A Framework for Adaptive e-Learning

by

Keith Maycock, BSc.

NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

Dissertation submitted in partial fulfillment of the requirements for candidate for the degree of Doctor of Philosophy

Department of Computer Science,
National University of Ireland, Maynooth, Co. Kildare, Ireland.

Supervisor: Dr. John G. Keating

October 2010

Abstract

Adaptive learning systems attempt to adapt learning content to suit the needs of the learners using the system. Most adaptive techniques, however, are constrained by the pedagogical preference of the author of the system and are always constrained to the system they were developed for and the domain content. This thesis presents a novel method for content adaptation. A personal profile is described that can be used to automatically generate instructional content to suit the pedagogical preference and cognitive ability of a learner in real time. This thesis discusses the manifestation of measurable cognitive traits in an online learning environment and identifies cognitive resources, within instructional content, that can be used to stimulate these manifestations.

There exists two main components for the learning component: Content Analyser and a Selection Model. The Content Analyser is used to automatically generate metadata to encapsulate cognitive resources within instructional content. The analyser is designed to bridge the perceived gap found within instructional repositories between inconsistent metadata created for instructional content and multiple metadata standards being used. All instructional content that is analysed is repackaged as Sharable Content Object Reference Model (SCORM) conforming content. The Selection Model uses an evolutionary algorithm to evolve instructional content to a Minimum Expected Learning Experience (MELE) to suit the cognitive ability and pedagogical preference of a learner. The MELE is an approximation to the expected exam result of a learner after a learning experience has taken place. Additionally the thesis investigates the correlation between the cog-

nitive ability and pedagogic preference of an author of instructional content and the cognitive resources used to generate instructional content. Furthermore the effectiveness of the learning component is investigated by analysing the learners increase in performance using the learning component against a typical classroom environment.

Acknowledgments

I am very grateful to my wife Bernie who has stuck beside me throughout a number of years, that has seen us live through all kinds of experiences. In particular, when Conor was born and Bernie did all the night feeds to ensure that I finish the PhD. Bernies support throughout the years focused me and helped to finish the PhD after starting a new job, without Bernies support and help I would not have finished in 2010. However with the thanks there is also disappointment, as after successfully defending the thesis I arrived home to be introduced to Dr Manny (our dog!), Dr Mammy, Dr Paul and Dr Conor, by Paul who was nearly three. I need to work at becoming a Professor now.

In addition to the support of my wife and children I have had great support from my family who have always supported my crazy ideas and instilled in me from an early age that anything was achievable, thanks Mam and Dad. I would like to thank my older broth Jonathan who sat behind me in the research lab at NUI, Maynooth. Even though we researched in completely different fields the respect for each other grew over the years at Maynooth and our countless discussions helped me to take a more pragmatic view towards my research. Education is a fantastic element of life and I am delighted that my mother and sister are currently pursuing third level qualifications. I am so proud that education is held in such high regard with my family and one day we can all be Doctors. I am very grateful to Carl my younger brother, who is one of my best friends for his support over the years. I believe that once Carl finds himself he will excel in whatever field he applies himself to.

John Keating has been an inspiration and friend throughout the PhD process. His guidance has helped me cope with the stresses of life as well as focusing on constructing a solution for the PhD. Thank you John.

To the guys and girls of the research labs in Maynooth I am eternally grateful for all our conversation explaining the latest trends in technology. In particular, great thanks to Jonathan Lambert who has been a friend and teacher, since first meeting him in the library at NUIM. Jonathan has been extremely supportive throughout the years and his expert understanding of programming helped me to complete my final year undergraduate project. Jonathan and I have had, and will continue to have countless conversations investigating pedagogic strategies and technologies and working together now I am sure he will achieve whatever he wants. I am very happy that we have been lucky enough to find work together.

Contents

List of Figures

List of Tables

Abbreviations used in thesis

ACO	Ant Colony Optimisation
ADL	Advanced Distributive Learning
AHS	Adaptive Hypermedia Systems
AICC	Aviation Industry CBT (Computer-Based Training) Committee
ALE	Adaptive Learning Environment
CA	Content Analyser
CABLE	Cognitive Apprenticeship Based Learning
CMS	Content Management System
CTM	Cognitive Trait Model
EO	Extremal optimisation
ESO	Exploration Space Control
GA	Genetic Algorithm
IEEE LTSC	IEEE Learning Technology Standards Committee
IMS	Innovation Adoption Learning
IPS	Information Processing Speed
ITS	Intelligent Tutoring Systems
FF	Fitness Function
LCS	Largest Common Subgraph
LMS	Learning Management System
LOM	Learning Object Metadata
LTWM	Long-Term Working Memory
MELE	Minimum Expected Learning Experience
MOODLE	Modular Object Oriented Development Learning Environment

MERLOT	Multimedia Educational Resource for Learning an Online Teaching
MRA	Multiple Representation Approach
NDLR	National Digital Learning Repository
PBL	Problem Based Learning
SCORM	Sharable Content Object Reference Model
SCORM CAM	SCORM Content Aggregation Model
SCORM RTE	SCORM Run Time Environment
SCORM SN	SCORM Sequencing and Navigation
SCO	Sharable Content Object
SOC	Self-organised criticality
VLE	Virtual Learning Environment

Definition of Terms used in thesis

The usage of the following terms differs in the general use, in computer science research, and in educational research. The following definitions describe the use of these terms throughout this thesis.

Adaption	Adaption refers to the generation of instructional content based on identified cognitive metrics within metadata describing instructional content associated with the cognitive ability of a learner.
Adaptive strategy	Adaptive strategy refers too the strategy that is used in the construction of new course material for an individual learner.
Blended learning	The use of both traditional class room teaching and on-line learning throughout a course.
Cognitive ability	The cognitive ability of a learner is defined as the metrics associated with how a learner consumes information.
Cognitive overload	Cognitive overload occurs when a learner is presented with too much information that causes interference to a learning experience.
Cognitive traits	The cognitive traits of a learner are individual traits that combine to form the cognitive ability of a learner.
Instructional object	Instructional object refers to a single unit of learning material. A SCORM compliant instructional object is described as a unit of learning material that must contain functionality to be tracked by a SCORM run time environment.

Learning experience	A learning experience is the output of a learner interacting with a learning environment.
Mathemagenic	Mathemagenic content is defined as content that is generated for a learner that will enable the learner to consume the information optimally.
Pedagogical preferences	The pedagogical preference of a learner refers to the learning style that a leaner uses to enhance their learning experience. In particular the pedagogical preference of learner in this thesis is involved with the classification of a learners VARK preference associated with an online learning environment.
Protocol	Refers to the steps involved with a learner interacting with the learning component throughout a learning experience.
Specification	A Specification is a description of an instructional course. The Specification contains characteristics that control the adaptive strategy when generating content for a learner. The Specification does not contain any instructional material.

Chapter 1

Introduction and Research Question

Adaptive learning systems have been in development since the early 1990s but have seen rapid development in more recent years. Coupled with an extensive increase of people entering into higher education, adaptive online learning systems may offer a potential avenue for higher eduction, either as a pure online strategy or a blended course (online course that typically has some traditional components). With the introduction of specifications like the Sharable Content Object Reference Model (SCORM), the Advanced Distributed Learning initiative (ADL) has attempted to standardize metadata specifications for learning content. However, it was found by Norm Friesen [2] that only fifty nine percent of people complete keywords in SCORM compliant learning objects, thus creating an impossible task for automated adaptive learning systems to use the metadata associated with the learning objects as an indication of the content. Furthermore the ADL team built SCORM (like all metadata standards) as a black-box specification where no process investigates the validity of the instructional content being referenced by the SCORM metadata, or the type of content contained within the Sharable Content

Object (SCO).

This research project is involved with the following research question:

Is it possible to construct an automated learning component that generates instructional content suited to the cognitive ability and pedagogical preference of a learner?

The goal of this thesis is to describe a suitable personal profile, consisting of the cognitive ability and pedagogical preference of a learner that has associated cognitive metrics found within instructional content. This thesis discusses the design, construction and evaluation of an automated learning component that is built to automatically generate instructional content using an evolutionary strategy, suited to the defined optimal personal profile.

1.1 Literature Review

In this section, the key literature relevant to this thesis is reviewed, in chronological order. The research completed within this research project has two distinct paths: identifying suitable cognitive traits and the investigation of suitable adaptation systems. In particular, this thesis is involved with the identification of suitable cognitive traits and pedagogical preferences of an individual that has an associative cognitive metric that can be automatically identified within instructional material. The literature review starts with early attempts of classifying the structure and capacity of Working Memory with Millers work [3] from 1956 and follows this research to the current thinking of how information is processed; for example, an investigation on the research conducted by Baddely[4], Cowan[5] and Ericsson and Kintsch[6] is carried out on the workings of memory capacity and storage limitations. Adaptive content strategies and systems are investigated in some detail, from the early work of Peter Brusilovsky [7] to Patels work [8] on improving cognitive traits by summative and formative assessment.

1.1.1 Cognitive traits and eduction philosophy

Miller [3] describes one bit of information as the amount of information that is needed to make a decision between two equally likely different alternatives. It is further suggested that N bits of information is required to decide between 2^N alternatives. Miller gives an account of a number of experiments determining the absolute judgment of unidimensional stimuli in contrast to the results found in determining the absolute judgment of multidimensional stimuli. Miller found that the span of absolute judgment and the span of immediate memory impose severe limitations on the amount of information that humans are able to perceive, process, and remember. If the stimuli are organized into several dimensions and successively into chunks of learning objects, the span of absolute judgment and the span of immediate memory are increased significantly.

Baddely et∼al [4] introduced the multicomponent model of working memory. This model is composed of two slave systems and a central executive system to control the flow of communication between the slave systems and to coordinate cognitive processes when more than one task must be completed at one time. The slave systems consist of a phonological loop and a visuo-spatial sketch pad. The phonological loop stores phonological information and prevents the decay of such information by constantly refreshing the information. The visuo-spatial sketch pad stores visual and spatial information and is used for the construction and manipulation of visual images. The sketch pad can be broken down further into two subsystems: a visual subsystem, responsible for shape, colour and texture and a spatial subsystem dealing with location.

This research thesis is involved with learning within an online learning environment. Considering working memory as a possible trait and the limited capacity associated with that trait as described above [3], the constructs of learning must also be considered. Dijk and Kinstch [9] discuss the tasks that must occur for basic text comprehension. The tasks that they identified are: perceptual features, linguistic features, propositional structure, macrostructure, situation model, control structure, goals, lexical knowledge, frames, general knowledge and episodic memory for prior text. Consequently, each of these tasks would impede on the general idea of working memory containing a limited bound.

Cowan [10] investigates the conceptions of memory storage, selective attention and their constraints within human information processing system. In particular the intersection of memory and attention was discussed, thus moving away from a simple static model for working memory capacity.

Ericsson and Kintsch [6] believe that there exists two structures within memory: working memory and long-term memory. However, they argue that there must exist some retrieval structures to allow for the expansion of working memory during certain conditions. They classify this expansion as having the ability to utilise long-term working memory (LTWM). For example, text comprehension requires all the following to take place: perceptual features, linguistic features, propositional structure, macrostructure, situation model, control structure, goals, lexical knowledge, frames, general knowledge and episodic memory for prior text [9]. Each of these components by themselves would exceed the capacity of short-term working memory, but are clearly needed in text understanding. This supports the concept of an additional storage device that can be used within certain circumstances. Ericsson and Kintsch [6] proposed that in certain situations of expertise that individuals can overcome the limitations of working memory and utilise the storage capacity of long-term memory. This storage facility can then be accessed through cues from the immediate memory. These proposals were supported by various experiments with text comprehension, mental calculations, and chess.

Kintsch et~al [11] discuss extensions to earlier research relating to the complex task of text comprehension [9]. Kintsch considers that every reader is able to form an episodic text structure during text comprehension, if the text is well written and if the content is familiar. Forming an episodic text structure allows the use of long-term working memory thus explaining how a complex process like text comprehension can be performed on a daily basis. Additionally, forming the episodic structure reduces the concept of a granular chunk as previously defined by Miller [3].

Baddeley furthered his model in 2000 [12] by introducing an episodic buffer as an additional component. This buffer is a temporal storage of phonological, visual, spatial and semantic information. The buffer is comprised of a limited capacity system that provides temporary storage of a multimodal code, that is the binding of information from the initial subsystems and long-term memory. The key characteristics of the new model focuses attention on the process of integration of information rather than viewing the sub-systems in isolation. This new model emphasis the importance of creating the link between the long-term memory and the sub-systems.

Laurillard [13] discusses the most common pedagogic strategy used in higher education, Knowledge acquisition. Knowledge acquisition, through lectures and reading, is referred to as a rhetorical activity seeking to persuade learners of an alternative way of looking at a world they already know through experience. Laurillard argues that this way of learning presupposes that a learner must be able to interpret a complex discourse of words, symbols, and diagrams in the required manner if the learner is to comprehend the correct meaning of the educational content. A number of studies carried out on the learners interpretation between structure and meaning have identified two contrasting approaches to studying a text: one known as a *holistic* approach where the learner views the educational content as a whole thereby preserving the structure of the content but learners may have difficulties with cognitive overload. The other known as an *atomistic* approach where the learner breaks the content up into granular pieces of information, hence distorting the structure of the content and losing the meaning. Laurillard investigates the potential of higher education and the problems associated with this protocol of imparting knowledge.

Cowan [5] regards working memory as part of long-term memory and not another component. Representations in working memory are a subset of the long-term memory. Working memory consists of two distinct levels. The first level consists of long-term memory representations that are activated. There is no limit to activation of representations in long-term memory. The second level is described as the focus of attention. The focus is regarded as capacity limited and can hold up to four of the activated representations at any given instant. This view of working memory is thus centered on the concept of monitoring the focus of attention and reducing the possibility of interference with the focus of attention throughout a learning experience.

Ericsson and Kintsch [6], as discussed earlier proposed that in situations of expertise, individuals can overcome working memory limitations. Guida et∼al [14] using the theory of text comprehension have proposed the *personalisation method* as a way to operationalise the LTWM. The *personalisation method* was tested with two groups in text comprehension. The personalised group recalled more objects and showed no sensitivity to interference (delay) and memory load than the non-personalised group.

Owen et∼al [15] showed that using an N-Back algorithm method for testing working memory capacity stimulates the same regions of the human brain when compared with the more established working memory tests, by performing a meta-analysis of normative functional neuroimaging studies.

1.1.1.1 Conclusion

This section of the thesis investigated the research associated with suitable cognitive traits and pedagogical strategies associated with online learning. In particular, the section was investigating the evolution of working memory throughout the last fifty years in order to establish the underlying principles of the operations of working memory and the associated capacity. Chapter three investigates these strategies for working memory to construct a suitable personal profile that contains cognitive traits, with associated cognitive metrics. The following section investigates the research associated with adaptive content strategies.

1.1.2 Content Adaptation using technology

Alty [16] stresses the importance of a user centered approach to multimedia interface design.The importance of various perspectives on multimedia interfaces is discussed and Alty posits that a multimedia interface should be viewed as a multisensory, multichannel, multitasking and multiuser approach to systems design and the emphasis should be on what such an approach offers to the user rather than what it technically comprises of. Additionally, the role of media within a learning environment is discussed. The role of media is seen to be complementary to education and cognitive development, however this importance is discussed in the recognition of the type of media to be used in the particular instance of instruction to properly convey the idea or concepts being put forward and also through the power of media combination.

Peter Brusilovsky et∼al [17] describes an approach for developing an adaptive electronic textbook and presents their implementation of *Interbook*. The authors identified the main problem associated with web-based courseware being that the content is typically developed to suit the typical pedagogy in most universities. The authors distinguish three different levels or steps of increasing complexity when developing their adaptive courseware. The design framework of *Interbook* is based on the architecture of ELM-ART and is fully discussed in the paper. The *Interbook* adaptive courseware approach was implemented and evaluated in several systems. It was found that the *adaptive guidance* provides significant assistance for novices, while *adaptive navigation* support provides significant assistance for the more experienced users.

The authors describe the environmental contexts of a learning environment [18]. These contexts include: the nature of the subject discipline and the level of its learning, the role of the human teacher and the suitability of an Intelligent Tutoring System (ITS) for the construction of a particular type of knowledge. It is suggested that in any joint cognitive learning space points of divergence are likely to arise due to the different teaching styles of educational designers and implementers thus inferring that different teachers could constrain the learning process in different ways, including defining the appropriate grain size for the particular individual being taught to maintain a cognitive load balance. The authors believe that the problem of designing learning resources should be addressed in the context of the nature of subject discipline instead of the overall educational theory.

The Adaptive Hypermedia Architecture (AHA) [19], as discussed by De Bra and Calvi consists of a user model defined by the learners knowledge about domain relevant concepts. This user model is created by the learner reading some content and then taking short quizzes. Every page that is displayed to the leaner contains two pieces of information: firstly, what user model elements must exist to allow a link to that page, and secondly, what the desired outcome would be after completing the page. This task of creating a one-size-fits all approach to learning based on experience gained through learning achieved after completing a learning unit is very inefficient and complex. The success of the AHA system is dependent on the ability of an author of instructional material to categorise and identify suitable passages through an instructional space.

Peter Brusilovsky and John Anderson [20] present an electronic ACT-R bookshelf, a system which supports learning ACT-R, a well-known theory in the field of cognitive psychology over the web. This paper uncovers concept-based knowledge representation behind adaptive electronic textbooks on the bookshelf, describes the main functionality of the system, provides some evaluation data, and speculates about possible extensions of bookshelf systems.

Ashok PATEL et∼al [21] discuss the potential, and pitfalls of various forms of assessment in a Cognitive Apprenticeship Based Learning Environment (CABLE).

The authors suggest the use of formative assessment to bridge the gap between the ‘Gulf of Execution’ and the ‘Gulf of Evaluation’ with respect to the learning of a new concept. A correlation is drawn between the outcome of traditional pedagogical strategies and summative assessment. It is suggested that badly designed summative systems can encourage wide spread adoption of shallow learning. Due to the rapid growth of technology the authors express the need for a ‘just-in-time’ ethos to learning. It is also suggested that a stronger emphasis is placed on formative assessment rather than on summative assessment.

Peter Brusilovsky [22] describes a concept-based course maintenance system that was developed for Carnegie Technology Education. The system is used to check the consistency and quality of a course through its life cycle. The problem that is being addressed in this paper is that all the tools available for content development are typically oriented implicitly for single author development. The author also discusses the potential advantages and pitfalls when indexing educational content with respect to some examples of some real world tools.

Ashok Patel et∼al [8] discuss a possible categorization of learning resources to match the different phases of skill acquisition. This paper also discusses an implementation of a cognitive apprenticeship-based learning environment by the Byzantium project and an independent feedback on its use in the real world. The tests that were carried out were in the numeric domain. The authors give a comparison between the learning ethos of a typical student in a classroom environment against the learning ethos of a life-long-learner. In constructing a sound pedagogical framework for their project they adapt Kurt VanLehn framework [23] for reviewing cognitive skill acquisition. The framework is broken up into three different phases: early, intermediate and late phase. Throughout the early phase the emphasis is on the learners becoming familiar with the domain concepts. During the intermediate phase the learner turns their attention to solving problems integrated with formative assessment. In the last phase the learner improves with practice and the assessment changes from formative to summative. In an independent study carried out at the University of Glasgow 71% of the students showed a preference for the Byzantium project.

Kinshuk et∼al [24] describe the Multiple Representation Approach (MRA) for presenting multimedia technology within intelligent educational systems. A strategy for implementing MRA on systems using the Cognitive Apprenticeship (CA) framework for task oriented disciplines where the main focus is on cognitive skill acquisition is discussed. The authors give an account of the CA framework and list examples of multimedia objects suitable for different tasks under the CA framework. It is discussed how MRA can be utilized to enhance a learning experience for learners with different domain competence levels, with respect to multimedia object selection and navigational object selection. The authors discuss general guidelines and recommendations on combining multiple multimedia objects to enhance the learning experience. The application of the approach in the design of the InterSim system is also described.

Ashok Patel et∼al [25] discuss the key aspects of Collin, Brown and Newmans Cognitive Apprenticeship Model and Pasks Conversation Theory [26] with respect to their implementation of an intelligent learning system. The paper focuses on the cognitive skills acquired through interactive learning and suggests that the different phases of skill acquisition are due to semantically semi-synchronous conversations. It is suggested that if a course is delivered by fine grained modules, no complex inferencing regarding the learners knowledge is required as a simple yes or no answer is adequate to apply adaptation strategies to the learners needs. The system that was developed recognizes all valid paths to a solution, thus supporting learners with different learning styles and adhering to the Pasks Conversation Theory. The advantages the World Wide Web offers in terms of Conversation Theory are also discussed.

Ashok Patel et∼al [25] discuss the key aspects of Collin, Brown and Newmans Cognitive Apprenticeship Model [27] and Pasks Conversation Theory [26] with respect to their implementation of an intelligent learning system. The paper focuses on the cognitive skills acquired through interactive learning and suggests that the different phases of skill acquisition are due to semantically semi-synchronous conversations. It is suggested that if a course is delivered by fine grained modules, no complex inferencing regarding the learners knowledge is required as a simple yes or no answer is adequate to apply adaptation strategies to the learners needs. The system that was developed recognizes all valid paths to a solution, thus supporting learners with different learning styles and adhering to the Pasks Conversation Theory. The advantages the World Wide Web offers in terms of Conversation Theory are also discussed.

He, S. et∼al [28] discuss the limitations of PBL learning environments. The authors address the problem of the learners becoming overwhelmed by the granularity of the problem and losing focus on the overall learning task by introducing adaptive technology into the PBL learning environments. A prototypical system was built based on the original architecture of the web-based intelligent educational systems incorporating a problem-based learning module. The system successfully introduced the student adaptivity into the PBL environment.

Peter Brusilovsky [7] provides a clear view on the process of Adaptive Hypermedia System (AHS) starting from the early design stage. The author illustrates the possible advantages AHS have over traditional Hypermedia Systems. The basic architecture of an AHS is composed of: a student model, knowledge space and hyperspace. The different implementations of the various components are contrasted to suit the needs of different systems. The author also reviews a number of modern AHS that are orientated to educational practitioners.

Lin et∼al [29] introduce the Cognitive Trait Model (CTM) that supplements performance-based student models by allowing relevant information, such as cognitive metrics about a particular student, to be transported to different domains. To illustrate the procedure of the trait analyser, a definition of working memory capacity is discussed. The effect the characteristics of working memory has on the learning process is also discussed. A number of manifestations of working memory capacity are identified from a broad range of researchers.

Kinshuk et∼al [30] discuss the process of modeling Inductive Reasoning Ability in a Virtual Learning Environment. The characteristics of Inductive Reasoning Ability are studied in relation to domain knowledge, generalization, working memory capacity, analogy, and hypothesis generation. The importance of supporting Inductive Reasoning is discussed with reference to a number of researches that address this problem. A limited list of manifestations of Inductive Reasoning Ability is produced. The list is prohibited despite the vast amount of research carried out on Inductive Reasoning Ability by the diverse viewpoints of inductive reasoning as well as the requirement of translatability of each manifestation into machine observable patterns.

Hong and Kinshuk [31] presents a mechanism to identify a learners learning style using the Felder Silver learning style theory. The learning style theory categorizes an individuals preferred learning into five dimensions: sensing / intuitive, visual / verbal, inductive / deductive, active / reflective and sequential / global. Due to pedagogical reasons the inductive / deductive dimension has been deleted. The developed system provides a questionnaire to enable learners to identify their learning style based on this theory. There are three possible degrees available for the four dimensions: mild, moderate and strong. The system assigns a default preference for mild and treats moderate and strong preferences as the same. This significantly reduces the combination of learning styles available.

Lin et∼al [32] categorizes adaptive techniques into two categories: adaptive navigation and content presentation. This paper investigates how and when the adaptive techniques can be used to support a learners working memory capacity. Additionally, an overview of popular techniques employed in modern adaptive learning systems is provided and the possible future trend of adaptive techniques is discussed.

Gabriela and Kenji [33] propose that there exists two type of adaptation in web-based tutorials: static and dynamic. They propose static adaptations to personalisation factors such as: learning styles, intelligence types, knowledge background, special interests, learning goals and beliefs. The authors propose, using the VARK inventory learning styles [1], to design the presentation of knowledge.

Owen Conlan et∼al [34] discussed a method used for the personalisation of news feeds using traditional adaptive hypermedia strategies and building semantic links between available news items. They investigated the advantages of using a strict ontology, where semantic matching is very high, against a loosely defined ontology. Their results showed that there did not exist a significant difference between the two cases, especially not to warrant the use of a strict ontology which is significantly more process intensive.

1.1.2.1 Conslusion

This section investigated adaptive learning systems and strategies. There exists a trend across most of the reviewed adaptive systems focusing on creating a number of threads through a learning space and then mapping these threads to a given student. This process of an author of educational instruction being in control of identifying and mapping instructional paths through a learning space is not very efficient as the cognitive ability and pedagogical preference of that author will influence the process. Chapters two and three investigate these adaptive strategies in more detail in order to design a suitable protocol for adapting content to suit the cognitive ability and pedagogical preference of a learner within an online learning environment.

1.1.3 Summary

This section summarised some of the important research papers, both in the area of adaptive hypermedia systems and techniques and within the classification of working memory capacity. These two research areas are critical to the foundation of the research carried out throughout this thesis. The following section details the contribution made by this research project in the area of adaptive learning technologies.

1.2 Thesis Contributions

The contributions of this thesis are split into two distinctive sections: A) Investigating a number of cognitive traits and environmental contexts of a learning environment that can be adapted to in an online learning environment, and B) the design and implementation of an evolutionary algorithm to automatically generate instructional content suited to the cognitive ability and pedagogical preference of a learner. Section A, is comprised of Chapters one through four, is concerned with the classification of suitable cognitive traits and pedagogical strategies independent of domain that can be mapped to cognitive metrics that encapsulate instructional content.. Section A also describes the design and implementation of a Content Analyser (CA) that automatically generates metadata. The CA takes as input an archive package and decouples the package to produce multiple SCOs. Additionally, for each SCO produced a metadata file is generated detailing information relating to the cognitive metrics found within the instructional content.

In the second of these sections, Section B, the design and implementation of an evolutionary algorithm to automatically generate instructional content suited to the cognitive traits and pedagogic preference of a learner to a minimum expected learning experience is discussed. Section B is comprised of Chapters five through seven. An investigation is carried out on the GA to determine the suitability of the algorithm as the solution space (no. of possible suitable learning objects) increases. Additionally, an analysis is performed firstly on the correlation between the personal profile (cognitive ability and pedagogic preference) of an instructional author and the metadata produced by the CA describing the cognitive metrics found within content generated by the instructional author. Secondly, a comparable analysis is performed on two cohorts of learners participating in a study to determine the appropriateness of using online learning content against the traditional rhetorical method of lecturing in a classroom environment.

1.3 Publications

Part of the work in this thesis has been published and presented in the publications listed in this section.

Journal publications

K. Maycock, and J. G. Keating. "A Framework for Higher Education", *WSEAS Transactions on Advances in Engineering Education*, **Issue 8**, **Volume 5**,pp. 539-548, August, 2008.

K. Maycock, and J. G. Keating. "Selection model to approximate a learner's performance prior to conducting learning experiences", *International Journal of Learning*, **Issue 5**, **Volume 13**,pp. 75–84, Jan, 2007.

K. Maycock, and J. G. Keating. "The Importance of Structure within an Adaptive Profile", *WSEAS Transactions on Advances in Engineering Education*, **Issue 1**, **Volume 3**,pp. 8–15, Jan, 2006.

Conference papers

K. Maycock, and J. G. Keating. "On-Demand Mathemagenic content for learn-

ers",*Proc. 5th WSEAS / IASME International Conference on Engineering Education*, Crete, July, 2008.

K. Maycock, and J. G. Keating. "Prototype of a learning component to maximise learning experiences", *Proc. CAL0́7: Development, Distribute & Debate*, Dublin, March., 2007.

K. Maycock, and J. G. Keating. "On-demand mathemagenic content", *Poster submission for CASCON*, Dublin, December, 2006.

K. Maycock, Sujana Jyothi, and J. G. Keating. "Dynamic profiling to enhance learning and reduce the cognitive load on each learner", *Proc. WEBIST, International conference on Web information Systems and Technologies*, Portugal, April, 2006.

K. Maycock, and J. G. Keating. "Bridging the gap between Adaptive Hypermedia Systems and the Sharable Content Object Reference Model", *Proc. 4th WSEAS Int. Conf. on E-ACTIVITIES (E-Learning, E-Communities, E-Commerce, E-Management, E-Marketing, E-Governance, Tele-Working) (E-ACTIVITIES '05)*, Miami, November, 2005.

K. Maycock, and J. G. Keating. "Building Intelligent Learning Management Systems to mimic the Teacher Student relationship ", *IEEE Learning Technology, Vol 7, Issue 1*, Washington D.C., January, 2005.

1.4 Outline of the thesis

In Chapter two traditional adaptive educational systems and strategies are investigated. In particular, these strategies are examined in terms of the process involved within the adaptive framework. The chapter investigates the lack of adoption of such systems into real world implementations. The Sharable Content Object Reference Model (SCORM) is discussed in detail. In particular the concept of granularity and learning object is investigated. Then follows a number of chapters that describe a new domain independent method of content adaptation.

Chapter three discusses the environmental contexts of an online learning environment and investigates different modes of learning that are stimulated in an online learning environment. In the same chapter, an investigation is carried out to identify suitable manifestations of cognitive traits that can be stimulated in an online learning environment to identify suitable cognitive metrics found within instructional content. This chapter also proposes a suitable personal profile that can be used to automatically evolve instructional content to suit the cognitive ability and pedagogical preference of a learner in an online learning environment.

Chapter four discusses the perceived inconsistencies found within learning object repositories and referencing standards. The chapter introduces a Content Analyser (CA) that is designed to automatically analyse instructional content and generate metadata to describe cognitive metrics within the content associated with the personal profile described in chapter three. In particular, Chapter four details a protocol for generating instructional content to enable the automatic generation of metadata. The CA automatically migrates the content being analysed to SCORM compliant learning objects packaged as independent Sharable Content Objects.

Chapter five describes in detail a Selection Model that is used to automatically generate content to suit the cognitive ability and pedagogical preference of a learner. In the same chapter, a comprehensive analysis of multiple evolutionary algorithms is explored to determine the most appropriate strategy for evolving instructional content. The complexity of the problem is investigated including a strategy for finding metrics for an evolutionary algorithm when dealing with an incomplete solution space. The Chapter also investigates the success of the evolutionary algorithm being able to evolve instructional content to a pre-determined minimum expected learning experience. Additionally, the Chapter is involved with describing a protocol for an author using the learning component and their ability to control the evolution process by establishing the minimum expected learning experience and the priority associated with the identified traits from the learners personal profile as discussed in Chapter three.

In Chapter six, an investigation is carried out to determine a suitable Learning Management System (LMS) / Content Management System (CMS) to incorporate the learning component into. This chapter also discusses the tests used to calculate a learners / authors personal profile (cognitive ability and pedagogic preference as discussed in chapter three) in detail.

Chapter seven is involved with the evaluation of the necessity and performance of the learning component. In particular, an analysis is carried out to determine the consistency of an author when generating instructional content. Additionally an investigation is carried out to determine the performance of the learning component against a traditional lecturing environment. The environmental contexts of the learning environment are discussed to ensure that no external influences disrupt the learning experience.

Finally, in Chapter eight the contributions this thesis have made to the field of adaptive learning systems are summarised. In addition, this chapter uses the learning component as a framework and suggests future possible projects that take advantage of this framework.

1.5 Concusion

This Chapter introduced the thesis research question;

Is it possible to construct an automated learning component that generates instructional content suited to the cognitive ability and pedagogical preference of a learner?.

The chapter discussed the main research papers in the fields of adaptive learning systems / adaptive techniques and working memory capacity. Further to the initial research question, the output of the desired component should not be designed by a single author; thus removing the typical problems associated with traditional adaptive hypermedia systems. The chapter also discussed at a high level the contributions of this research project to the area of adaptive learning systems. The following Chapter investigates adaptive educational systems and strategies in more detail.

Chapter 2

Theory and Background

Currently there are roughly seventy million people in higher education worldwide. This number is expected to more than double before the year 2025 to over 160 million people [35]. One possible solution to cater for the expected influx of people entering into higher education is to automate the process of learning. In an ideal situation as discussed Gilbert and Han [36] there would exist an infinite number of teachers each having their own unique pedagogical strategies so that a learner could choose a teacher that suited their own learning style. This is unrealistic practically, and will certainly increase the demand for automated personal learning efficiency. However, this is not an elementary task. If we look at the results of a number of studies carried out on the performance of individually tutored students against the performance of an average student in a typical classroom environment, we find that, the speed with which different students progress through instructional material varies by a factor of 3 to 7 [37]. An average student in a typical classroom environment asks on average 0.1 questions every hour in contrast to an individually tutored student asking on average 120 questions every hour [38]. Furthermore the achievement of individually tutored students will exceed that of classroom students by as much as two standard deviations [39] - an equivalent which is equal to raising the performance of 50 percentile students to that of 98 percentile students. These results show the vast range of differences between the learning capabilities of each learner and demonstrate that the delivery mode of the educational experience is a critical factor in producing a positive learning experience.

Learning Management Systems (LMS) like Moodle [40], Sakai [41], Blackboard [42], and Desire2Learn [43] act as a framework for educational providers to organize and deliver their instructional content in a standard way. They also offer some blended learning facilities to promote a constructivist approach to learning, for example using discussion forums. No content adaptation is taken into consideration, consequently these platforms only act to transfer the educational sector into an online environment including an easy to use interface to enable the management of educational material. Without an element of suitable adaptation embedded into these systems, these technologies could disadvantage learners as their learning would be constrained by the cognitive ability and pedagogical preference of the author of the instructional content and embedded into an organised structure environment, that also requires learners to comprehend. Other learning technologies such as Adaptive Hypermedia Systems (AHS)[7] and Intelligent Tutoring Tools (ITT) [44] [45] [46] [47] are focused on developing the learning potential of a learner. In particular, AHS are designed to adapt to the needs of the learner with respect to their domain experience, while recent ITT helps to develop cognitive skills of a learner [44]. Traditional work carried out on *intelligent tutoring* in the 70s and 80s was restricted by the computational power of the time. Buggy [45] and West [46] were involved with the identification of shortcomings in the learning experience to infer strategies for increasing the learning experience, including the introduction of stimulus to ignite the experience. Scholar [47] was involved with a highly connected network of facts, concepts and procedures to aid in computer assisted instruction. Currently processing power is not an issue and it is possible to implement strategies as seen in Buggy [45] West [46] and Scholar [47] as an on-demand strategy. Although these learning technologies have their strengths and weaknesses, they are constrained by the pedagogical preference of the author of the learning technology and are all subject to the specific system for which they are developed.

This thesis focuses on the foundation of the Advanced Distributed Learning (ADL) initiative and their production of a standardized reference model to reference instructional material as learning objects. The ADLs goal to produce the highest quality of instructional material tailored to the individual needs of each user anytime anywhere [48] is evaluated. To bridge our perceived gap between traditional adaptive learning technologies and SCORM, an explicit consideration is taken to explore the different environmental contexts of a learning experience[49]. These include the type of learning objects, the level the knowledge is being taught at and the various methods of delivering the content to the users. In addition to evaluating adaptation techniques and the environmental contexts of a learning experience, this thesis investigates the reusability of instructional content within educational repositories, such as *Multimedia Educational Resource for Learning and Online Teaching (MERLOT)* [50], *Jorum* [51] and the *National Digital Learning Repository (NDLR)* [52]. The thesis is mainly concerned with the introduction of a Content Analyser (CA) that enables an easy transformation to a single referencing standard that automatically generates metadata concerned with stimulating suitable cognitive resources within an online learning environment and a Selection model that gives SCORM conforming Learning Management Systems / Content Management Systems the capability of automatically generating instructional content to a minimum expected learning experience.

The following sections investigate the strategies and adaptation techniques used in Adaptive Hypermedia Systems (AHS) and the impact these systems have on the real world. The function of AHS is questioned in terms of wide spread use and the ability of the system to be used outside the scope of a simple project. An analysis is carried out the on the Sharable Content Object Reference Model (SCORM) to determine if that model would be suitable for referencing instructional content.

2.1 Adaptive Hypermedia Systems

Adaptive Hypermedia systems have been in development since the early 1990s [7]. Despite the vast amount of research conducted in this area, there has been a lack of adoption into real world systems. Reasons for this include: high cost of production, lack of credible evidence to support the cost or benefit, and limited subject matter as discussed by Murray [53]. They extend the one-size fits all [54] approach of hypermedia systems by using personalisation strategies to adapt content to suit a given learner. Typically AHS [17][20][7][24][19][34][55] [56] operate on a closed world model, whereby, all the hypermedia is annotated prior to a learning experience and the adapted strategy is already defined by the author of the adaptive system. Eklund [57] distinguished two categories of features within a hypermedia system suitable for adaptation: content adaptation and navigation adaptation. Adaptive navigation techniques such as direct guidance, adaptive hiding or re-ordering of links, link annotation, map adaptation [58], link disabling and link removal [59] can be used to control both the size and level of the instructional space available to each learner. Adaptive content presentation operates at the domain level. The information can be adapted to various types of media and difficulty to meet the needs of each user. Adaptive systems typically alter the navigation model or the presentation model for the content. They build a model of the users preferences, goals and knowledge and use this model throughout the interaction with the user.

In constructing any AHS there are three main components: the knowledge space, the hyperspace and the student model. The knowledge space represents a collection of knowledge elements which represent individual concepts. Typically the first step in building an adaptive hypermedia systems model is to annotate the instructional space according to some adaptation strategy. The simplest construction of the knowledge space is an unconnected scatter of knowledge elements. The most common type of link is a pre-requisite link giving the author of an AHS the ability to make sure that a concept is known before the student moves onto the next concept. Semantic links have also been applied to different AHS. The hyperspace represents the actual content, which is available to be presented to the user. Using some form of mapping, a mapping is created between the knowledge space and the hyperspace. The student model represents the preferences, goals and knowledge of each user. A mapping is also created between the student model and the domain knowledge elements in the knowledge space.

AHS are very useful in any application area where users of the hypermedia system have essentially different goals and knowledge and where the hyperspace is reasonably large. AHS overcome this problem by using information stored in the user model to adapt the information and links being presented to the given user. Although AHS and similar learning technologies have their strengths and weaknesses, they are constrained by the pedagogical preference and cognitive ability of the author of the adaptive learning technology. Additionally, traditional AHS citePBAdaptiveCourseware[20][7][24] are constrained within the application that they were developed for, while more recent AHS [19] are build upon Java taking advantage of platform independence and can be reused on any machine with the same environment. The main problem associated with AHS and the lack of wide spread adoption into the real world is that the systems are usually designed and created by an author of instructional material and thus the AHS is constrained to the cognitive ability and the pedagogical preference of that author. Chapter seven shows the inconsistencies found with instructional authors in terms of the cognitive metrics that are typically found within instructional content, when a number of authors were required to generate a number of learning objects.

2.1.1 Summary

In summary, AHS despite their great interest and research in the area, have not seen wide spread adoption into the real world. This lack of adoption is due to the unproven benefits of AHS, poor or inconsistent implementations and the systems being constrained to the cognitive ability and pedagogical preference of the author of the adaptive system. However, with the introduction of specifications like SCORM, enhanced adaptive content presentation is possible given the fine granularity of learning objects. The following section discusses the Sharable Content Object Reference Model (SCORM) in detail and particularly how this reference model can be utilised to create content suited to content adaptation.

2.2 Sharable Content Object Reference Model

The Advanced Distributed Learning (ADL) initiative was established in November 1997. The ADL team were established to bring an element of consistency to the online learning arena, their mission statement is as follows:

"our mission is to bring the highest quality of instructional material, tailored to the needs of each individual anytime anywhere"[48]

The ADL produced the Sharable Content Object Reference Model (SCORM) as the backbone for producing reusable-learning objects. Each learning object is fully described and delivered within a content package as seen in Fig 2.1. The content package consists of a manifest. The Manifest, consists of: Metadata, Organizations, Resources and Sub Manifests. The metadata section is used to describe in full the version of SCORM and type of content being delivered. The Organisations section details the sequencing information of the various learning objects that are encapsulated within the content package. The Resources section is fully described using XML metadata elements to describe the content that is being delivered. Sub manifests can also be used to create structured courses with different layers of depth. Physical Files can also be stored locally within the manifest.

SCORM is built on the proven work of prominent organisations such as: Aviation Industry CBT (Computer-Based Training) Committee (AICC)[60], Innovation Adoption Learning (IMS)[61], IEEE Learning Technology Standards Committee (IEEE LTSC)[62] and ARIADNE[63]. The ADL captured the best components of all the previous standards and used this as a framework for producing SCORM, as seen in Figure 2.2. The goals of the ADL team were to:

- *Identify and recommend standards for training software and associated services purchased by Federal agencies and contractors.*
- *Facilitate and accelerate the development of key technical training standards in industry and in standards-development organizations.*

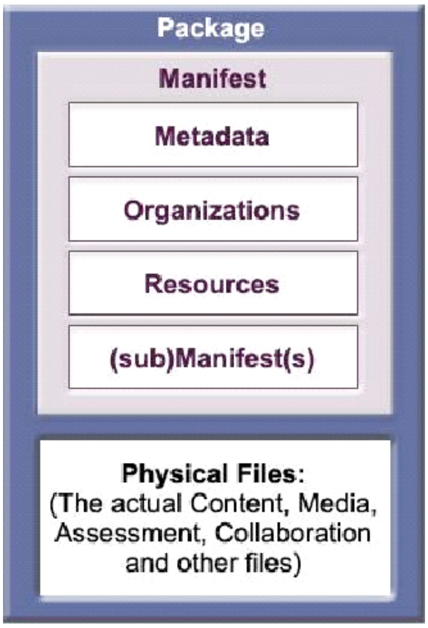


Figure 2.1: Components of a SCORM content package

Figure 2.2: Organisations and Standards that SCORM built their model on.

- *Establish guidelines on the use of standards and provide a mechanism to assist DoD and other Federal agencies in the large-scale development, implementation, and assessment of interoperable and reusable learning systems.*

The SCORM reference model bridges the gap of technological obsolescence with various LMSs / CMSs / VLEs. SCORM provides an API (Application Programming Interface) to allow content authors to create instructional content to monitor the flow of a learning experience. Most widely used learning environments (Moodle, Blackboard ...etc) conform to the SCORM reference model as a SCORM component with additional support for other referencing standards. A learning environment interacts with the SCORM throughout a learning experience. The learning environment contains software that automates training event administration through a set of services that; launches learning content, keeps track of learners progress and sequences learning content.

Assets and Sharable Content Objects (SCOs) exist within the SCORM. An asset can represent anything from a text file to an image or a sound file. A SCO can be represented as one or more assets that must contain at least one particular asset that utilizes the SCORM RTE (Runt Time Environment), hence a SCO represents the lowest level of granularity that can be tracked by a Learning Management System (LMS). By aggregating assets and SCOs together, courses and lessons can be generated as seen in Figure 2.3.

Figure 2.4 illustrates a graphical representation of the reusability of assets and SCOs aggregated from raw data elements into complete courses. It can be seen that the reusability of the learning content decreases with an increase of context in the learning content. Recommendations from the ADL team on levels of granularity of instructional object are suited to producing content without any context as

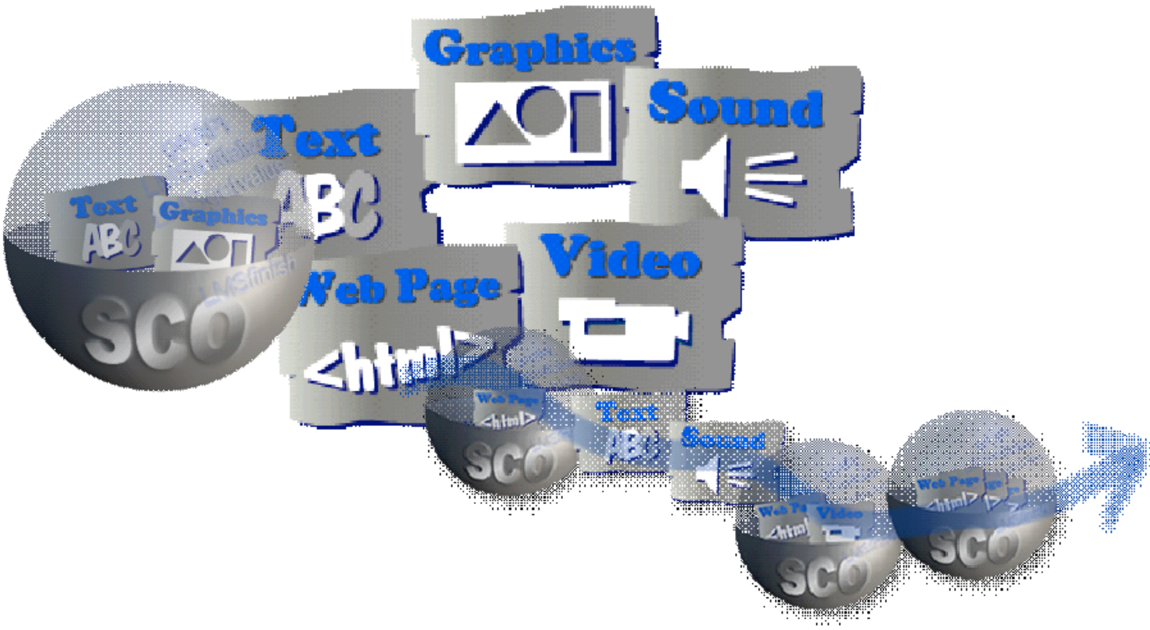


Figure 2.3: Aggregation of SCOs and Assets forming courses

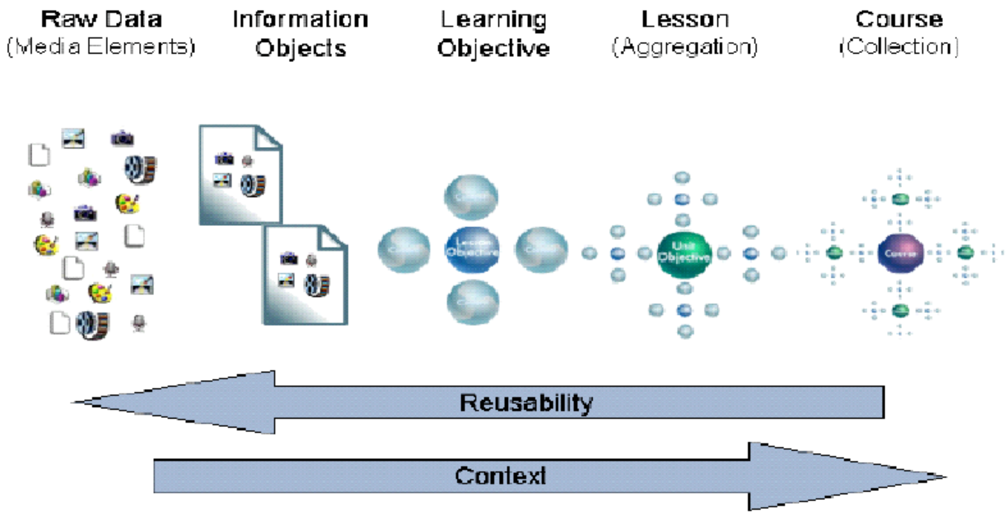


Figure 2.4: Reusability of Assets and SCOs

this would inherently increase the usability of the instructional content. However, the granularity of the instructional content should demonstrate a single concept and contain some context describing the concept. Various strategies associated with the granularity of learning content and the possibility of adaption exists. In particular, Laurillard [13] argues that the structure of the learning content embodies the meaning of the content, in contrast to the ADLs best practice of designing content with no context to increase the reusability of the content. However, in an online learning environment to produce an adaptive system avoiding the traditional frameworks of AHS (whereby an author of instructional content would generate a number of paths through some instructional space), the structure of the content needs to be modified. This process can be achieved by appropriately generating instructional content at a granular level to embody some context with associated metadata. If all the content was structured in such a way, simple adaptation could be performed by strategically swapping SCOs or elements within SCOs depending on a learner interacting with the learning environment and the availability of suitable instructional content.

The SCORM is defined within three books: The Content Aggregation Model (CAM), SCORM Run Time Environment (SCORM RTE) and the Sequencing and Navigation (SCORM SN).

- The CAM defines the learning content using specified metadata elements to ensure:
  - that the components are packaged in a suitable organisation for transport from system to system.
  - that adequate metadata is used to enhance the possibility of search and discovery in order to maximize the reuse of granular learning objects.

  - to ensure that suitable sequencing and navigation strategies are described to enhance the learning experience.

- The SCORM RTE ensures maximum portability of SCORM compliant learning content. The RTE contains support to allow an author to create functionality to monitor the progress of the learner throughout an instructional experience on multiple platforms.

- The Sequencing and Navigation (SN) defines the various methods of delivering courses to clients. Within the SN, SCORM defines four control modes for delivering of instructional content:

  - User choice
    - The learner is able to choose any learning object within the information space.
    - This type of learning would suit a holistic learner enabling the learner to freely navigate through the learning space.
  - Flow navigation
    - The Learning Management System (LMS) determines the next activity to deliver with respect to the learners navigation request.
    - This type of learning environment would suit an atomistic learner constrained by their interactions with the learning environment.
  - Choice exit
    - When disabled, the learner cannot choose another activity while the current activity is still in progress.
    - This type of learning environment could be implemented in conjunction with user choice or flow navigation to make sure that all activities are completed.

    * It could also be useful when a content developer is delivering formative assessment for a particular activity.

  - Flow navigation

    * Restricts a learner from revisiting a previously visited learning object (user choice must be disabled).

2.2.1 Summary

The SCORM standard has been developed using the best practices of previous organisations such as Aviation Industry CBT (Computer-Based Training) Committee (AICC)[60], Innovation Adoption Learning (IMS)[61], IEEE Learning Technology Standards Committee (IEEE LTSC)[62] and ARIADNE[63] and is controlled and defined by data model elements which are monitored by the SCORM RTE. The granularity of the learning objects is critical to the reuse of the instructional content. The characteristics of the SCORM model as defined above make it an ideal candidate for wide spread adoption of an automated learning component that utilises the model as its referencing standard. Many current LMSs contain a SCORM RTE , for example [40], [41], [42], for running SCORM content. Developing the automated learning component around the SCORM would enable easy migration of adapted content into any LMS that contains a SCORM RTE, thus adhering to the second goal of this research project.

2.3 Conclusion

This chapter investigated the potential of AHS and explored the lack of wide spread adoption in real world systems. Furthermore the chapter discussed in detail the SCORM referencing model used for referencing instructional material. In conclusion, AHS have been researched for many years but are still evolved around a static model based loosely on the interpretation of an author of instructional material and are typically constrained to the cognitive ability and pedagogical preference of the author of the system. AHS require huge resources in their creation from identifying suitable adaptation strategies and annotation styles to mapping these to suitably defined student models associated with the domain in which the AHS is involved with. AHS in their current format will remain a research topic with little chance of wide spread adoption due to the constraints already discussed. The chapter also discussed the Sharable Object Content Reference model that is used to reference instructional material and the possibility of using this model to build an automated learning component that is capable of generating adaptive content across multiple platforms. The following chapter investigates the creation of a personal profile that could be used to automatically generate instructional content. In particular the chapter is concerned with the identification of suitable cognitive traits and pedagogic preferences that have an associated cognitive metric within instructional content that can be automatically identified. This strategy will remove the reliance on metadata inconsistencies found within learning object repositories such as: MERLOT, Jorum and NDLR, and overcome the black-box problem associated with metadata creation.

Chapter 3

Optimal Personal Profile

One of the problems with most adaptive educational systems is that authors of educational material are likely to have different ideas on the best teaching practices, which can hinder the development of a learners learning experience. Additionally instructional authors have their own cognitive ability and pedagogical preference which would impede the learning experience of some learners. This thesis is involved with the construction of a learning component that is capable of generating instructional content suited to the cognitive ability and pedagogical preference of a learner, to be able to produce mathemagenic content for any learner using the learning component independent of domain. This chapter is focused on the creation of a personal profile that could be used to automatically generate instructional content. In particular the chapter is concerned with the identification of suitable cognitive traits and pedagogic preferences that have an associated cognitive metric within instructional content that can be automatically identified. Once suitable traits are identified these can be used as a framework for automatically generating metadata associated with the personal profile of a learner, thus avoiding the black-box method for metadata creation as discussed in Chapter 2.

Initially the chapter is concerned with understanding the environmental contexts of a learning environment and creating a mapping of these contexts to a suitable adaptive strategy. Additionally, this chapter investigates current adaptive strategies used for increasing the potential learning experience and reducing the possibility of interference occurring within the learning environment. Finally the chapter introduces a Personal Profile that is used as the underlying framework for the learning component.

3.1 Environmental contexts of a learning environment

Most student models are focused on the specific domains with which they interact with, for example, the domain concepts competence and domain skills required. Such student models are referred to as performance based student models and include the student competence state models [64] and process state models [65]. To create a truly adaptive learning environment across multiple domains suitable cognitive traits and pedagogic preferences of a learner should be catered for and mapped to the environmental contexts of a learning environment. These contexts include the nature of the subject discipline and the level of its learning; the characteristics of the learning material and the role of the human teacher [18]. Support should also be available for dealing with a learners learning profile. The profile should consist of the entire learners educational history, cognitive ability and pedagogical preference.

The teacher plays various roles in an educational system including providing learning objects, selecting and scheduling other learning technologies, managing the curriculum and overseeing the learners progress through instructional material. A serialist teacher may feel more enthusiastic about a tightly constrained educational system designed on the building blocks metaphor, while a holist teacher may be motivated by a loosely constrained educational system that allows zooming in and out of fine grained details. Similarly a pragmatist teacher may prefer a focus on practical applications while a theorist teacher may prefer logical analysis [18]. Developing an educational system around the SCORM would easily be able to overcome the problem of the teacher being in full control of the learning experience in terms of learning object delivery. SCORM SN as detailed in Chapter two, describes multiple modes of suitable delivery options to suit various categories of learners (for example, a holistic learner would have a User Choice sequencing enabled).

A learning style is defined as the unique collection of individual skills and preferences that affect how a student perceives and process learning material [66]. The learning style of a student will affect the potential of the outcome of the learning experience. Research has been carried out for decades on defining and classifying learning styles. Many of these theories are in practice today, for example, the Theory into Practice Database [67] provides 50 major theories of learning and instruction, such as Kolbs learning style theory [68], Gardeners Multiple Intelligence theory [69], Felder-Silverman Learning style theory [70], Litzinger and Osif Theory of learning styles [71], Myers-Briggs Type indicator [72]. There are many existing systems that are able to adapt to students learning styles, for example [73][36][74] , however these systems are constrained to the domain in which they were developed for.

.

3.1.1 Summary

In summary, creating an automated learning component that automatically generates instructional content suited to the cognitive ability and pedagogical preference of a learner, the environmental contexts of the learning environment should be taken into consideration. Identifying suitable cognitive traits enables the production of a general learning component with the desired adaptive functionality that is independent of domain knowledge. Unlike traditional adaptive hypermedia systems [17][20][7][24][19][34][55][56] once the traits of the learner have been identified the model can be used across multiple domains.

The following section discusses suitable adaptation strategies that are independent of domain. In particular, two well known strategies for reducing the cognitive load on a learner: Multiple Representation Approach (MRA) [24] and Exploration Space Control (ESC) [75].

3.2 Adaptation independent of domain

To create a truly adaptive learning environment across multiple domains the cognitive ability and the pedagogical preference of a learner should be taken into consideration (see Maycock et al. [76]). Successful adaptation requires some correlation between the environmental contexts of a learning environment and the personal profile of a learner. These environmental contexts include the type and delivery protocol of the learning content. Brusilovsky [17] distinguished two categories of features within a hypermedia system suitable for adaptation: content adaptation and navigation adaptation. Adaptive navigation techniques such as direct guidance, adaptive hiding or re-ordering of links, link annotation, map adaptation [58], link disabling and link removal [59] can be used to control both the size and level of the instructional space available to each learner. Adaptive content presentation operates at the domain level. The information can be adapted to various types of media and difficulty to meet the needs of each user. However, with the introduction of specifications like SCORM, enhanced adaptive content presentation is possible given the fine granularity of learning objects. In addition to theses strategies two main techniques are used; Multiple Representation Approach (MRA) [24] and Exploratory Space Control (ESC) [75] can be used to fine tune learning experiences. The following section details the advantages of both techniques and discusses how these techniques are incorporated into our proposed learning environment architecture.

3.2.1 Multiple Representation Approach

MRA is used to change the presentation of domain knowledge concepts, in terms of the complexity and granularity, to suit the learners cognitive ability and progress through a learning experience. It enhances the educational systems design to suit the learners perspective. There are various types of multimedia objects, each stimulating different cognitive responses. Audio stimulates imagination, video clips stimulate action information, text conveys details and diagrams convey ideas [16]. Generating MRA compliant learning objects in a learning environment can reduce the cognitive load by using similar multimedia objects to convey domain concepts. If any media objects are omitted during the MRA process they must be available to a user on specific request, reducing the possibility of losing any relevant information. There are three different types of filtering used in MRA: *restriction*, *extension* and *approximation*. Restriction is used when a learning object contains an excessive number of media objects, thereby causing cognitive overload. A subset of these media objects may be selected to produce an MRA compliant learning object conveying the current domain concept. If several different MRA compliant learning objects are available then the combination of media objects offering the best learning experience suited to that learners cognitive ability may be selected. When the number of media objects is insufficient to produce an MRA compliant learning object, extension may be used. An LMS will search learning object repositories to find suitable learning objects that will enhance that learning object and make it MRA compliant. If a learning object was poorly designed, and the complete learning object cannot be made MRA compliant, the largest multimedia rich subset is selected. The process of extension is then carried out on the reduced learning object.

MRA is a great concept; delivering different learning objects to individuals based on the learners personal profile. However, it is argued by Laurillard [13] that the structure of the learning content embodies the meaning of the learning content. It should not be possible for an adaptive learning environment to change the structure of learning content thereby potentially changing the meaning of the content and subsequently changing the potential learning experience. However, if enough learning objects exist and are created suited to the granularity level described in Chapter two then multiple modifications can occur without impacting the on the learning experience.

3.2.2 Exploration Space Control

ESC limits the learning space to reduce the cognitive load of each learner and to make sure that learners do not get lost in hyperspace [75]. In our proposed system, ESC is used in the exploration of further reading once a learning experience has concluded. The exploration elements catered for are the learning content and navigational paths. When dealing with the learning content, the ability of the student to interpret the content exactly as the content developer expected, is a very complex task and depends on the learners cognitive ability. There have been many studies carried out on how learners perceive instructional material, in particular, Phenomenography (Laurillard, 2002) (Marton and Booth, 1997) (Ramsden, 1988) (Ramsden, 1998). This is successful at illuminating how students deal with structure and meaning. These studies have led to the identification of two contrasting approaches to studying content, i.e. an atomistic approach and a holistic approach. Learners utilizing a holistic approach interpreting some content retain the concepts that are trying to be conveyed but may suffer some cognitive overload. Learners utilizing an atomistic approach lose the structure of the content being delivered, hence, may have a different interpretation to the actual meaning.

3.2.3 Critique of adaptive strategies

Kinshuk et al. [75] believe that the reduction of sensory resources describing an instructional object depends on the ability of a learner. In 1956 however, Miller [3] reviewed the current research to determine the Working Memory Capacity (WMC) of an individual and found that an individual could store between 5 and 9 items in their WMC for one-dimensional content. It was also discovered that when the number of dimensions describing the content increases, the amount of items that can be stored in the WMC of an individual increases exponentially. An adaptive learning environment should not reduce the number of dimensions, potentially the WMC of a learner, throughout a learning experience. The Virtual Learning Environment (VLE) could enhance the learning experience by ensuring that multiple modes of learning are simultaneously stimulated throughout a learning experience. Table 3.1, adapted from [75], shows how resources in a learning environment can be adapted to suit the cognitive ability of a learner and in particular shows the relationships between WMC and Information Processing Speed (IPS).

Resources / Cognitive Traits	WMC	IPS
	H / L	H / L
Paths	+ -	+ -
Path Relevance	- +	- +
Amount of Info.	+ -	+ -

Table 3.1: Relationship between Working Memory Capacity and Information Processing Speed

In Table 3.1 the "+" symbol indicates an increase in the number of resources to adapt to the cognitive ability, and the "−" symbol indicates a decrease in the number of resources to adapt to the cognitive ability. If a learner has been categorised to have high WMC then for the purposes of adapting to the number of paths, relevance of paths and the amount of information, the learner would be classified to having high IPS. Similarly, if a learner has been categorised as having low WMC then for the purposes of adapting to the number of paths, relevance of paths and the amount of information, the learner would be classified as having low IPS. Content developers are responsible for producing small granular learning objects that adequately describe a domain concept. Each learning object that is created should take into consideration the different types of media and their optimal effect on a learning experience.

3.2.4 Summary

This section discussed some of the contradictions found within the adaptive strategy research. In particular, Kinshuk [75] proposing to modify the structure of the content as an attempt to reduce the cognitive load of the learner is in stark contrast to Laurillards understanding that the structure of the content also embodies the meaning [13]. The learning component that was developed within this research project utilises the SCORM model and the granular structure of that model to produce content. Unlike a closed model typically found within AHS the learning component starts with no content and stitches a course together. Extending the initial research question;

Is it possible to construct an automated learning component that generates instructional content suited to the cognitive ability and pedagogical preference of a learner?

to include,

independent of domain and ensuring that no meaning is lost from adaptive strategies.

The following section investigates *Working Memory Capacity*, as an appropriate cognitive trait that could be used within our personal profile. Both traditional and modern research on the limitations and functionality of WMC is discussed. In addition, trackable manifestation of WMC are discussed in order to identify automatic strategies that could be used to calculate the WMC of a learner utilising the learning component without using pre-diagnostic testing methods.

3.3 Working Memory Capacity

Working Memory Capacity also known as Short-Term Store (STS) facilitates temporal storage of recently perceived information, allows active retention of a limited amount of information, (7 +/- 2 items), for a short period of time [3]. Since

Millers early investigation of memory having a limited amount of space for immediate storage and the possibility of greatly increasing this capacity by introducing extra dimensions into the learning material there has been a vast amount of research conducted. Especially through the introduction of neural imagery research, which has been able to identify specific regions of the frontal cortex associated with temporary memory. Coming from the vast amount of research, three main models for working memory capacity have emerged: Baddelys model [4], Cowans Model [77][5] and the theory of Ericsson and Kintsch [9].

3.3.1 Baddeley Model

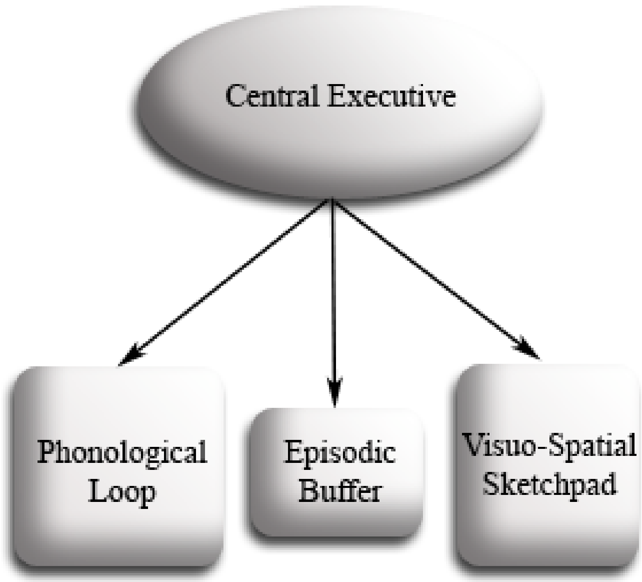


Figure 3.1: Schematic of Baddeley's Model

Alan Baddeley and Graham Hitch [4] introduced a multicomponent model for working memory in 1974. This model is composed of two slave systems and a central executive system to control the flow of communication between the slave systems and for coordinating cognitive processes when more than one task must be completed at one time. The slave systems consist of a phonological loop and a visuo-spatial sketch pad. The phonological loop stores phonological information and prevents the decay of such information by constantly refreshing the information. The visuo-spatial sketch pad is used to store visual and spatial information and is used for the construction and manipulation of visual images. The sketch pad can be broken down further into two subsystems: a visual subsystem, responsible for shape, colour and texture and a spatial subsystem dealing with location. Baddeley [12] furthered his model in 2000 by introducing an episodic buffer as an additional component. This buffer represents a temporal storage of phonological, visual, spatial and semantic information. The buffer is comprised of a limited capacity system that provides temporary storage of a multimodal code, that is the binding of information from the initial subsystems and long-term memory. The key characteristics of the new model focuses attention on the process of integration of information rather than viewing the sub-systems in isolation. This new model emphasis the importance of creating the link between the long-term memory and the sub-systems. Figure 3.1 represents a schematic of Baddeleys model.

3.3.2 Nelson Cowan's Model

Cowan [5] regards working memory as part of long-term memory and not another component. Representations in working memory are a subset of the long-term memory. Working memory consists of two distinct levels. The first level consists of long-term memory representations that are activated. There is no limit to activation of representations in long-term memory. The second level is described as the focus of attention. The focus is regarded as capacity limited and can hold up to four of the activated representations at any given instant. This view of working memory is thus centered on the concept of monitoring the focus of attention and reducing the possibility of interference with the focus of attention throughout a learning experience.

3.3.3 Ericsson and Kintsch

Ericsson and Kintsch believe that there exists two structures within memory: working memory and long-term memory. However, they argue that there must exist some retrieval structures to allow for the expansion of working memory during certain conditions. They classify this expansion as having the ability to utilise Long-Term Working Memory (LTWM). For example, text comprehension requires all the following to take place: perceptual features, linguistic features, propositional structure, macrostructure, situation model, control structure, goals, lexical knowledge, frames, general knowledge and episodic memory for prior text [11]. Each of these components by itself would exceed the capacity of short-term working memory, but is clearly needed in text understanding. Kintsch et∼al [9] consider that every reader is able to form an episodic text structure during text comprehension, if the text is well written and the content is familiar. Forming an episodic text structure allows the use of long-term working memory thus explaining how a complex process like text comprehension can be performed on a daily basis. Guida et∼al [14] [9], using the theory of text comprehension have proposed the personalisation method as a way to operationalise the LTWM.

3.3.4 Trackable Manifestations of WMC

The personal profile that is required for this research project must include the cognitive ability and pedagogic preference of a learner that can be used both as the driving framework for automatic metadata generation (i.e. profile that includes cognitive traits that have associated cognitive metrics that can be identified within instructional content) and appropriate manifestations to enable the automatic generation of a learners profile by interacting with a learning environment. Working Memory Capacity (WMC), as described above, contains the following manifestations:

- Constantly revisiting learned materials very shortly indicates signs of low WMC [78].
- People with a greater tolerance to interference have higher WMC [79].
- Frequently missing steps or losing components during a long sequence calculation or procedure indicate signs of low WMC [80].
- Working Memory is known to vary with age [81].
- For learners with high WMC it is likely that they will follow the curriculum sequentially, thereby reducing the number of trans-state violations [82] [83], for example, moving to an unexpected state.

3.3.5 Personal Profile model

All three models of working memory have been subject to great acclaim however, they are all trying to categorise the same cognitive process and have all completely different interpretations of the same process. Cowans model is centered on the idea that working memory is not disjunct from long term memory but is split into two separate components. Cowans model is limited in capacity just like George Millers from 1956. Ericsson and Kintsch believe that there exist two separate components within memory: working memory and long-term memory. Their main distinctiveness is in the underlying process at which information is retrieved from long-term memory, which describes certain conditions that enable an expansion of working memory. Finally Baddeley describes a multicomponent model for working memory, consisting of two slave systems and a central executive for transporting information between the slave systems.

The thesis is focused on investigating the possibility of constructing an automated learning component that generates instructional content to suit the cognitive ability and pedagogical preference of a learner independent of domain. To evaluate the learning component suitable metrics must be identified to create a personal profile, however the learning component should be independent of the metrics selected and extensible to any pedagogic strategy requirement. In order to establish a suitable personal profile for testing the learning component, the environment in which the learning takes place must be categorised and understood. The profile should include the cognitive ability of the learner to ensure that adaptation can occur across multiple domains. Cattel-Horn-Carroll definitions project is involved with the classification of a taxonomy of human cognitive abilities, in terms of broad and narrow categories [84]:

- Auditory Processing
- Fluid Intelligence / Reasoning
- General (domain specific) knowledge
- Kinesthetic Abilities
- Long-term Storage and Retrieval
- Olfactory Abilities
- Psychomotor Abilities
- Psychomotor Speed

- Reading / Writing Abilities
- Short-term Memory
- Tactile Abilities
- Visual-spatial Abilities

Taken the environmental contexts of the learning environment into consideration as defined in this Chapter, these categories are reduced to the following categories:

- Auditory Processing
- Fluid Intelligence / Reasoning
- General (domain specific) knowledge
- Long-term Storage and Retrieval
- Reading / Writing Abilities
- Short-term Memory
- Visual-spatial Abilities

Additional reductions can be applied to the list of categories: the personal profile should be independent of domain, the effects of robotic voices on online learning environments is unknown, however it can be assumed that there would not exists enough robotic voices to suit each individual learner, consequently placing some learners at a disadvantage using the learning component and Fluid reasoning was also eliminated as it is associated with mental operations to solve problems and would be deemed more suitable to specific domains or gaming applications. The reduced set of categories is defined as the following:

- Long-term Storage and Retrieval

- Reading / Writing Abilities
- Short-term Memory
- Visual-spatial Abilities

The personal profile needed to be categorised as metrics that could be identified automatically in instructional content for the generation of suitable metadata. The VARK element represents the visual-spatial category, as the learning environment conducts learning experiences within an online learning environment the VARK learning style is restricted to suit the visual constructs of the learning unit. The Long-term Storage and Retrieval category / Long-term memory is removed as the learning component will initially generate content that is independent of educational history. This category would have great benefit when considering the associative learning skill of the learner, however as there does not exists enough learning experiences from each student the associative learning skill cannot be used. The reading / writing ability category is defined by the readability level and the information processing speed of a learner. These elements along with the working memory of learner identify the constructs for determining a *chunk* when interacting in an online learning environment. In particular the readability level of instructional content is used as a minor indicator of the suitability of instructional content for a given learner.

All three above models for working memory have components that can be generalised and reused in an online learning environment. Our proposed personal profile consists of: working memory capacity, pedagogic preference of a learner, information processing speed and the readability level of the learner. The proposed personal profile is thus mainly categorised into two categories: working memory and pedagogic preference. Unlike Ericsson and Kintsch theory on WMC, forming episodic text structures may increase the working memory capacity, however, could impede on the potential learning experience as the underlying structure of a module encapsulates the learning outcomes [13] and should not be decoupled (but could be modified) to potentially increase the working memory. If the language used in the learning material is of a comparable standard to the learners, the instructional space limited to suit the working memory capacity and if the delivery protocol is directly related to the pedagogic preference of the learner, it should be possible to determine the expected minimum learning experience prior to conducting the learning experience.

The metrics that describe the element of the personal profile are:

- Working Memory Capacity
- Readability Level
- Information Processing Speed
- VARK

3.3.6 Summary

In summary, this section investigated WMC, from Millers work in 1956 [3] up until the work of Baddely[4], Cowan and Ericson and Kintsh [9]. All three recent above models for working memory have components that can be generalised and reused in an online learning environment. The proposed personal profile for the learning component consists of: working memory capacity, pedagogic preference, information processing speed and the readability level of the learner. The proposed personal profile is thus mainly categorised into two categories: working memory and pedagogic preference. If the language used in the learning material is of a comparable standard to the learners, the instructional space limited to suit the WMC and if the delivery protocol is directly related to the pedagogic preference of the learner, it should be possible to determine the expected minimum learning experience prior to conducting the learning experience.

3.4 Conclusion

This chapter described a suitable personal profile that could be used by the learning component to automatically generate mathemagenic content for each learner. In particular, the chapter was focused on identifying suitable adaptive strategies independent of domain knowledge. The personal profile for the learning component was described. The following Chapter details a protocol to bridge the perceived gap between the inconsistencies found in repositories and instructional content within the repositories. Chapter four also investigates the use of SCORM as a referencing standard and discusses statistics that yield a lack of consistency when referencing instructional material. This leads to the analysis and development of a Content Analyser that automatically generates SCORM compliant instructional content with additional metadata describing the cognitive metrics found within the instructional content to avoid using a closed loop system like traditional AHS.

Chapter 4

Content Analyser

There exist many instructional content repositories, for example, *Multimedia Educational Resource for Learning and Online Teaching (MERLOT)* [50], *Jorum* [51] and the *National Digital Learning Repository (NDLR)* [52]. These repositories contain various types of instructional content including text files, word documents, PDF documents, presentations, complete SCORM packages, SCOs etc... Metadata can be defined as data describing other data and is typically produced external to the creation of instructional content in a black-box fashion. This method of metadata generation is insufficient as no guarantee exists between the actual content and the metadata describing the content. Furthermore it was found by Norm Freisen [2] that only 57% of content authors complete keywords within Learning Object Metadata (LOM) files associated with SCORM content, consequently this results in a large amount of learning objects with insufficient metadata, for search and discovery.

In general, the goal of creating suitable metadata is to allow a process to identify your instructional content for reuse. Metadata associated with a learning object should be designed in such a way, firstly, to be easily recognisable as the instructional content in terms of domain specific searches (domain relevance), and secondly the metadata should reflect cognitive stimulus required for interacting with the learning object in an optimal learning experience. Without metadata reflecting the internal design of the instructional content it would be impossible to develop a reliable automated process for content adaptation. Neither of these conditions are common practice, thus resulting in inconsistencies within learning object repositories and insufficient consistent metadata for search and discovery.

The Content Analyser (CA) is focused on bridging the perceived gap between repositories, standards and inconsistency of learning objects. The CA was designed to automatically generate metadata for some instructional content that stimulates the cognitive traits and pedagogic preference of each learner (as discussed in Chapter three), thus addressing the second condition stated above. The following section explores the protocol of the CA in detail. A complete example illustrating a sample piece of instructional content is described and the metadata that was produced is examined to reflect the cognitive metrics found within the instructional content.

4.1 Inside the Content Analyser

The CA was designed to automatically generate metadata for some instructional content that stimulates the cognitive traits and pedagogic preference of each learner. The CA takes as input some instructional content (.txt files, .doc files, .html files or .zip files), decouples the content and generates Sharable Content Objects (SCOs) with added metadata to describe the type of information, the amount of information, the size of the instructional space, the readability level of the content and the VARK representation of the instructional material. These metrics form the foundation of the evolutionary process described in Chapter five to evolve instructional content to suit the needs of a learner.

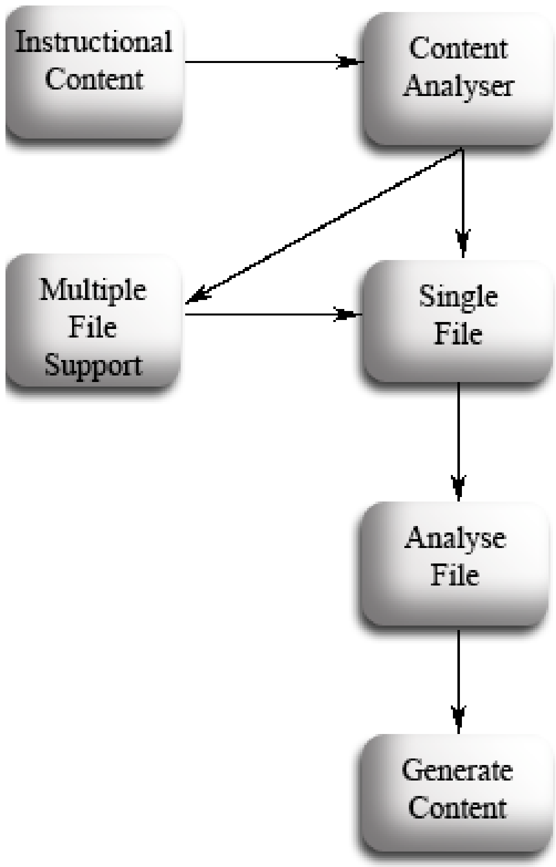


Figure 4.1: Content Analyser

Figure 4.1 depicts a simple protocol for the Content Analyser (CA). Instructional content is inputted into the content analyser, either as a single file submission or as an archived package. If an archived package was inputted into the CA, the package is decoupled and each file is treated as a single file entry. The CA automatically generates metadata for each file describing the cognitive resources and type of information required to stimulate the personal profile of a learner as described in chapter three. Each file is repackaged as a Sharable Content Object (SCO). The CA uses the Java Open Document (JOD)[85] JAR files to interact with OpenOffice running as a background process listening on port 8100, to allow for easy file transformation between multiple file formats. This is essential for calculating the metadata associated with the learning objects.

The following subsection details the constraints involved when generating instructional content. These constraints are designed to reduce the computational complexity time of automatically generating metadata for instructional content and act as guidelines for content authors.

4.1.1 Developing compatible content for the CA

Instructional content is taken into the content analyser and SCOs are produced with associated metadata to stimulate the cognitive ability and pedagogic preference of a learner. The following list represents the constraints when developing the instructional content:

- The final course that is outputted to the learner is constructed from a repository of instructional content and not from external sources. When developing a course the typical demographic of the learner should be taken into consideration in terms of their educational competence. The author should not presuppose any educational background that is not mentioned within the specification.
- The input for the Content Analyser can be a single File submission or a .zip submission. When submitting a zipped package the relevant path should not be stored when creating the archive as seen in Figure 4.2.

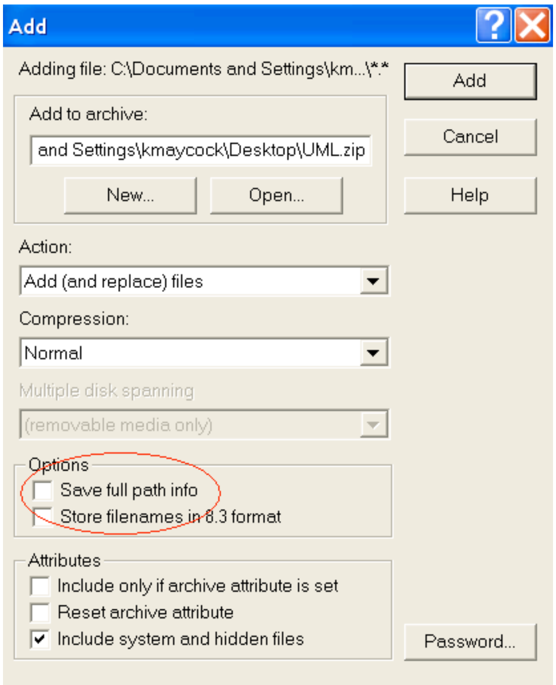


Figure 4.2: Excluding the relative path information

- A separate file should be generated for each learning object described in the specification.
- It is recommended to use external links when constructing the content. External links should act to either explain a concept in greater detail or to further strengthen a concept. There can be a maximum of three links per learning object. Each link must start with *"link_"* followed by the link name, example, *link_Mylink*.
- The main learning objects should be written as html documents. External link can be either, .txt, .doc or .html documents. It is recommended that all graphics be PNGs but it is not essential. When including an image in your learning objects you should use the following syntax; *"WIDTH="* and *"HEIGTH="*.

- Images should be supported by textual information relating to the image. The textual information explicitly relating to an image should be referenced as follows within the main learning object, <p ID="imageName">textual information </p>.

- When creating visual constructs such as **boldface** you should leave a gap between the last element of the visual constructs and the termination symbol, as the regular expression for a word is one or more alpha characters followed by a white space or a line termination symbol.

- When creating the instructional content time should not be spent creating complex background designs as the instructional content will be stripped of formatting constructs and reconstructed to suit the pedagogic preference and cognitive ability of the learner.

4.1.2 Summary

In summary, this section introduced the Content Analyser (CA) that was used to bridge the percieved gap between learning object repositories and the inconsistencies found within metadata standards. The CA uses the JOD libraries [85] to allow for multiple file formats to be included within the instructional content. The CA uses the SCORM file format as the default output after analyzing files. In addition, this section discussed the protocol for generating instructional content compatible with the CA and listed some constraints imposed on authors of instructional content. The following section details the metrics found within instructional content suited to the personal profile discussed in Chapter three. Additionally the following section discusses an automatic process for generating metadata from instructional content and details the advantages this process has over a traditional black-box method for metadata creation.

4.2 Stimulating Cognitive Resources

Metadata for describing instructional content is typically created external to the instructional content in a black-box fashion. This method of metadata generation is insufficient for automated content generation as no guarantee exists between the actual content and the metadata describing the content. The CA automatically produces metadata to describe the cognitive metrics found within instructional content suited to the personal profile described in Chapter three. In addition to identifying these metrics the CA identifies the author of the instructional content and keeps track of this information. Metadata 1 gives an example of a metadata file that was generated by the Content Analyser (CA) and in particular shows the author contact information.

Metadata 1 Contact information produced by the Content Analyser

```
-<SCOMetadata>
  -<GeneralInfo>
    <Author>Keith_Maycock</Author>
    <Contact>kmaycock@ncirl.ie</Contact>
   </GeneralInfo>
  +<CognitiveResources></CognitiveResources>
 </SCOMetadata>
```

The personal profile that was identified to be appropriate for an online learning environment consists of Working Memory Capacity (WMC), Readability, Information Processing Speed (Information Processing Speed) and the Pedagogic preference, as discussed in detail in Chapter three. Metadata 2 gives an example of the measurements describing the cognitive metrics found within instructional content.

The IPS indicator is used as an estimation of the working memory of an individual. The cognitive metrics found within instructional content that stimulate a learners personal profile are: the amount of content, the readability of the instructional material and the VARK representation of the content. These metrics are described below:

Metadata 2 Illustrating the cognitive metrics found by the Content Analyser


```
-<SCOMetadata>
  +<GeneralInfo></GeneralInfo>
  -<CognitiveResources>
    <AvailableScreen>92.5</AvailableScreen>
    <VisualTolkens>13</VisualTolkens>
    +<images></images>
    -<Readability>
      <FleschReadingEase>39.83</FleschReadingEase>
      <FleschKincaidGrade>12</FleschKincaidGrade>
    </Readability>
      <amount>127</amount>
      <VARK>16.97</VARK>
    +<links></links>
  </CognitiveResources>
</SCOMetadata>
```


- *amount:* the amount is an indicator of the volume of words found within the instructional content.
  - This metric is used to calculate an approximation towards the WMC

of a learner. Multiple file formats are catered for using the Java Open Document (JOD) libraries to interface with Open Office. A regular expression is defined to describe a suitable *word* and then a simple calculation is performed.

– The working memory of an individual has been extensively researched as described earlier. Three models of working memory that have emerged from this area are: Baddelys model, Cowans Model and the theory of Ericsson and Kintsch. Unfortunately all three models have their differences and different interpretations of a capacity associated with the WMC of a learner. The concept of a *chunk* of information is discussed without referring to a specific definition of a chunk, especially in a general term. Within online learning the problem is further increased as the exercise is not to remember several digits but is related to text comprehension, which requires all of the following to take place: perceptual features, linguistic features, propositional structure, macrostructure, situation model, control structure, goals, lexical knowledge, frames, general knowledge and episodic memory for prior text [9]. All of these components taken separately would exceed any limitation of working memory, however Kintsch et∼al [11] believes that every reader is able to form episodic text structures during text comprehension. Furthermore, if a single sentence is considered, constructed using suitable visual stimulus (suited to a learners pedagogic preference) and containing a level of readability approximating the learners readability level this establishes the foundation of understanding a *chunk* within an online learning environment. Additionally if the granularity of the learning content is described as previously stated at the concept level, this will further enhance the working memory of the learner as a single concept should contain information relating to the concept and not contain too many external interruptions diverging from the overall meaning of the instructional content.

- *FleschReadingEase:* is used as an indicator of the readability level of the learner. All readability formulas are limited, especially when applied to specific learners and settings. The readability level is used as a metric for the adaption process to enhance the WMC metric. It should be noted that the readability just like the other identified metrics could be removed from the adaptive process and other traits be included.

  – The metric is calculated as follows:

  $$206.835 - ((avgSyllables * 84.5) + (avgWords * 1.015))$$

  where,

    * *avgSyllables:* is the average number of syllables contained in each word. A syllable is defined by the International Phonetic Alphabet as one of the following: *ea, i, e, a, o, aw, a, oo, u, ir, a's, es, ee, ar, er, ay, o, y, ough, oy, oor, air, our, ear, ere.*
    * *avgWords:* is the average number of words contained in each sentence.

- *VARK:* This method takes as input an absolute file name and returns a double value indicating the percentage of the screen that is composed of visual elements.

  – These visual elements are identifiers for the visual resources as described by Neil Flemming describing the VARK learning preference [1].

  * The following elements are used for identifying visual identifiers: *"b", "i", "tt", "sub", "sup", "big", "small", "hr", "strong", "em".*
  * The image / objects are defined by: *"IMG or img", "AREA or area", "map or MAP", "object or OBJECT", "param or PARAM".*

- The value of the VARK representation is calculated as follows:

$$VARK = \left(\frac{totalVisual}{words}\right) * availableScreen$$

where,

$$avaliableScreen = 100 - \left(\left(\frac{pixel}{screensize}\right) * (100)\right)$$

and,

totalVisual = the total number of visual constructs as defined above
words = total number of words found within the instructional content as defined above
pixel = total screen covered by the image or object constructs as defined above

The following subsections detail the metadata associated with two components within instructional material: images and external links. In particular, the following subsections are concerned with the potential interruption that can occur due to changing the structure of the instructional material as discussed by Laurillard [13].

4.2.1 The importance of structure

Laurillard discussed the problems associated with decoupling instructional material and modifying the possible meaning of instructional content [13], as discussed in Chapter three. However when the granularity of the learning material is at a conceptual level and there exists enough learning resources, it should be possible to insert or remove images (with associated textual information) without destroying the overall meaning of the instructional content.

Ensuring that no meaning is lost in the addition or removal of an image, all associated references and text associated with the image must also but added or removed. The metadata in Metadata 3 allows an automated process to automatically insert or remove images and provides all the metadata required to update the cognitive metrics found within the instructional content. It can be clearly seen in Metadata 3 that an image has an associated name, dimensions, word count and visual tokens. These metrics are used to calculate the impact that the image will have on the evaluation of instructional content against the personal profile of a given learner. Chapter five details the process for evaluation of instructional content in more detail.

4.2.2 Controlling the instructional space

Metadata 4 shows metadata describing instructional content which contains two links. The first link contains zero images but contains information relating to all the cognitive metrics as described in Chapter three. In chapter four section 1.1, the process for using external links to support the explanation of a concept in greater detail or to further strengthen a concept was discussed. Unlike images external links can simply be treated as another concept file without any embedding issues. It can be clearly seen that a link contains all the required information associated with the cognitive metrics found within the instructional material and can also contain additional links. Chapter five details strategies for estimating the potential effect that the size of the instructional space can have on a learner interacting with the learning component.

Metadata 3 Metadata produced by the CA associated with an image


```
-<SCOMetadata>
  +<GeneralInfo></GeneralInfo>
  -<CognitiveResources>
    <AvailableScreen>92.5</AvailableScreen>
    <VisualTolkens>13</VisualTolkens>
   -<images>
      <NoOfImages>1</NoOfImages>
     -<image>
        <imgtitle>usecase</imgtitle>
        <imgDimensions>200:300</imgDimensions>
       -<imgText>
          <imgWords>76</imgWords>
          <imgVT>13</imgVT>
        </imgText>
      </image>
    </images>
   +<Readability></Readability>
    <amount>127</amount>
    <VARK>16.97</VARK>
   +<links></links>
  </CognitiveResources>
</SCOMetadata>
```

External links create a complex and expansive instructional space, however, if all the instructional content on the internet was filtered through the Content Analyser (CA) there would exist a huge pool of resources within the learning object repositories. The problem now changes from generating instructional metadata suited to the cognitive ability and pedagogic preference to creating efficient algorithms to reconstruct the learning objects in a suitable fashion to ensure no loss of meaning from the instructional space.

4.2.3 Summary

This section identified suitable metrics associated with the personal profile, described in Chapter three, found within instructional content. The section was also focused on components within instructional content that could be used to modify instructional content without effecting the meaning of the desired instructional content. In addition this section discussed the complexity issues introduced if the CA was used to migrate huge volumes of data consequently changing the problem of creating suitable metadata that reflects the cognitive ability and pedagogical preference of a learner to an evolutionary problem.

4.3 Conclusion

In conclusion, the Content Analyser was designed and constructed to bridge the perceived gap between the inconsistencies found with instructional content within content repositories and the lack of consistency found with metadata creation. Consequently, this creates an environment whereby traditional Adaptive Hypermedia Systems (AHS) cannot be used in the real world as their closed loop approach is too restrictive, however if a closed loop approach was not used AHS would still

Metadata 4 Metadata produced by the CA associated with a Link


```
-<SCOMetadata>
  +<GeneralInfo></GeneralInfo>
  -<CognitiveResources>
    <AvailableScreen>92.5</AvailableScreen>
    <VisualTolkens>13</VisualTolkens>
   +<images></images>
   +<Readability></Readability>
    <amount>127</amount>
    <VARK>16.97</VARK>
   -<links>
      <NoOfLinks>2</NoOfLinks>
     -<link>
       -<LinkCognitiveResources>
          <Linkname>link_name</Linkname>
          <LinkAvailableScreen>100.0</LinkAvailableScreen>
          <LinkVisualTolkens>0</LinkVisualTolkens>
         -<LinkImages>
            <NoOfImages>0</NoOfImages>
          </LinkImages>
         -<LinkReadability>
            <LinkFleschReadingEase>0.0</LinkFleschReadingEase>
            <LinkFleschKincaidGrade>12.0</LinkFleschKincaidGrade>
          </LinkReadability>
```

Metadata 4 Metadata produced by the CA associated with a Link

```
            <Linkamount>121.0</Linkamount>
          <LinkVARK>0.0</LinkVARK>
          <LinkNoOfLinks>0</LinkNoOfLinks>
        </LinkCognitive Resources>
      </link>
     +<link></link>
    </links>
  </CognitiveResources>
</SCOMetadata>
```

not be ready for wide spread adoption as the information available is inconsistent (multiple referencing standards) with insufficient metadata. In chapter three, a unique personal profile was described that included the cognitive traits and pedagogical preference of a learner, which had associated cognitive metrics within instructional content designed for an online learning environment. Chapter four detailed the process of reading in multiple file formats and reducing the content to a simple format using the JOD library and creating suitable metadata for the content. Once the metadata is created the content is repackaged as SCORM compliant content. Additionally the chapter discussed complexity issues associated with the content analyser harvesting too much information. The problem now changes from generating instructional metadata suited to the cognitive ability and pedagogic preference to creating efficient algorithms to reconstruct the learning objects in real time. The following chapter investigates various evolutionary algorithms, in order to traverse a potentially unsearchable space to construct a course adapted to the individual needs of each learner. Additionally, an analysis is performed on the metrics of such an algorithm to ensure that the algorithm is an optimal solution.

Chapter 5

Selection Model

The Content Analyser, discussed in detail in Chapter four, enabled the automatic generation of metadata to suit the cognitive resources found within instructional content. This analyser bridged the perceived gap between the inconsistency of content found within instructional repositories and also the inconsistency with metadata generation for SCORM content. This analyser is a critical component to the research question of the thesis;

Is it possible to construct an automated learning component that generates instructional content suited to the cognitive ability and pedagogical preference of a learner?

Using the CA repositories of learning objects can be generated with the appropriate metadata associated with suitable cognitive resources as discussed in Chapter three. This Chapter is involved with a Selection model that is used to harvest the instructional material within generated repositories. The Selection model is the nucleus of the learning component, it identifies and reengineers instructional content, using a genetic algorithm to produces mathemagenic content suited to the individual needs of each learner. This chapter firstly investigates the use of evolutionary algorithms as an appropriate method of evolving instructional content. Secondly, a high level protocol is discussed for interacting with the learning component. Additionally, the various metrics governing the genetic operators of the GA are investigated to ensure an optimal evolutionary strategy. Identifying suitable metrics (rate of Mutation, Selection operator, type of CrossOver method) for the genetic operators is a complex process especially when there exists an incomplete solution space. This strategy is achieved by creating a suitable comparable problem that has a complete instructional space. The genetic operators are then examined using this pseudo problem. Finally, the chapter concludes with a discussion investigating the performance of the algorithm to find suitable instructional content.

5.1 Suitable searching strategies

The core function of the Selection model is to search an instructional repository and take chunks of instructional material suited to the individual needs of a learner, until the final course that is delivered scores a fitness value above the Minimum Expected Learning Experience (MELE), that is set by an author of the specification. To achieve this functionality the following conditions have been identified as necessary components for a searching strategy:

- An author controlled adaptive threshold metric to allow the author set the exit requirement for suitable courses. This allows multiple authors the freedom to choose the appropriate exit requirement, for example, MELE is above seventy percent.

- An author controlled adaptive metric to favor instructional content based on the cognitive resources within the content. With the expected growth in e-learning this metric will allow authors using different pedagogic strategies the freedom to control the evolutionary strategy based on strengths of individual cognitive traits.

- The functionality should allow for fast identification of suitable objects, the dissemination of the instructional content and recombination of various components while keeping track of the authors of the instructional content.
- One of the biggest problems when an automated process is mining through very large instructional spaces is the possibility of the process arriving at a local minimum (crowding problem). The functionality should consider this when constructing the evolutionary algorithm.
- The content that is produced does not need to be a perfect match to an ideal specification (a specification that has been modified to include the metadata associated with the learners personal profile).

The classification of suitable algorithms for solving such problems are known as evolutionary algorithms. Evolutionary algorithms, unlike traditional methods like linear programming scale extremely well. Additionally the evolution process is not a linear evolution, during the initial phase (early epochs) the evolutionary strategy excels exponentially and over time the evolution rate degrades. Generating course content is suited to this model as the MELE should never be set at 100%, as a learner interacting with the learning component should be given the opportunity to exceed the expectations of the learning component. By definition the MELE estimates the minimum threshold for a learning experience not the maximum. The following subsections briefly explore some evolutionary algorithms and their applications to identify a suitable candidate for evolving instructional content.

5.1.1 Ant colony optimisation

Ant Colony optimisation technique is based on the natural habits of a colony of ants searching for food. Initially the ants would move randomly searching for food. Once a successful search was returned to the colony the ant would leave a trace of pheromones showing the path from the colony to the food. Over time successful paths become probabilistically favored for subsequent travel. These types of algorithms have been used to solve various combinatorial optimisation problems, including the Traveling Salesman Problem [86]. Ant colony optimisation algorithms would be suitable for the initial identification of suitable learning objects, however they would not be suitable for the dissemination and reengineering of the content while keeping track of the individual authors of the instructional content.

5.1.2 Cultural algorithm

Cultural Algorithms are an extension of genetic algorithms which include extra information regarding the *Belief Space* [87]. The knowledge held by the population about the *Belief Space* is classified into several categories: Normative knowledge, domain specific knowledge, situational knowledge, temporal knowledge and spatial knowledge. After each epoch of the evolution strategy the *Belief Space* is updated. Cultural algorithms have been successfully applied to solve the Royal Road problem [88] as suggested by Holland [89].

The Content Analyser (discussed in Chapter four) details the construction of an automated component to migrate content into a suitable format for the Selection model. This allows for an easy translation for instructional content thus resulting in an immeasurable amount of learning objects. Using Cultural algorithms as a suitable approach would be an ideal solution, as additional information regarding the domain knowledge could easily tracked. However, the experiments implemented are in a controlled environment and do not require any additional information relating to the domain knowledge thus negating the requirement for Cultural Algorithms as the extra overhead associated with interfacing between the population and the belief space would be considered insufficient.

5.1.3 Extremal optimisation

Extremal Optimisation (EO) algorithms were initially designed as local search algorithms for combinational problem spaces, but include mutation strategies to shift the search optimising strategy to focus on another segment of the instructional space [90]. Self Organised Criticality (SOC) is an optimisation heuristic based on a single attractive critical point throughout the evolutionary process. The strategy is based on the evolution of a single solution unlike genetic algorithms were there exists a population of solutions. The main drawback of using EO or SOC as an effective algorithm for generating instructional content would be the that EO does not support the dissemination and reengineering of instructional content during each epoch of the evolutionary process.

5.1.4 Reactive Search Optimisation

Reactive Search Optimisation (RSO) is the common name for a family of local search algorithms. RSO algorithms, unlike most typical evolutionary algorithms do not require the initial stage of fine tuning metrics associated with the search strategy, for example, using a genetic algorithm a researcher would initially need to run experiments to estimate the appropriate rate of mutation, the appropriate selection operator and crossover strategy suitable for each problem [91]. RSO achieves this unique fine tuning throughout the search by constantly reflecting on past experiences when navigating through the solution space. RSO was not deemed suitable for evolving instructional content as the overhead associated with fine tuning throughout the evolution process would be considered inefficient as the learning component needs to be an on-demand application that can be executed any number of times.

5.1.5 Simulated annealing

Simulated Annealing (SA) is a generic probabilistic method used for locating a good approximation to the global minimum of a solution space. The inspiration for SA comes from the annealing in metallurgy, whereby a material is initially heated and then proceeds to a controlled cooling to reduce the inconsistencies found within the material. SA works by works by evolving towards to global minimum. The probability of selecting a less fit neighbour is reduced as the initial time (T) approaches zero. Initially SA allows this migration to a less fit solution to avoid arriving at a local minimum. SA was deemed not to be suitable for evolving instructional content as the time bound associated with the SA limits the evolutionary process especially where the solution space is large. Once the time degrades the search strategy becomes a greedy search. SA has been successfully applied to many problems, example solving the *Traveling Salesman Problem* [92].

5.1.6 Genetic Algorithms

Genetic Algorithms are a search optimisation technique based on natural evolution. Initially a population of candidate solutions are randomly generated from the solution space. These candidate solutions are then evaluated using some fitness criteria and then genetic operators such as, Crossover, Mutation and Selection occur on the population each epoch until some predefined threshold is met. This process seems to be appropriate for evolving instructional content as the threshold can be controlled by the authors and there is not any extra influence on the evolutionary process like time. However, there exists a few problems:

- The instructional space for learning objects is potentially vary large and incomplete as there exists an unlimited amount of variables for the constructs within the learning objects.
- Genetic Algorithms have a tendency to approach a local minimum [93].

Firstly, when estimating the metrics for the genetic operators it is essential to have a complete instructional space. When there exists a complete instructional space it is simple a repetitive process of trying different metrics for the genetic operators and running the evolutionary process. With an incomplete instructional space, a comparable problem (a similar problem in terms of the structure and genetic operator constructs) was created that had a complete instructional space. Once these metrics are found for the comparable problem the same metrics can be used for evolving instructional content. Secondly to avoid the problem of crowding (arriving at a local minimum) multiple demes (populations) are created across the solution space and run in parallel communicating after each epoch.

5.1.7 Summary

This section investigated evolutionary algorithms to solve the problem introduced by the Content Analyzer (CA) as discussed in Chapter four, if the CA was used to migrate huge volumes of instructional content. Initially the requirements for the evolutionary algorithm were identified and discussed as the foundation requirements of the evolutionary process. Genetic Algorithms were identified as a suitable evolutionary strategy to tackle the problem of evolving instructional content to suit the personal profile of a learner. Additionally, this section also discussed additional problems associated with Genetic Algorithms, for example an incomplete instructional space and avoiding the algorithm arriving at a local minimum. The following section discusses a suitable respreentation of the elements required for representing a specification of a course being used by the Selection Model. Additionally a high level protocol is discussed that details the flow of communication for a learner interacting with the learning component using the Selection model to drive the creation of a suitable instructional course.

5.2 Selection model to automatically generate content

Given a learner profile consisting of the cognitive ability and pedagogical preference of the learner, it should be possible to construct a course to suit the cognitive ability and pedagogical preference of the learner. The selection model of the learning component uses a genetic algorithm to automatically evolve instructional content to suit a learners personal profile and is based on specifications. A specification contains a list of concepts in an unconnected hierarchical structure. Each specification contains a number of SCORM Learning Object Metadata (LOM) elements describing the content (as seen in Table 5.1); however there is no instructional information stored within the metadata files.

The metadata elements used to describe the specification are typical SCORM metadata elements as defined by the SCORM Run Time Environment (SCORM RTE). Additionally there exists metadata requirements associated with a specification to control the evolution process for optimal content generation. The Minimum Expected Learning Experience (MELE) is set by an author when constructing the specification. The MELE is an approximation of the learners capacity for a successful learning experience measured as a percentage. The MELE is used by the GA as a threshold for the fitness function. On each epoch of the evolution process

Full name	The full name of the specification is displayed at the top of the screen and in the specification listings.
Short name	The specification short name is used as a reference to a specification, specifically when sending emails.
ID number	The ID number of a specification is only used when matching this specification against other similar specifications, consequently the ID number forms part of a specifications signature.
Summary	The summary for the specification is stored in the database and retrieved only when a learner chooses a specification
MELET	MELET is measured as an approximation towards a learners' expected minimum learning experience prior to conducting a learning experience. MELET represents a percentage for the expected outcome.
Cognitive Traits	Cognitive Traits is used to allow an author to set the weights for different cognitive traits. This will ensure that each course generated for each leaner will be more focused on a particular cognitive trait.
Duration	this represents the duration of the course
Enrollment Key	this setting is set as an extra security feature for access to a specification
Force Language	this setting is used for creating content in a specified language

Table 5.1: SCORM Metadata elements used to define a Specification.

the current generation of content is evaluated and measured against the MELE. If the content is not suitable (i.e. the fitness value of the most optimal course generated is less than the MELE), the evolutionary process continues. The genetic algorithm uses the MELE as a threshold for the evolutionary process. The author also sets the Cognitive Traits field, indicating which cognitive trait is of greater importance. For example, if the author needs to generate instructional content that is focused on the working memory of the learner then the author would select the appropriate CT value.

Table 5.2 describes all the metadata associated with a concept defined within a specification. All the elements are defined by the SCORM RTE with the exception of the *Typical learning time*. The element is estimated for each individual throughout the evolutionary process to ensure that a suitable course is constructed for a given time period. The following sub section describes briefly a high level protocol for the learning component and an individual learner interacting with the system.

Title	The title element describes the title of a concept.
Description	The description element describes the target SCORM Content Model Component.
Keyword	The keyword element is used to add specific words or phrases to ensure the reusability of the learning content.
Coverage	The coverage element is used to describe time, culture, geography or region to which the SCORM Content Model Component applies..
Structure	The structure element describes the underlying structure of the SCORM Content Model Component.
Aggregation level	Defines the aggregation constraints on the material.
Size	The size element represents the size of the digital SCORM Content Model Component in bytes.
Interactivity type	Represents the dominant mode of learning.
Learning resource type	Represents the specific kind of SCORM Content Model Component.
Interactivity level	The interactivityLevel represents the degree of interactivity characterizing the SCORM Content Model Component.
Semantic density	Represents the degree of conciseness of the SCORM Content Model Component.
Context	Represents the principal environment within which the learning should take place.
Typical learning time	Represents an approximation of the typical time it takes to work through the SCORM Content Model Component.

Table 5.2: SCORM metadata elements used to describe Sharable Content Objects (SCO) within a specification.

5.2.1 High level protocol for learning component

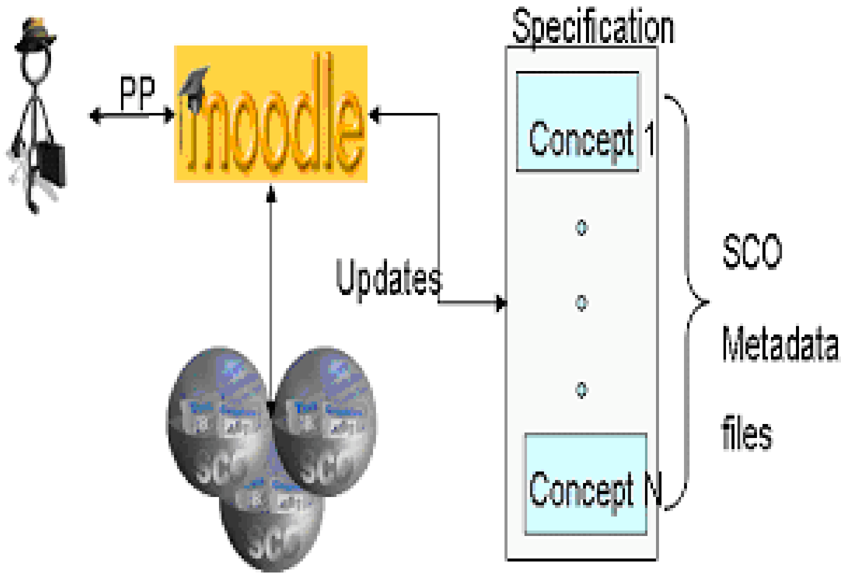


Figure 5.1: A Learner interacting with the Learning Component

Figure 5.1 shows a high level description of a learner interacting with the learning component. A learner logs into the LMS. If the learners personal profile is not known then the learner must complete four short online tests: VARK test, N-Back test, readability test and an information processing test. Chapter 6 details the design of the four tests. However, if the personal profile of the learner is known, the learner can choose a previously defined course or select a specification. If the learner selects a specification the LMS retrieves the learners personal profile and updates the specification to create a unique specification suited to the cognitive ability and pedagogical preference of the learner. This specification is also called an ideal specification as discussed earlier. The LMS uses a genetic algorithm to find optimal learning objects as defined by the LOM files contained within the specification. Once suitable learning objects are defined the course is delivered to the learner.

5.2.2 Summary

This section defined a specification that could be used with the Selection model and the corresponding SCORM data model elements that are included within the specification. In addition a high-level description of the protocol for the Selection model was discussed detailing the interaction of a learner with the learning component. The following section discusses Genetic Algorithms and the associated genetic operators in detail.

5.3 Genetic Algorithms

Genetic Algorithms (GA) [93] are search optimising algorithms based loosely on natural evolution. Initially a sample population of hypothesis are generated from the solution space. These hypotheses depend greatly on the problem being solved. Members of the initial population give rise to the members of subsequent populations by performing genetic operations such as, Selection, Crossover and Mutation. GAs have been successfully applied to a variety of learning tasks and optimisation problems, for example Grefenstette [94] has successfully applied GAs that learn sets of rules for robot control.

5.3.1 Genetic Algorithms explored

Genetic Algorithms address the problem of searching a solution space of hypotheses candidates to identify a predefined best hypothesis. This best hypothesis is found by calculating the fitness value for each individual in a population on each epoch and is returned once a pre-determined fitness value is reached. The solution space available for the candidate population is a number of metadata files describing the contents of instructional content. These metadata files consist of information relating to the cognitive ability and pedagogic preference of the ideal specification of a learner. A typical genetic algorithm is described in Table 5.3. The inputs for the algorithm are the fitness function, a threshold for the evolutionary process, the number of the individuals to be included in the population, the rate of mutation and the proportion of the population to be involved in crossover. It should be noted that the main loop within the above algorithm produces a new population after each epoch. Producing the new population requires the use of three different genetic operators: Selection, CorssOver and Mutation.

Selection occurs on the population with various strategies. Typically the population replicates selecting individuals for the new population according to some probability function. The selection operator that is described in the Table 5.3 is called roulette wheel selection, whereby an individual is selected depending on the ratio of its fitness value towards the other individuals in the population. Various methods of using fitness to select hypothesis from the population have been proposed. For example, Tournament selection randomly selects two individuals from the population a number of times (typically the number of individuals in the population) and selects the individual for progression based on a random function dependant on the fitness values for the two individuals. Tournament often yields a more diverse population than roulette wheel [95]. In another method called Rank and Truncation the population is simply ranked according to the fitness values of the individuals within the population. The next generation of individuals in the population is simply the best half of the previously ranked population doubled.

Genetic Algorithm(Fitness Function, Fitness Threshold, p, r, m)

Fitness Function: assigns a fitness value to an individual from the population.

Fitness Threshold: This specifies the termination criteria for the evolutionary process.

p: the number of hypothesis to be included in the population.

r: the fraction of the population to be replaced by CrossOver after each generation.

m: the rate of mutation.

•Initialise the Population (P), creating p candidate hypothesis.

•Using a Fitness Function evaluate each hypothesis (h) within P

•While MaxFitness(h) $<$Fitness Threshold

    1: Using a Selection strategy select candidate hypothesis to proceed to the next generation.

    2: Select (r*p/2) pairs of hypothesis from the population.

    For each pair of hypothesis (h1, h2), produce two new hypothesis by applying the CrossOver strategy. Add the new offspring to the new Population.

    3: Perform Mutation on the Population. Mutation selects m candidate hypothesis from the population with a uniform probability and implements a mutation on each of the selected candidates

    4: Perform an evaluation on the new population that has been created, compute fitness (h).

•Return the hypothesis from the population that yields the highest fitness.

Table 5.3: Typical Genetic Algorithm.

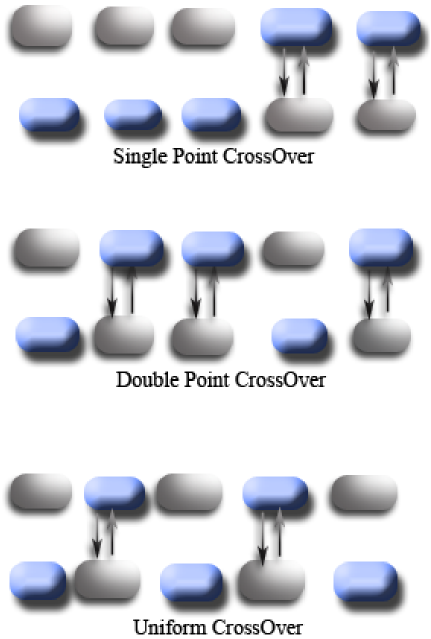


Figure 5.2: CrossOver strategies used with genetic algorithms

CrossOver is implemented on a proportion of the population. CrossOver swaps portions of individuals to form new individuals for the next epoch. Figure 5.2 shows three different CrossOver strategies: Single-point crossover, Two-point crossover and Uniform crossover. Single-point crossover selects at random a position along the crossover mask and randomly selects two individuals from the population. Once the individuals are selected the tails of the individuals are swapped to form two new individuals. Two-point crossover selects two random positions along the crossover mask and randomly selects two individuals from the population. Portions of these two individuals are swapped according to the positions from the crossover mask to form two new individuals. Uniform crossover selects a random amount of crossover points along the crossover mask and selects at random two individuals from the population. Two new individuals are formed by swapping elements between the two individuals according to the crossover points along the crossover mask. Mutation occurs on a percentage of the population. The rate of mutation varies depending on the problem being solved. Mutation randomly selects a number of individuals from the population on each epoch. Once an individual is selected a portion of the individual is selected to mutate. The mutate operator is dependant on the problem being solved. Mutation is very useful in the evolutionary process to avoid the GA evolving to a local maximum and simply takes in content from the solution space that was not in the original population. The fitness function defines the criteria for ranking the potential hypotheses from the population. If the problem was to control traffic flow then the fitness function would be an estimation of the throughput of all the junctions for a given setup of traffic lights.

5.3.2 Summary

This section describes the general functionality of a Genetic Algorithm (GA). In particular, the genetic operators of a GA were described in detail. The following section describes the problem for evolving instructional content. The inconsistencies of the instructional space is discussed, and consequently the associated problems involved with training the GA (identifying suitable metrics for the genetic operators). The following section also describes a suitable comparable problem that is used as the framework for identifying suitable metrics for the GA, for evolving instructional content. Additionally, the performance of the GA for evolving instructional content is analysed and discussed.

5.4 Using a GA for course construction

Building a genetic algorithm that evolves course content suited to the cognitive ability and pedagogical preference of a learner requires the identification of suitable metrics found within instructional content as discussed in Chapter four. Unfortunately, the complete instructional space is infeasible to create as there is no upper bound on the content that stimulates the learners personal profile. Without having the complete instructional space finding suitable metrics becomes an issue. The approach that was followed to identify suitable metrics was to describe a suitable comparable problem with a complete instructional space and train the GA over this problem. The following subsections detail the identification of suitable metrics associated with the comparable problem and subsequently detail the strategies associated with the genetic operators for evolving instructional content.

5.4.1 Comparable problem with complete solution space

A genetic algorithm to determine the largest common sub graph between two isomorphic graphs was developed as the framework for our genetic algorithm to identify the correct domain knowledge elements suited to the cognitive ability and pedagogic preference of a learner. Both problems are identical as all SCORM learning objects contain an activity tree consisting of the structure and navigational flow of the learning content. The graphical representation of a learning object would consist of Sharable Content Objects connected in a hierarchical structure, however the cognitive resources within the SCOs would represent arcs joining the SCOs. In designing the genetic algorithm to find the isomorphic relevance between two graphs, experiments were conducted to determine the rate of mutation, the correct selection operator and the effectiveness of gene repair to maximize the structure matching technique. The following sub sections detail the experiments conducted to estimate the relevant metrics associated with the genetic operators.

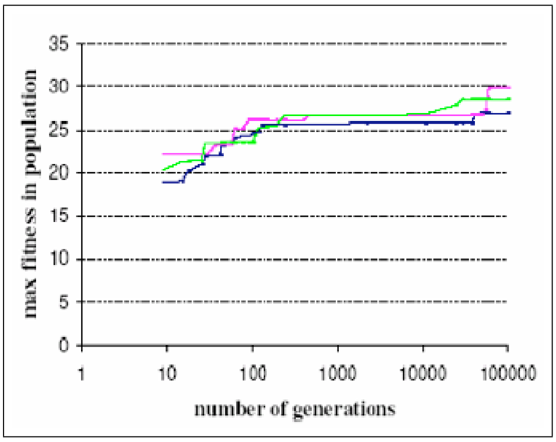
5.4.1.1 Mutation rate

Twenty five experiments were conducted to estimate the correct mutation rate. All experiments were carried out on isomorphic graphs with twenty lines. The graphs were randomly generated using a domain range of zero to twenty, ensuring that the graphs were highly connected. The initial population consisted of one hundred individuals. Single point mutation was carried out on each of the randomly selected individuals from the population. Rank and Truncation selection was implemented. To ensure that all the lines are being matched up a genetic operator called gene repair was implemented. For each set of isomorphic graphs ten different rates of mutation were tested in steps of two, from zero to twenty. A mutation rate of eight percent was found to be most optimal for the structure matching technique.

5.4.1.2 Effectiveness of Gene Repair

One hundred experiments were carried out on isomorphic graphs with twenty lines to determine the effectiveness of gene repair. The graphs were randomly generated using a domain range of zero to twenty, ensuring that the graphs were highly connected. The initial population consisted of one hundred individuals. Single point CrossOver was conducted on each of the randomly selected individuals with a Mutation rate of eight percent using Rank and Truncation selection. All experiments were stopped after one thousand generations when gene repair was incorporated into the evolution strategy. It was found that, on average with gene repair implemented the GA would arrive at a fitness value of ninety percent.

It can be seen from Figure 5.3 that all three iterations when gene repair was incorporated reached one hundred percent mapping (scoring a fitness value of 40) in less than one thousand epochs. In contrast, without gene repair implemented the maximum fitness reached after ten thousand epochs was a seventy five percent



(a) (b)

Figure 5.3: Investigating the performance of gene repair on isomorphic graphs: **(a)** three best implementations with gene repair incorporated, **(b)** three best implementations without gene repair incorporated.

mapping. This is a significant reduction in the computational time required for identifying isomorphisms between the graphs.

5.4.1.3 Identifying a Selection Operator

Ten experiments were carried out to determine the most optimal selection operator for the LCS problem. All experiments were carried out on isomorphic graphs with twenty lines, and each of the generated graphs were tested using Rank and Truncation selection, Roulette Wheel selection and Tournament selection. The graphs were randomly generated using a domain range of zero to twenty, ensuring that the graphs were highly connected. The initial population consisted of one hundred individuals. Single point crossover was carried out on each of the randomly selected individuals with a mutation rate fixed at eight percent. Gene Repair was implemented in all experiments. It can be clearly seen in Table 5.4 that all three selection operators performed well for finding the LCS between the isomorphic

Exp Max Fitness	Rank and Truncation	Roulette Wheel	Tournament
1	95%	100%	100%
2	100%	90%	82.5%
3	100%	80%	100%
4	100%	82.5%	100%
5	100%	87.5%	82.5%
6	100%	90%	77.5%
7	100%	82.5%	100%
8	100%	90%	82.5%
9	100%	87.5%	100%
10	85%	90%	82.5%

Table 5.4: Selection Operator Performance.

graphs. Rank and Truncation selection was selected as the most suitable selection operator with an average success rate of ninety eight percent.

5.4.2 Genetic Operators for evolving content

Genetic operators are described as the components of a GA that perform bit operations to aid in the evolution of some problem until a predefined threshold is arrived at. The bit operations vary depending on the problem. The GA for evolving instructional content will use a Mutation rate of eight percent, Single point CrossOver, Rank and Truncation and will incorporate Gene Repair. However instead of allowing duplications to arrive into the population before Gene Repair is implement all genetic operators will explicitly avoid duplications.

One of the criteria for the search algorithm described at the start of Chapter five is that a track record of the author of the instructional content should be kept throughout the evolutionary process. Metadata 1 in Chapter four gives a sample metadata file that was generated. It can be clearly seen under the segment *GeneralInfo* that both the authors contact information and name are represented. During a genetic operation this information is passed along with the genetic modification. Thus ensuring when a course is constructed all authors could be potentially rewarded as per unit of instruction. The follow sub sections detail the individual constructs and strategy for Mutation, CrossOver and the Fitness Function used in the GA for evolving instructional content.

5.4.2.1 Mutation Operator for evolving content

The rate of Mutation that has been selected for the GA to evolve instructional content is eight percent as previously discussed above. This means that on each epoch of the evolutionary process eight percent of the population is going to be mutated. Mutation performs an extremely import function as it acts as the only method to avoid the evolutionary process arriving at a local minimum. To calculate a percentage of the population depends on the representation of the problem being solved. For example, in our GA the population consists of Individuals, each individual represents a candidate course consisting of a number of Sharable Content Objects. Consequently, the granularity of a single object subjected to possibly mutation is defined as an individual SCO, therefore eight percent of the total number of SCOs within the population are mutated at each epoch of the evolution process.

Three different types of Mutation can occur when a Mutation is implemented:

- Complete Concept Mutation: this is where a concept is randomly selected from the population and removed from the population. The Mutation function then selects a suitable random Sharable Content Object from the instructional space. This newly selected learning object replaces the removed object.

- Links Mutation: this mutation is focused on the extra information supporting the learning object. A learning object is randomly selected from the population. Throughout a links mutation either the complete set of links associated with a learning object are removed or replaced, or a single link is randomly deleted or inserted. Metadata 4 in Chapter four shows an example of the metadata associated with a link as created by the Content Analyser.

- Image Mutation: A learning object is randomly selected from the population. The Mutation function then randomly removes or inserts an Image. Metadata 3 in Chapter four gives an example of the metadata associated with an image as produced by the Content Analyser. After an image mutation has occurred all the relevant fields within the SCO metadata are updated to reflect the newly modified learning object.

5.4.2.2 CrossOver Strategy used for evolving content

The CrossOver strategy that has been selected to be the most appropriate strategy for evolving instructional content is *single point crossover*. It was decided that *single point crossover* would be most suitable as there exists a limited number of possible CrossOver points within the metadata describing the instructional content, to ensure that the completed course produced consists of complete learning objects. The CrossOver strategy performs n / 2 crossovers on each epoch of the evolutionary process, where n represents the number of individuals within the population. The possible points where crossover can occur are at: concept level, links level.

- Concept level: When CrossOver occurs at the concept level two individuals are randomly chosen from the population (candidate courses) and *single point crossover* occurs as described earlier to produce two new candidate courses.

- Links Level: When CrossOver occurs at the links level two individuals are randomly chosen from the population. A crossover point is then chosen from within the individuals and then *single point crossover* occurs as described earlier to produce two new candidate courses.


```
- <SCOMetadata>
  - <GeneralInfo>
      <Author>Keith14</Author>
      <Contact>contact</Contact>
    </GeneralInfo>
  - <CognitiveResources>
      <AvailableScreen>10</AvailableScreen>
      <VisualTolkens>110</VisualTolkens>
    + <images></images>
    + <Readability></Readability>
      <amount>250</amount>
      <VARK>94</VARK>
    - <links>
        <NoOflinks>1</NoOflinks>
      - <link>
        - <LinkCognitiveResources>
            <Linkname>linkA</Linkname>
            <LinkAvailableScreen>30</LinkAvailableScreen>
            <LinkVisualTolkens>10</LinkVisualTolkens>
          + <LinkImages></LinkImages>
          + <LinkReadability></LinkReadability>
            <Linkamount>250</Linkamount>
            <LinkVARK>71</LinkVARK>
          </LinkCognitiveResources>
        </link>
      </links>
    </CognitiveResources>
  </SCOMetadata>
```

```
- <SCOMetadata>
  - <GeneralInfo>
      <Author>Keith28</Author>
      <Contact>contact</Contact>
    </GeneralInfo>
  - <CognitiveResources>
      <AvailableScreen>10</AvailableScreen>
      <VisualTolkens>20</VisualTolkens>
    + <images></images>
    + <Readability></Readability>
      <amount>250</amount>
      <VARK>90</VARK>
    - <links>
        <NoOflinks>1</NoOflinks>
      - <link>
        - <LinkCognitiveResources>
            <Linkname>linkA</Linkname>
            <LinkAvailableScreen>60</LinkAvailableScreen>
            <LinkVisualTolkens>120</LinkVisualTolkens>
          + <LinkImages></LinkImages>
          + <LinkReadability></LinkReadability>
            <Linkamount>240</Linkamount>
            <LinkVARK>184</LinkVARK>
          </LinkCognitiveResources>
        </link>
      </links>
    </CognitiveResources>
  </SCOMetadata>
```


Figure 5.4: Candidate metadata files randomly selected for link crossover

Figure 5.4 gives an example of two randomly created metadata files. These files have been selected for links crossover. Figure 5.5 show the resultant new candidate courses produced after crossover has occurred on the initial two randomly chosen individuals.

```
- <SCOMetadata>
  - <GeneralInfo>
      <Author>Keith14</Author>
      <Contact>contact</Contact>
    </GeneralInfo>
  - <CognitiveResources>
      <AvailableScreen>10</AvailableScreen>
      <VisualTolkens>110</VisualTolkens>
    + <images></images>
    + <Readability></Readability>
      <amount>250</amount>
      <VARK>94</VARK>
    - <links>
        <NoOflinks>1</NoOflinks>
      - <link>
        - <LinkCognitiveResources>
            <Linkname>linkA</Linkname>
            <LinkAvailableScreen>60</LinkAvailableScreen>
            <LinkVisualTolkens>120</LinkVisualTolkens>
          + <LinkImages></LinkImages>
          + <LinkReadability></LinkReadability>
            <Linkamount>240</Linkamount>
            <LinkVARK>184</LinkVARK>
          </LinkCognitiveResources>
        </link>
      </links>
    </CognitiveResources>
  </SCOMetadata>
```

```
- <SCOMetadata>
  - <GeneralInfo>
      <Author>Keith28</Author>
      <Contact>contact</Contact>
    </GeneralInfo>
  - <CognitiveResources>
      <AvailableScreen>10</AvailableScreen>
      <VisualTolkens>20</VisualTolkens>
    + <images></images>
    + <Readability></Readability>
      <amount>250</amount>
      <VARK>90</VARK>
    - <links>
        <NoOflinks>1</NoOflinks>
      - <link>
        - <LinkCognitiveResources>
            <Linkname>linkA</Linkname>
            <LinkAvailableScreen>30</LinkAvailableScreen>
            <LinkVisualTolkens>10</LinkVisualTolkens>
          + <LinkImages></LinkImages>
          + <LinkReadability></LinkReadability>
            <Linkamount>250</Linkamount>
            <LinkVARK>71</LinkVARK>
          </LinkCognitiveResources>
        </link>
      </links>
    </CognitiveResources>
  </SCOMetadata>
```


Figure 5.5: New Candidate metadata files produce after crossover has occurred

5.4.2.3 Evaluating a Candidate course

The success of a genetic algorithm in solving any problem is centered on the effectiveness of the *fitness function* being able to calculate the fitness of each candidate solution. The *fitness function* that was developed for calculating the fitness of a candidate course generated to suit the cognitive ability and pedagogical preference of a learner focuses on the cognitive metrics found within the instructional content and the structure of the content (including all links). The algorithm is described as follows:

- let P denote the population,
- let w denote the metric representing the learners working memory as described within their personal profile,
- let v denote the metric representing the learners VARK score as described within their personal profile,
- let r denote the metric representing the learners Readability level as described within their personal profile.
- let a denote the multiplier associated with the strength of the working memory for evolving instructional content,
- let b denote the multiplier associated with the strength of the VARK score of the learner,
- let c denote the multiplier associated with the strength of the Readability of the learner,
- let wn denote the number of metadata fields associated with the working memory of a learner,

- let *vn* denote the number of metadata fields associated with the VARK score of the learner,
- let *rn* denote the number of metadata fields associated with the Readability of the learner.
- let *wi* denote the value associated with the parameter describing the cognitive metric associated with the working memory.
- let *vi* denote the value associated with the parameter describing the cognitive metric associated with the VARK,
- let *ri* denote the value associated with the parameter describing the cognitive metric associated with the Readability,
- let *num* denote the number of elements that are being assessed taken into account the strength of the multipliers.

Additionally, there exist two functions called *getActual* and *getStructure*. These methods allow the *fitness function* to calculate an overall score associated with the complete learning objects as a unified course structure.*GetActual* takes as input a double value and returns a representation of the score taken into account the complete structure of the learning object. *GetStructure* determines the complete structure of the learning object element independent of the individual elements described within the learning object (see *http://www.cs.nuim.ie/kmaycock/fitnessfunction* for more details).

The *fitness function* is described as follows:
fitness function(P, w, v, r, a, b, c, num, boolean [] needed)

For every learning object in each individual the fitness is calculated for all the elements associated with working memory, VARK or the Readability level of the learner,

The working memory elements are calculated as follows:

$$\sum_{wn=1}^{wn} \left(\frac{getActual\left(\|w - wn\| * 0.4\right) * a}{2} \right)$$

denoted by *tw*; representing the total score achieved by the working memory elements.

The readability elements are calculated as follows:

$$\sum_{rn=1}^{rn} \left(\frac{getActual\left(\|r - rn\|\right) * c}{2} \right)$$

denoted by *tr*; representing the total score achieved by the readability elements.

The VARK elements are calculated as follows:

$$\sum_{vn=1}^{vn} \left(\frac{getActual\left(\|v - vn\|\right) * b}{2} \right)$$

denoted by *tv*; representing the total score achieved by the VARK elements.

Each individual within the population consists of a number (n) of learning objects as described above. The following formula gives the fitness of an individual within the population:

$$\sum_{i=0}^{n}\left(\left(\frac{(tv+tr+tw)}{a+b+c}\right)+getStructure\left(LO_i\right)\right)$$

5.4.2.4 Mapping the Personal Profile

VARK Score	**Personal Profile Metric**
VRK VAK VAR	35
VK VA VR	50
VvStrong	100
Vmild	60
VARK	25
Anything else	0

Table 5.5: VARK score mapping to suitable elements for fitness function.

Chapter three discussed a suitable Personal Profile that could be used to generate instructional content in an online learning environment. The profile consists of: Working Memory Capacity, Information Processing Speed, VARK and the Readability. The fitness function as described above performs calculations on the elements of the personal profile and consequently a mapping is required to a suitable format. Table 5.5 represents the mapping from the results of the VARK test to a suitable format for the fitness function. It can be clearly seen that: for a single strong visual preference the learners score is 100, for a single visual mild preference the learners score is 60, for a bi-modal preference including the visual elements the learner is given a score of 50, for a tri-modal preference including the visual element the learners score is 35, for a multi-modal preference including all elements the learners score is 25, and for all other categories the learners score is 0.

N-Back Score	Category	Personal Profile Metric
2-2.5	vLow	50
2.6-2.9	Low	100
3-3.5	Medium	150
3.6-3.9	High	200
4+	vHigh	250

Table 5.6: N-Back score mappings to suitable elements for fitness function.

Table 5.6 shows the categories of results for the Working Memory Capacity associated with the learners profile. The Information Processing Speed is calculated as a percentage of accuracy for a learner interacting with instructional content in an online learning environment and is multiplied with the N-Back score. It can be clearly seen that the categories of learners results from the N-Back score map into numerical elements that can be manipulated in the fitness function. The Readability score that the learner gets is used directly in the calculations with the fitness function.

5.4.3 Avoiding a the crowding problem

The Crowding problem exists with genetic algorithms typically when the solution space is large and the initial population is constrained by the distribution of the candidate hypotheses. Mutation is used to reduce the possibility of the evolutionary process arriving at a local minimum (state of Crowding). However, if Mutation is implemented crowding can still occur. To reduce the possibility of crowding a parallel implementation should be considered. The following sub sections briefly describe the possibilities for a parallel implementation and then describe the design of the solution created to reduce the possibility of the learning component arriving at a local minimum.

5.4.3.1 Parallel possibilities for a GA

Genetic Algorithms are very suited towards a parallel processing implementation. There exist two main approaches to parallelisation: course grain parallelization and fine-grained parallelisation. Coarse grain approaches typically create multiple populations or split the population into subdivisions, called demes and have an associated processor for each deme. Cross fertilisation occurs between demes at regular intervals and each member of the population is updated to the successful sub-divisions within the population. Fine-grained implementations typically have an associated processor for every individual within the population and cross fertilisation occurs at different intervals.

The parallel implementation that was implemented to reduce the possibility of the learning component arriving at a local minimum was a coarse grain implementation. When the initial populations are being generated, each population selects candidate hypotheses from unique portions of the solution space. The populations communicate once a suitable learning object is found ensuring that a suitable asynchronous protocol is implemented.

5.4.4 GA for Optimal Learning Objects

A sample population of learning objects was generated to test the genetic algorithm. This population consisted of twenty different concepts each containing one thousand randomly generated LOM files to mimic a real world problem where the full learning space is not available. A specification was randomly selected from the population with an expected minimum learning experience of 71.2%. The specification contained eight learning objects. The genetic algorithm was run one-hundred times; the best three implementations of the algorithm are seen in Figure 5.6.

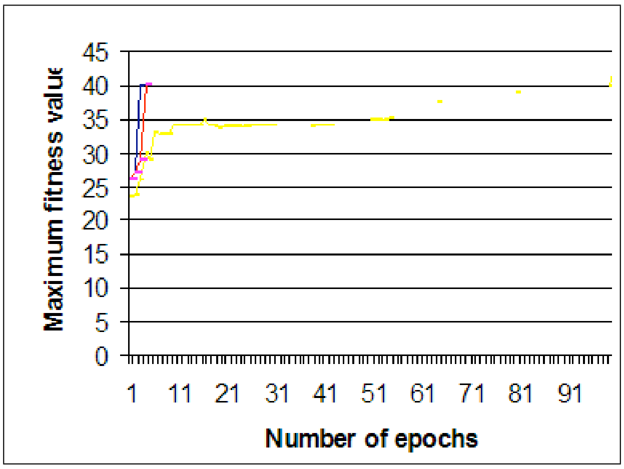


Figure 5.6: Genetic Algorithm to find optimal learning objects

On average the genetic algorithm took 43502 milliseconds to run for ten thousand epochs. The maximum obtainable fitness value that an individual in the population can attain is fifty six (for one hundred percent expected minimum learning experience.), however as the limit was set for 71.2 % the expected fitness value is forty. It can be clearly seen from Figure 5.6 that all three iterations ran to a maximum fitness value of above forty in less than one hundred epochs, taken on average 435.02 milliseconds. An analysis was carried out on the courses that were developed and it was found that in all iterations the genetic algorithm was successfully able to identify suitable learning objects within one thousand epochs for the selected time interval.

5.4.5 Summary

In summary, this section identified suitable metrics for the genetic operators associated with a GA for evolving instructional content. The section described a comparable problem that was used as the framework for training the GA, as the solution space for evolving instructional content was incomplete. The traditional genetic operators, such as, *Selection*, *CrossOver* and *Mutation* were extended to represent suitable genetic operators associated with evolving instructional content. The optimal metrics found within the training GA were applied to the GA for generating instructional content. This GA was used to evolve instructional content for a pseudo content repository, consisting of twenty thousand metadata files, describing learning objects. The GA was able to identify suitable courses within 435 milliseconds. In addition, this section addressed parallel construction possibilities for the evolutionary algorithm.

5.5 Conclusion

In conclusion, this chapter described the deign and implementation of a suitable evolutionary strategy capable of generating instructional material suited to the personal preference of a learner as described in chapter four. A Genetic Algorithm was deemed to be the most suitable evolutionary strategy for evolving instructional content, and consequently strategies for each of the genetic operations were discussed in detail. Due to the incompleteness of the instructional space a suitable comparable problem domain with a complete solution space was defined and a

GA was trained to solve the problem. These metrics acted as the foundation for the GA for evolving instructional content and were able to construct instructional courses from twenty thousand learning object metadata files within 435 milliseconds.

The following chapter is involved with the integration of the learning component into a suitable learning management system. The chapter describes the tests used to construct a learner profile. In particular, the chapter is involved with the construction of a repository of learning content and an evaluation of the consistency of the instructional authors when generating instructional content in terms of the cognitive metrics found within the generated instructional content.

Chapter 6

Learning Component Environment

This chapter investigates the suitability of a Content Management System as a shell for the learning component. This chapter is also focused on describing the personal profile tests, as described in chapter three, used to calculate the Working Memory Capacity (WMC), Information Processing Speed (IPS), the Readability level and the VARK representation for each learner / author prior to interacting with the learning component.

6.1 Moodle

The learning component was build as an evolutionary strategy for evolving instructional content but requires a CMS / LMS to import the learning component for easy use for a large population of learners. The requirements for a suitable CMS / LMS for the learning component are limited:

- the chosen LMS / CMS is required to be able to handle SCORM compliant content, i.e. the LMS / CMS must contain a SCORM RTE.

- the architecture of the chosen environment must be Open Source and designed in a modular fashion to enable integration of the learning component.

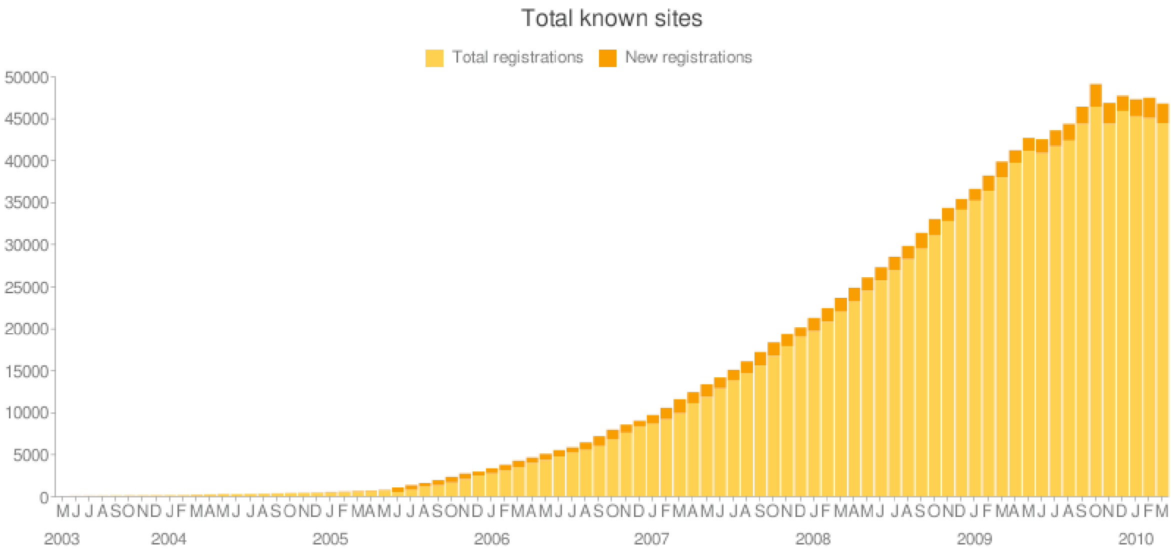


Figure 6.1: Total known Moodle cites worldwide

A number of CMS were considered, for example Moodle [40], and Sakai [41]. Moodle was chosen as the preferred CMS to import the learning component functionality, due to the rapid acceleration and adoption of Moodle throughout the higher education community. The system that imports the learning component only acts as a shell environment and could be easily incorporated into any such environment, providing the pre-conditions outlined above are taken into consideration. Moodle is an open source CMS designed around a social constructivist framework. Figure 6.1 (taken from moodle.org) shows the growth of the total known Moodle sites around the globe. Currently, Moodle is being used by 32 million users in over 205 countries and has been translated into 80 different languages [40]. Moodle was selected as a suitable CMS for the learning component due to the simple modular design and Moodles status as being an open sources CMS that complies with SCORM content. The learning component was developed as a number of applications that are all accessed through a Moodle block.

6.1.1 Summary

In summary, this section identified a suitable environment (Content Management System / Learning Management System) that would enable the learning component to interact with learners. In particular this section identified the requirements for such an environment. Firstly, the environment should support SCORM compliant content and secondly the environment should be an Open Source project and be designed in a modular fashion to allow for easy integration. Moodle [40] was chosen as the preferred CMS. The following section discusses the tests involved with generating the personal profile of a learner as discussed in chapter three.

6.2 Personal Profile

The learning component that was developed overcomes the current problems with the inconsistencies between referencing standards for metadata creation (see Chapter four), improper use of metadata creation and also the typical rhetorical methodology of lecturing in third level education [13]. In order for learners to take advantage of the learning component a number of traits must firstly be calculated to represent the working memory capacity, the information processing speed and accuracy of knowledge acquisition, the readability level of the learner and the pedagogic preference of the learner. These metrics form the basis for our proposed personal profile as discussed in Chapter three.

The following subsections detail the methods of calculating the metrics for the cognitive ability and pedagogic preference of a learner as described in the personal profile. The relationship between the metrics and the evolutionary process is also discussed.

6.2.1 Personal Profile Tests

Once the learner logs into the learning environment the learner must complete all the tests relating to the personal profile before the learner can view any available specifications.

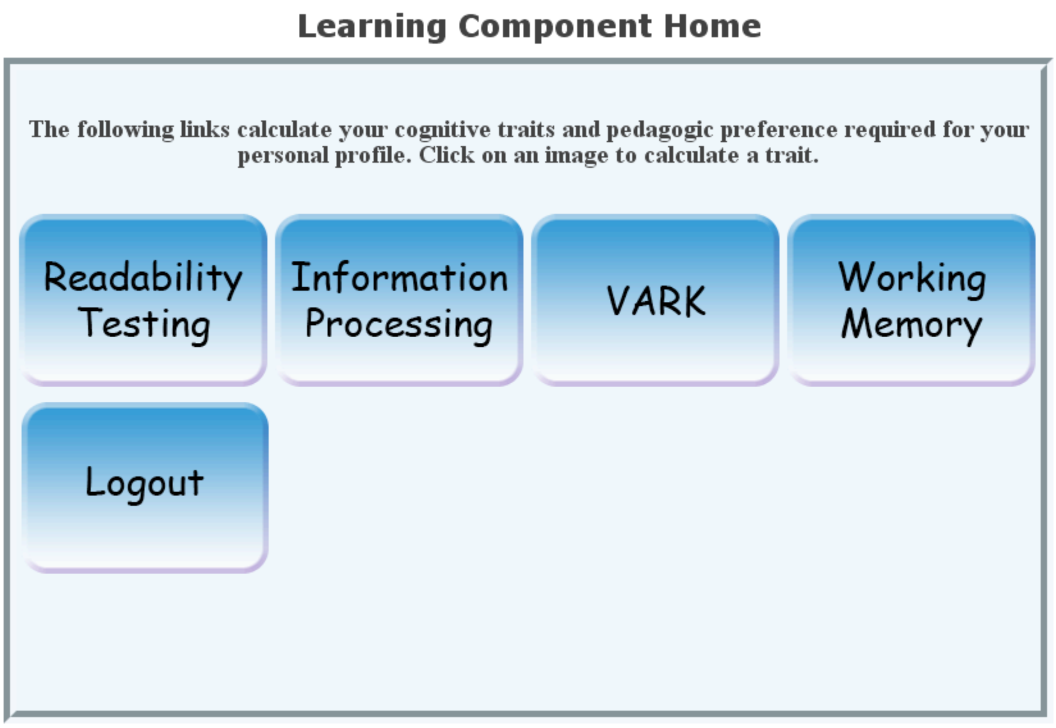


Figure 6.2: Tests to calculate the Personal Profile traits

However, if the learner has already completed these tests then the learner can choose a pre-defined course for that learner or choose a new specification. If a new specification is chosen the learning component will harvest the repositories created by the Content Analyser and produce the required course suited to the personal profile of the learner. Figure 6.2 represents a snap shot of the view the learner has once logged into the learning component, prior to conducting the initial tests. The learner must click on the associated link to take the test relating to the element of the personal profile being tested. The following subsections details each test that must be completed before the learner can view the available specifications.

6.2.2 Working Memory Capacity Test

The working memory capacity of the learner is calculated using an N-Back algorithm. Owen et∼al [15] showed that using an N-Back algorithm method for testing working memory capacity stimulates the same regions of the human brain when compared with the more established working memory tests, by performing a meta-analysis of normative functional neuroimaging studies. The N-Back strategy was chosen as the most appropriate method as the delivery of the N-Back test will be conducted in the same method and environment that the learner will interact with the instructional content. It was decided that only a visual representation would be given to learners throughout the N-Back test as typical machine voices are very robotic and it is beyond the scope of this research to investigate the effectiveness of a robotic voice engaging with a learner throughout a learning experience. Figure 6.3 depicts the learners view once the Working Memory test has been selected. The initial screen briefly explains to the leaner the simple protocol of an N-Back algorithm, whereby learners are shown images one-by-one and must remember every image location in terms of how many images have appeared since a given image. As can be seen in Figure 6.3 the leaner is being shown images of fruit and must remember the order in which the images appeared. The selection of fruit and vegetables to be displayed as the components for the working memory test was chosen as the demographic of a typical third level student would be familiar with all the elements of the test. It was important everyday elements were selected as

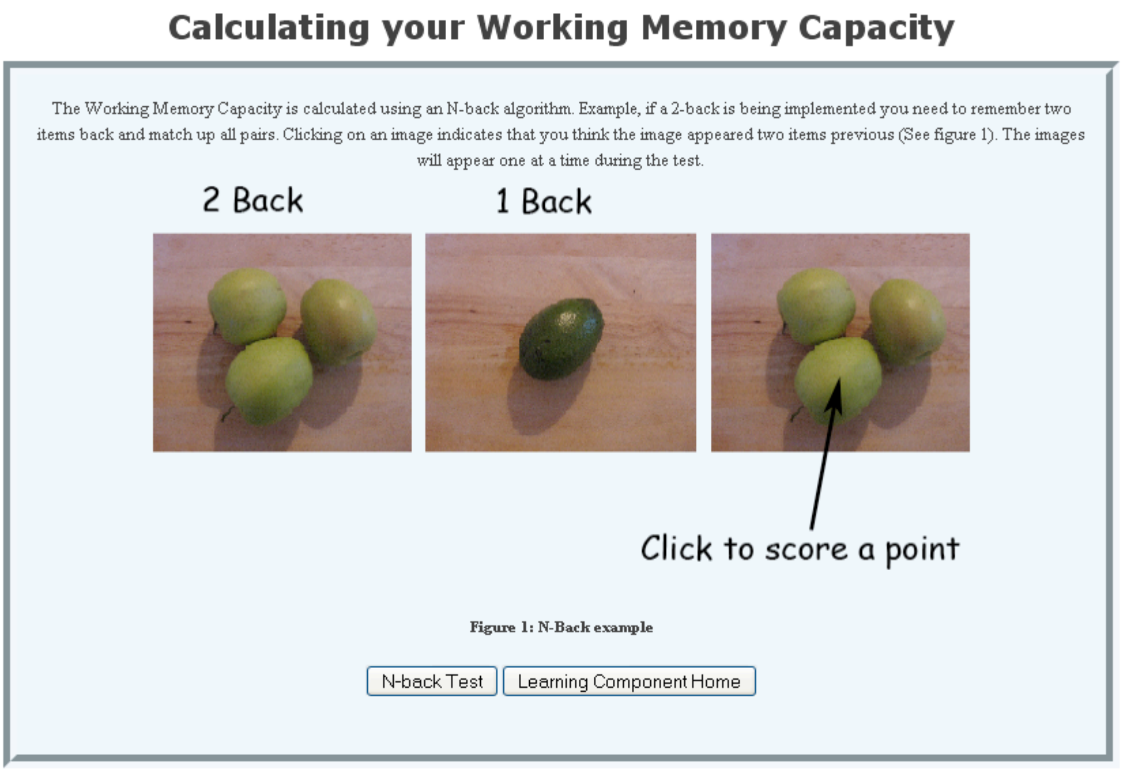


Figure 6.3: NBack algorithm for calculating the working memory of an individual

the WMC would be different depending on the ease at which the elements were perceived. Using such elements it is believed would be more representative of our WMC model as throughout our online engagement with learners, learners will be given instructional content mapped close to their personal profile.

Learners start the working memory test with a two back implementation. The learners are shown a sequence of twenty images within a two back implementation. A point is awarded to the learners score if the learner selects an appropriate image that has appeared N-Back and the learner is also awarded a point if the learner successfully does not select an image that has occurred N-Back. Learners are also decremented points if they falsely select or do not select an image. The learners proceed onto the next level of the N-Back test if they successfully succeed to get a score of eighty percent in any nback iteration.

6.2.3 IPS and knowledge acquisition Test

The IPS and knowledge acquisition is centered on creating a metric that represents the information processing of the learner. Once the learner has selected the

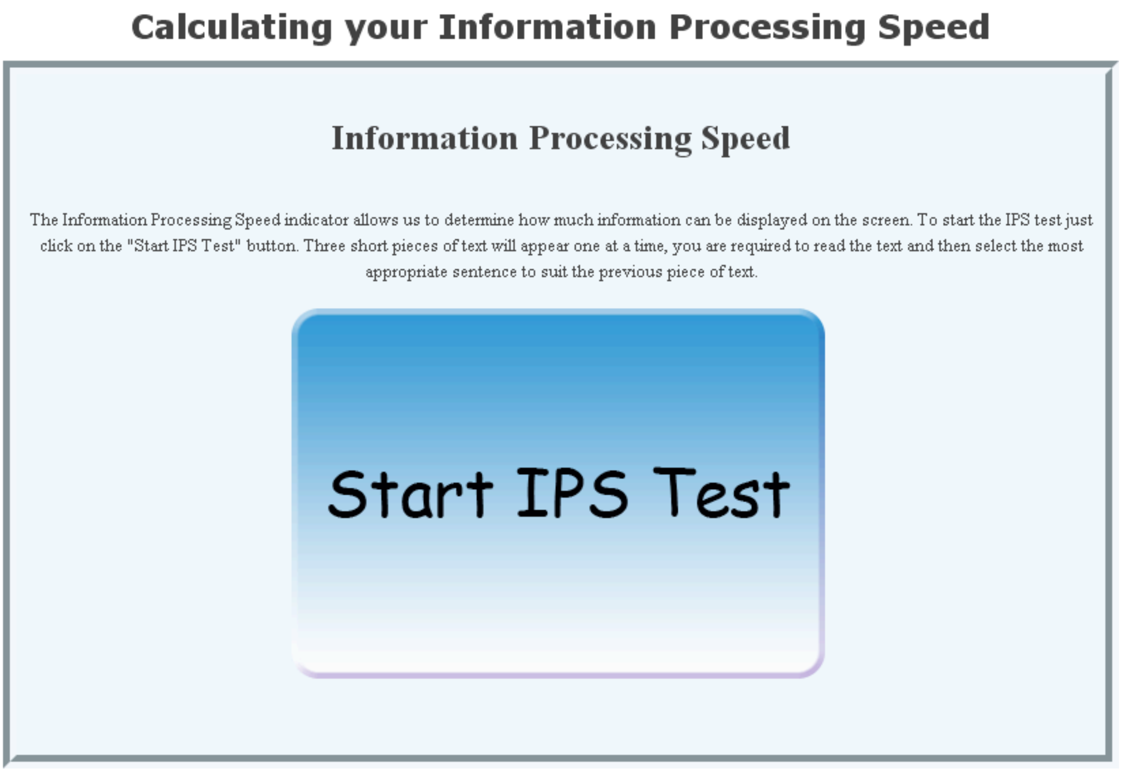


Figure 6.4: Tests to calculate the Information Processing Speed

IPS Test the learner is shown Figure 6.4. The IPS Test evaluates the learners ability to read unfamiliar, diverse pieces of instructional content delivered to the learner in the same manner as a typical learning experience using the learning component. The result of the test indicates how long it took the learner to read through the instructional content using the system clock and the accuracy of the knowledge acquisition.

After each piece of instructional content the learner is given four sentences to choose which sentence most accurately describes the previous piece of instructional content. The learner is awarded marks depending on the sentence selected. The pieces of instructional content that are displayed to the learner throughout the IPS test were chosen as the typical demographic of learners taken part in the testing of the effectiveness of the selection model will have little experience in the testing domain, consequently the testing material included instructional content relating to *Genetic Algorithms*, *Limes Disease* and *Cryptography* as a typical first year Computing student would have very little understanding of the testing material. This methodology was also applied to the selection of the domain content.

6.2.4 Readability Test

The readability level of the learner is calculated using the Flesch reading ease metrics [96]. The Readability level of the learner acts to reinforce the WMC of the learner in our model, as the learners will be given instructional content were no prior knowledge exists but the content will be adapted to suit their own style of writing thus reducing the possibility of interference throughout the learning experience. Interference can occur within a learning experience when the flow of instructional content is untimely broken: either as an external migration of domain knowledge or difficulty in understanding the textual information. The level of the language used will only act as a minor indicator for the evolving content. The learner is given an empty text area to compile a piece of text as seen in Figure 6.5. The exercise is not time bound and once the learner is completed the metrics are calculated and the personal profile is updated.

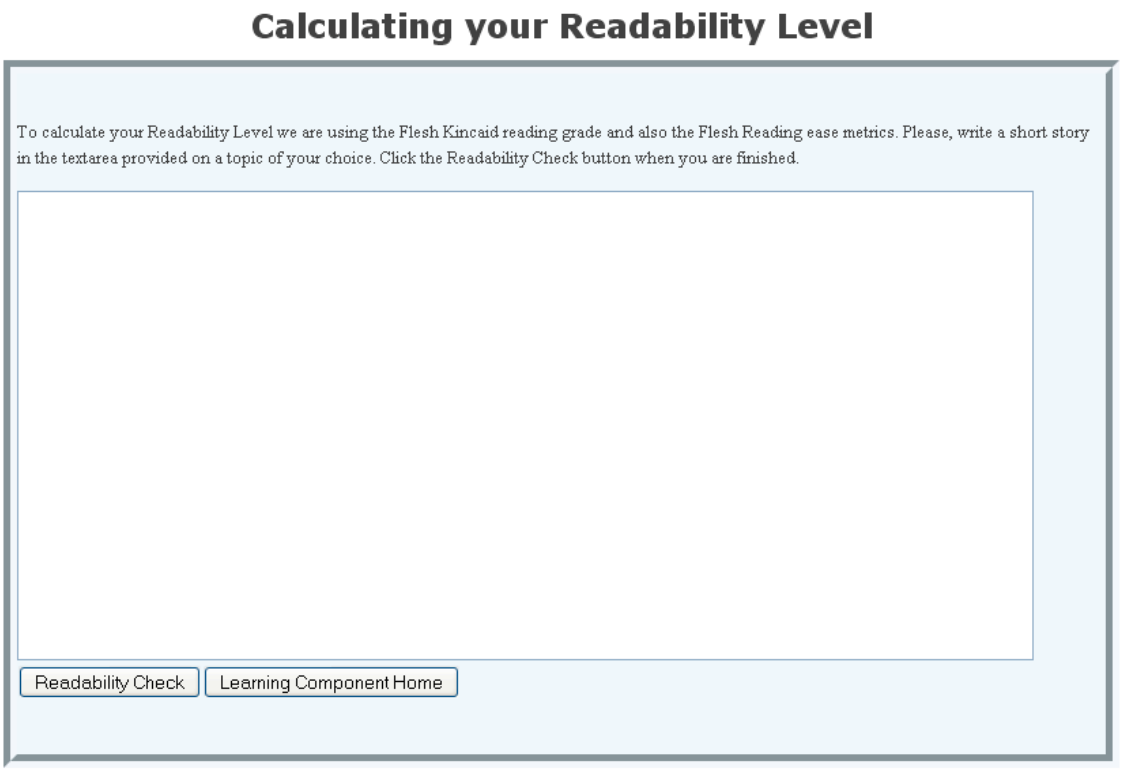


Figure 6.5: Calculating the Readability level

6.2.5 VARK Test

The pedagogic preference of the learner is represented as an estimation of the delivery mode of the learning content and is mapped to reflect the WMC of the learner and also the VARK process of calculating a preference for a mode of learning. The VARK component of the pedagogic preference of the learner is calculated using the VARK questioner as developed by Neil Fleming[1], seen in Figure 6.6. VARK is an acronym made from the initial four means of communication (Visual, Aural, Read / Write and Kinesthetic). Learners use these modes when they are taking in or given out information. They also have preferences for one or more modes of learning. Within an online learning environment these modes are restricted to Visual and Read / Write. Robotic voices have been omitted for possible inclusion to include the Aural mode of learning as discussed earlier in this Chapter. The VARK test is composed of a questioner with a number of radio buttons as possible answers. On each question a learner can choose multiple answers, for more information see [1].

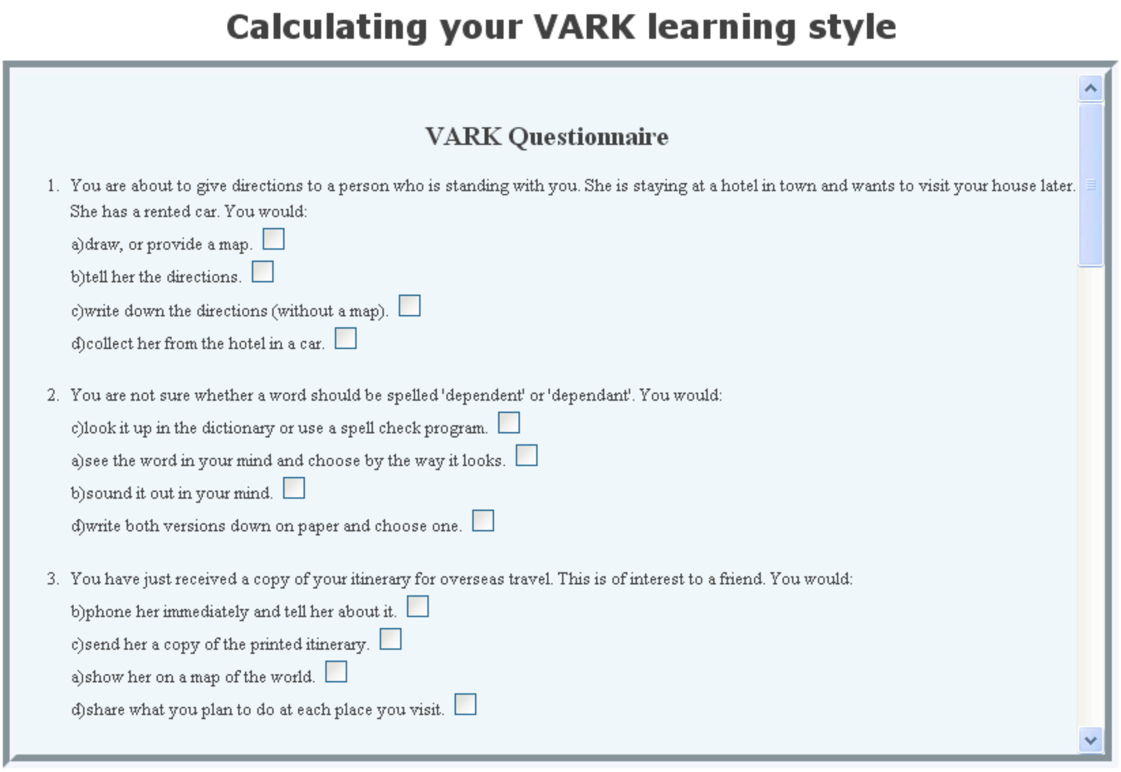


Figure 6.6: VARK questioner designed by Neil Flemmon [1]

6.2.6 Summary

In summary, this section discussed the tests to calculate the traits defined in chapter three to generate a suitable personal profile for a learner interacting with the learning component. Once the learner has concluded all the tests the learner is given extra functionality to view all the available specifications. When a learner selects a specification the Selection model retrieves the learners personal profile and constructs a unique ideal specification suited to the personal profile of the learner (as discussed in chapter five). The Selection model uses a genetic algorithm to construct a course suited to the ideal specification that has a predefined minimum expected learning experience. The following section details additionally functionality available to authors of instructional content. In particular the next section discusses functionality to create a specification and analyze instructional content.

6.3 Author functionality

When a user is classified as an author they are given extra functionality to create a specification and analyze a file as seen in Figure 6.7. Chapter five details the metadata requirements for both a specification and a concept. The metadata elements used to describe the specification are typical SCORM metadata elements as defined by the SCORM Run Time Environment (SCORM RTE) as seen in Figure 6.8. Additionally there exists metadata requirements associated with a specification to control the evolution process for optimal content generation. The *summary* field is used by the learning component to show a summary of available specifications to potential learners. The Minimum Expected Learning Experience (MELE) is set by an author when constructing the specification (detailed in chapter five). The default value for this field is seventy percent to allow for flexibility in both the learner exceeding expectations and the evolutionary strategy finding optimal courses. The author also sets the Cognitive Traits field, indicating which cognitive trait is of greater importance. For example, if the author needs to generate instructional content that is focused on the working memory of the learner then the author would select the appropriate CT value.

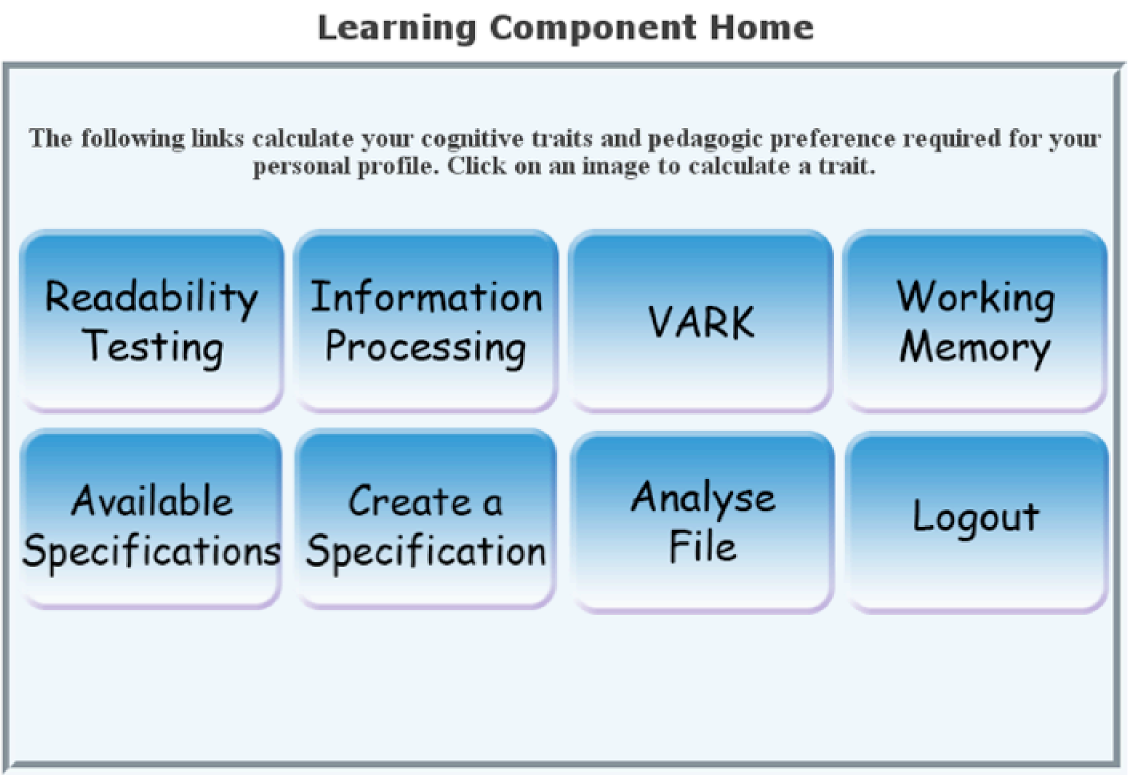


Figure 6.7: Test to calculate the VARK score of a Learner

Figure 6.9 illustrates all the metadata elements associated with a concept defined within a specification. All the elements are defined by the SCORM RTE with the exception of the *Typical learning time* as discussed in chapter five.

6.3.1 Summary

This section described the author functionality associated with the learning component. Each author is allowed to construct specifications, which in turn are dis-

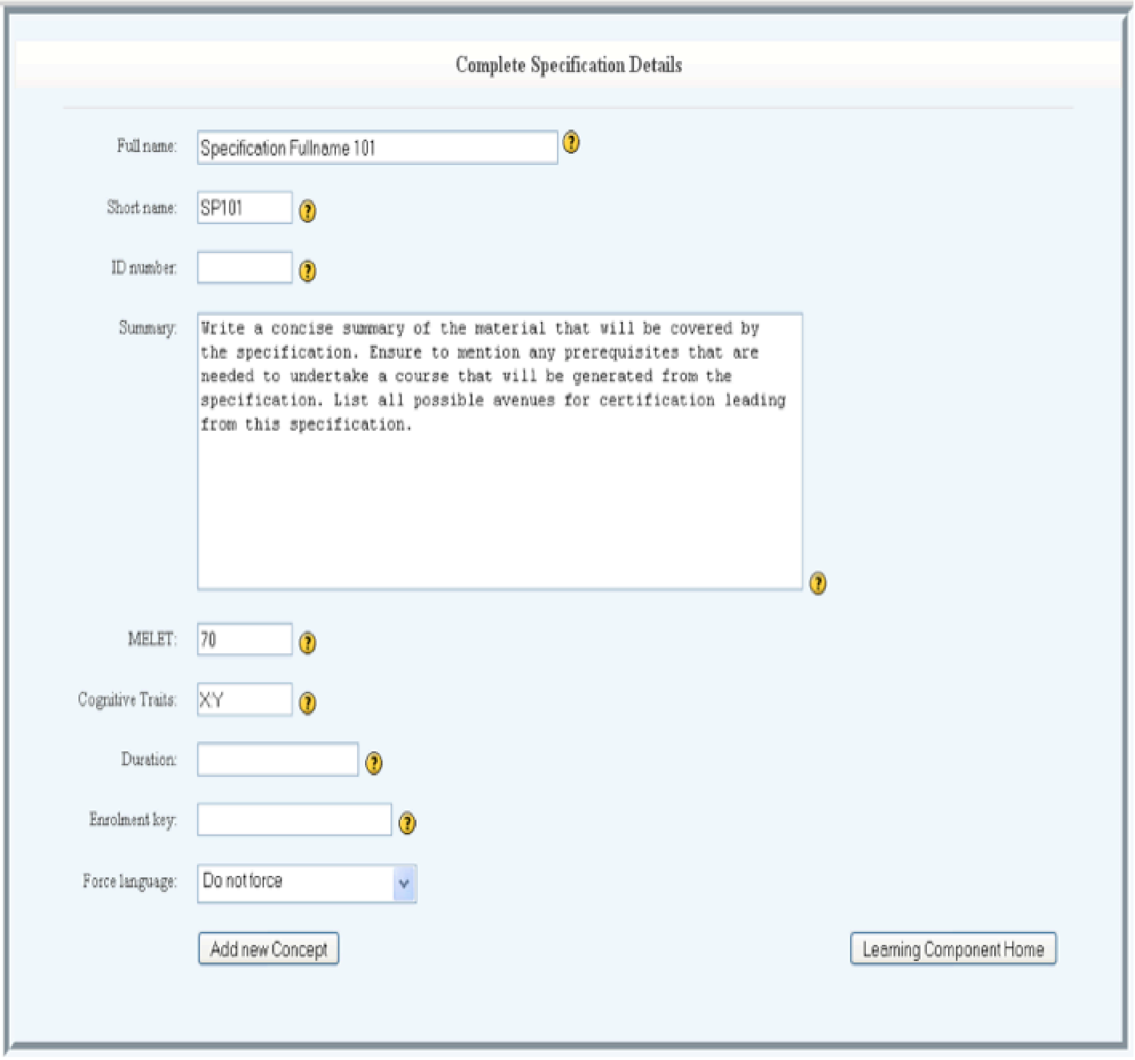


Figure 6.8: Test to calculate the VARK score of a Learner

Figure 6.9: Test to calculate the VARK score of a Learner

played to potential learners. The learning component that was designed allows an author to set two unique fields: *MELE* and the *Cognitive Traits* fields. The MELE field is used by the evolutionary component to evolve instructional content to a pre-determined minimum expected learning experience. This feature, unlike traditional Adaptive Hypermedia Systems (AHS) enables the automatic generation of instructional content independent of the author of the specification. Additionally, the evolutionary process will not terminate until the MELE is reached. The Cognitive Traits field is used by authors of instructional content to emphasize greater importance for a cognitive trait. This feature can be very useful as a research tool in determining the effects of different types of content on populations of learners.

6.4 Conclusion

This chapter discussed the migration of the learning component into a suitable CMS / LMS and the front end user experience involved with utilizing the leaning component. Moodle was chosen as an appropriate CMS for the learning component. This chapter discussed the requirements for such an environment but also detailed that the learning component is a self contained unit that was simply embedded into the CMS. The tests required for generating the personal profile, detailed in Chapter three were discussed. In addition, the chapter also described the core functionality uniquely associated with the learning component: MELE and Cognitive Traits. These additional components allow the learning component to approximate a suitable course to an expected learning outcome matched against a unique ideal specification generated by the learning component. Unlike traditional AHS the author of the specification sets the expectation but is not involved with generating instructional content for any learner.

The next chapter is involved with testing the learning component. A number of studies are carried out to determine the effectiveness of the learning component. Firstly, an investigation of the correlation between the cognitive ability and pedagogic preference of instructional authors and the metadata, produced by the CA detailing the cognitive metrics found within the instructional content generated by the authors is carried out. This is used to determine if an author is consistent when generating instructional content. Consequently, if the traits of an instructional author are measured will those traits reflect the content being produced. If a strong correlation exists; suitable authors can be easily matched up with suitable learners and the learning component will need to be upgraded and a traditional AHS would be more suitable for adapting content to suit the needs of learners. Additionally the learning components performance is measured against a traditional class room environment and also the learning components evolutionary process is analysed in determining the effectiveness of the evaluation criteria.

Chapter 7

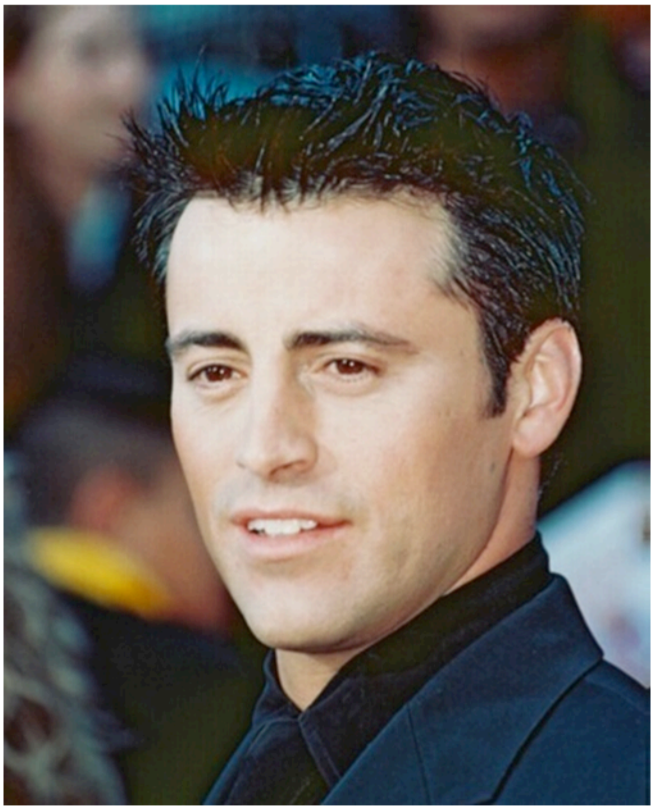
Learning Component Evaluation

This chapter is involved with the evaluation of the learning component. Initially this chapter investigates the use of student developed content to use as a repository for the learning component. The chapter also focuses on an analysis investigating the correlation between the cognitive ability and pedagogic preference of an author of instructional content and the identified cognitive metrics found within instructional content generated by an author. Eight instructional authors generated content based on a simple specification for a short course on UML. The authors were, firstly required to participate in the previous experiments (discussed in Chapter six) to obtain their personal profile scores. Secondly, the authors generated content that conformed to the requirements of the Content Analyser (as described in Chapter four). An analysis was performed investigating the hypothesis that instructional authors are not consistent when generating instructional content, additionally there is no connection between the cognitive traits of an author and the cognitive metrics that are produced when creating content, thus reducing the possibility of AHS being incorporated into real world systems. This chapter is also involved with the evaluation of the learning component. The chapter discusses experiments conducted to determine the performance of the learning component against a traditional lecturing experience. An in-depth analysis is carried out to investigate the correlation between the minimum expected learning experience (determined by the evolutionary process in constructing instructional content) and the actual result obtained by each learner after completing a quiz following a learning experience. Additionally the chapter investigates the correlation between the individual traits and the results obtained by taken a quiz after a learning experience has concluded. The following subsections firstly describe *Pearsons* correlation process and then discuss the correlation between the cognitive traits of an author and the cognitive metrics found within metadata generated by the Content Analyser.

7.1 Student Content

It was decided to develop a short course on introduction concepts relating to UML to evaluate the success of the learning component. The success of the learning component is a measure of the ability of the Selection model to produce instructional courses suited to the cognitive ability and pedagogical preference of a learner as described in a suitable personal profile in Chapter 3. The first database of content that was generated was by students. The students that were identified as suitable candidates to generate instructional content were second year Higher Certificate students that had completed a module on UML and covered all the learning outcomes associated with the short course on UML. Students were also seen as ideal candidates as they would be closely matched to the to the proposed target audience of first year students in terms of academic level and technical writing abilities. Twenty students generated instructional content to form the repository. Students were given access to the internet and their notes and enough time to produce instructional content. The content was generated by individual students and not in a collaborative environment. This content was analysed to ensure that the leaning outcomes would be covered by the student generated material. Unfortunately, the students used TXT language throughout the generated content and also images to reflect technical terms that were not related (in terms of the context of the instructional content) as seen in Figure 7.1. It was decided that TXT language would not be suitable as some students would not be familiar with the language and also the academic quality of the content did not appropriately cover the learning material.

Actor



An actor interacts with ur system

Figure 7.1: Student generated data, describing a UML Actor.

The following section details the process involved with generating, author instructional content. An analysis is discussed investigating the hypothesis that instructional authors are not consistent when generating instructional content.

7.2 Instructional Authors

Eight instructional authors were involved in generating instructional content. Initially the authors completed the tests defined in chapter six to identify their personal profile and then all authors generated instructional content suited to a specification for a short course on UML. The following subsections use Pearsons correlation to determine linear dependance between each of the identified traits and the metadata produced by the Content Analyser. In addition the correlation between the metadata produced for each of the concepts by the CA and the metrics associated with the authors personal profile is calculated to determine the consistency of the author when generating instructional content in terms of the cognitive metrics that the author uses when generating instructional content.

The following subsections investigate the correlation between the metadata produced by the Content Analyser and the cognitive ability and pedagogical preference of an instructional author.

7.2.1 WMC and metadata

The *Pearson* correlation between all metadata describing all concepts associated with working memory was calculated as -0.1359089. This result means that no correlation exists on a global scale for WMC. Table 7.1 details the correlation between WMC and the metadata generated for each individual concept to determine if there exits a trend across all concepts or if the WMC trait is dependent on the authors interpretation of the concept. The WMC of an individual is seen as a constant trait (cognitive trait) that can be improved upon over time as seen by Kinshuk [8]. If the trait remained constant it would be conceivable to allow an author of instructional content generate a complete course for a particular learner as a simple pairing method could be used to match up suitable authors with suitable learners, however if there exists a huge variance with the trait across multiple concepts this simple pairing process would not be suitable.

Trait	**Concept**	**Correlation**
WMC	Actor	-0.01911375
WMC	Functional Requirements	-0.5846814
WMC	Relationship	-0.652822
WMC	Use-Case	0.1910910
WMC	Generalisation	-0.08211009

Table 7.1: This table shows the correlation between the WMC of an author and the WMC metadata that was generated for each of the concepts using the Content Analyser.

It can be clearly seen in Table 7.1 that there exists a significant variance between the WMC of an author and the WMC metadata that was generated for each of the concepts using the Content Analyser. The following section investigates the correlation between the Readability and the metadata produced to classify an author and the metadata to classify the instructional material.

7.2.2 Readability and metadata

The *Pearson* correlation between all metadata describing all concepts associated with readability was calculated as -0.03904613. Table 7.2 details the correlation between the Readability level of an author and the metadata generated for each individual concept to determine if there exits a trend across all concepts or if the Readability level is dependent on the authors interpretation of the concept.

Trait	**Concept**	**Correlation**
WMC	Actor	0.6699855
WMC	Functional Requirements	-0.2587091
WMC	Relationship	-0.4500881
WMC	Use-Case	0.04846115
WMC	Generalisation	0.1025095

Table 7.2: This table shows the correlation between the Readability of an author and the Readability metadata that was generated for each of the concepts using the Content Analyser.

It can be clearly seen in Table 7.2 that there exists a significant variance between the Readability of an author and the Readability metadata that was generated for each of the concepts using the Content Analyser. The following section investigates the correlation between the VARK and the metadata produced to classify an author and the metadata to classify the instructional material.

7.2.3 VARK and metadata

The *Pearson* correlation between all metadata describing all concepts associated with VARK was calculated as 0.04493267. Table 7.3 details the correlation between the VARK representation (described in Chapter four) of an author and the metadata generated for each individual concept to determine if there exits a trend across all concepts or if the VARK level is dependent on the authors interpretation of the concept.

Trait	**Concept**	**Correlation**
WMC	Actor	0.5007831
WMC	Functional Requirements	0.591608
WMC	Relationship	0.2639435
WMC	Use-Case	-0.1490301
WMC	Generalisation	-0.1506956

Table 7.3: This table shows the correlation between the VARK representation of an author and the VARK metadata that was generated for each of the concepts using the Content Analyser.

It can be clearly seen in Table 7.3 that there exists a significant variance between the VARK representation of an author and the VARK metadata that was generated for each of the concepts using the Content Analyser.

7.2.4 Summary

The previous subsections were involved with an investigation of the consistency of authors to generate instructional content. In particular eight authors were given the task of creating instructional content suited to a module descriptor for a short course on UML. These authors were required to complete personal profile tests, as described in Chapter six to calculate their own personal profile. This profile was subsequently used as the evaluation criteria for determining the correlation between the authors and the metadata produced for the courses that were constructed. The investigation determined if a suitable author was found for a suitable learner, using the metrics described within the personal profile as discussed in Chapter four, would mean that an author would be able to create mathemagenic content for the learner across multiple domains. It was found that an author could not create consistent (in terms of cognitive metrics found within the instructional content) instructional content within the context of a short course on UML. Furthermore it was found that an author does not create content suited to their own personal profile, so matching an author to a suitable learner using the metrics described within the personal profile would not be recommended. In summary, there exists inconsistencies when generating content, between learning objects and matching the cognitive metrics to the author of instructional material. These results demonstrate that an automated component should be used to create instructional content avoiding traditional approaches of content adaptation, such as, AHS. The following section is involved with an evaluation of the learning component that was created to overcome the inconsistencies within learning object repositories, referencing standards and traditional adaptive learning systems.

7.3 Evaluation of learning component

This section is involved with an evaluation of the learning component. Thirty nine students took part in the evaluation process of the learning component. Initially all the students completed a survey to determine any previous experiential learning in relation to UML. The surveys showed that no student had any previous learning experience with UML content. The students then completed all the tests as discussed in Chapter six to determine their cognitive ability and pedagogic preference. The tests were carried out in a studio classroom environment, where each student had ample room and access to their own computer for the duration of the experiments. Once students completed the initial tests the student were randomly divided into two cohorts: one group to be subjected to a traditional introductory lecture on UML (see Appendix A for Module descriptor and details of learning outcomes for short course on UML, Appendix B for lecture slides from typical classroom environment and Appendix C for an example of content generated for a learner participating using the learning component) and the other group remained in the studio classroom to participate in an introductory lecture on UML developed by the learning component for each individual learner. Students using the learning component were monitored by two laboratory attendants to ensure that once the student had completed the learning content that the monitor was switched off. Both groups were not allowed to take notes throughout the learning experience and completed a short quiz on UML (see Appendix D for quiz and marking scheme) after their learning experience had concluded. The following sub sections investigate the validity of the experimentation detailing the validity of assessment (using multiple examiners to mark examinations) and group selