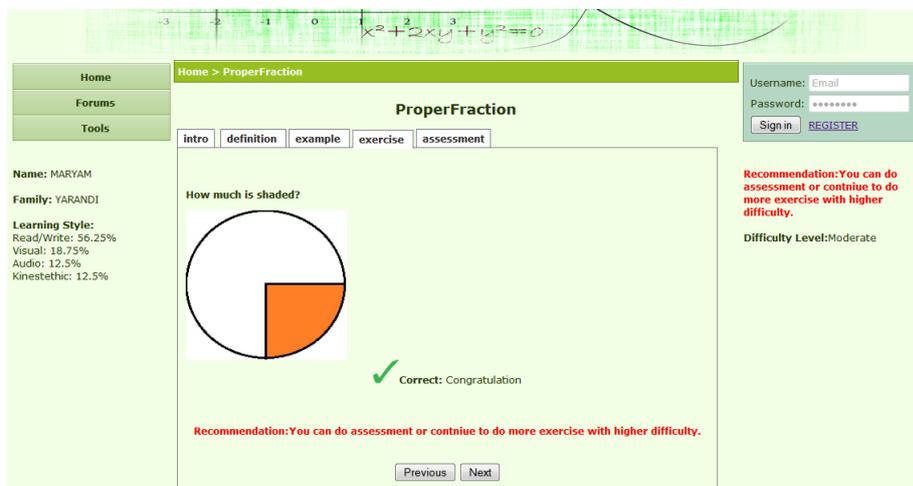


Figure D.11: Recommended adaptive guide

The screenshot shows a web-based educational interface. At the top, there is a header with a coordinate plane and the equation $x^2 + 2xy + y^2 = 0$. Below the header, the page is titled "ProperFraction". On the left side, there is a navigation menu with "Home", "Forums", and "Tools". Below the menu, user information is displayed: Name: MARYAM, Family: YARANDI, and Learning Style: Read/Write: 56.25%, Visual: 18.75%, Audio: 12.5%, Kinesthetic: 12.5%. On the right side, there is a login section with fields for "Username: Email" and "Password: *****", and buttons for "Sign in" and "REGISTER". Below the login section, a red recommendation message reads: "Recommendation: You finish this topic. Please select Assessment Tab to do related Assessment." and "Difficulty Level: Difficult". The main content area has tabs for "intro", "definition", "example", "exercise", and "assessment". The "assessment" tab is selected, showing a question: "How far along each line are the arrows?". Below the question is a diagram with a horizontal line from 0 to 1 and five vertical lines extending downwards. A green checkmark and the text "Correct: Congratulation" are displayed below the diagram. At the bottom of the assessment area, there are "Previous" and "Next" buttons. A red recommendation message is also present at the bottom of the main content area: "Recommendation: You finish this topic. Please select Assessment Tab to do related Assessment."

Figure D.12: Recommended adaptive guide

Appendix E

Domain Mathematics Ontology

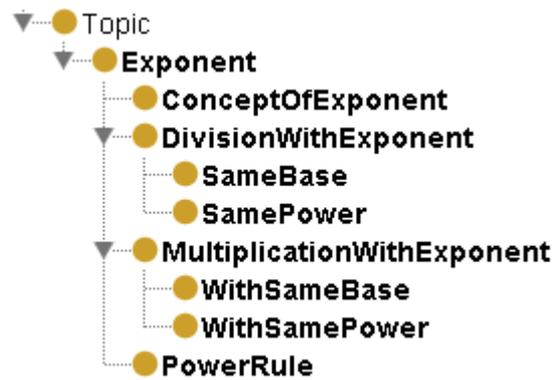


Figure E.1: Domain ontology topic Fraction

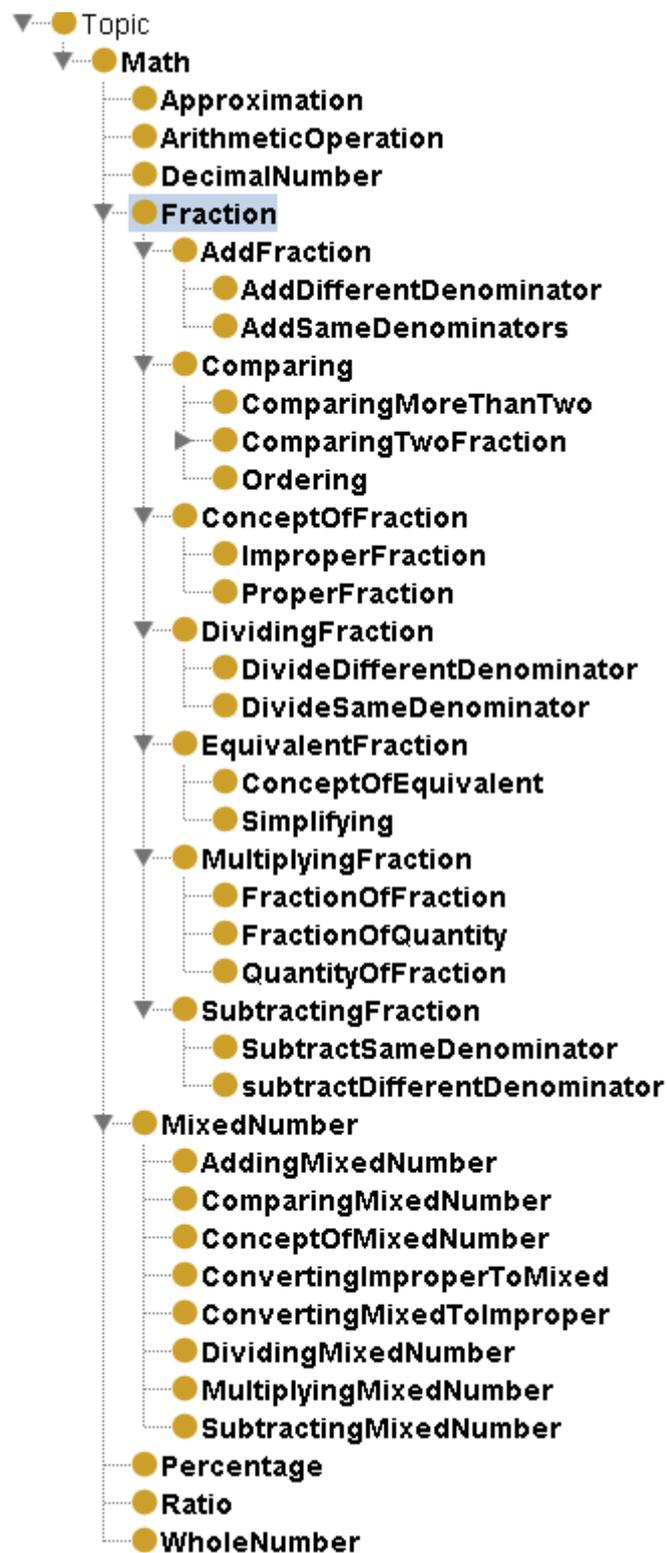


Figure E.2: Domain ontology topic Exponent

Appendix F

Student Evaluation Questionnaire

Feedback Form

Age range: 16 or below 17 or above

Please rate how strongly you agree or disagree with each of the following statements by circling the appropriate number. (Scale is from 1 being strongly agree to 5 being strongly disagree):

	Strongly Agree	Agree	Neither Agree Nor Disagree	Disagree	Strongly Disagree
Comparing the Adaptive and the Non-adaptive System					
1- The adaptive system was more helpful for learning a new topic than the non-adaptive system.	1	2	3	4	5
2- It was easier to solve my difficulties with the help of the recommendations of the adaptive system than in the non-adaptive system.	1	2	3	4	5
3- I prefer the adaptive system because it provides interactive features for learners.	1	2	3	4	5
4- I prefer adaptive recommendations because it accurately brings helpful supplementary contents tailored to my needs and I did not have to look for them in the entire course.	1	2	3	4	5
Satisfaction with Adaptive Navigation					
5- The colours used in the course structure make the items more clear.	1	2	3	4	5
6- The generated annotated course structure is easy to navigate.	1	2	3	4	5
7- The adaptive annotations of the annotated course structure helped me to choose the most appropriate next topic to learn.	1	2	3	4	5
Appropriateness of the Personalised Content					
8- The personalised services provided by the adaptive system satisfied my preferences.	1	2	3	4	5
9- I am satisfied with the difficulty level of materials recommended by the adaptive system.	1	2	3	4	5
10- I am satisfied with the difficulty level of the exercises and assessments recommended by the adaptive system.	1	2	3	4	5
11- I am satisfied with the quality of the personalisation.	1	2	3	4	5
12- I benefited from the materials recommended by the adaptive system.	1	2	3	4	5
Usage of the Adaptive Guidance					
13- The adaptive guidance helps me to solve my difficulties.	1	2	3	4	5
14- The adaptive guidance helps me to plan the next step of my learning.	1	2	3	4	5
15- The adaptive guidance helps me to access sufficient material, examples and exercises to solve my difficulties.	1	2	3	4	5
16- I tried to use the guidance offered in the adaptive system.	1	2	3	4	5
Learner's Interest					
17- I believe that the system is user-friendly.	1	2	3	4	5
18- The adaptive system can promote my learning interests.	1	2	3	4	5
19- I feel that the time passes very quickly when I use the adaptive system to learn mathematics.	1	2	3	4	5
20- I would recommend this system to my classmates.	1	2	3	4	5

What did you least like about the adaptive system?

What did you most like about the adaptive system?

Please use this area to write down any other comments you may have: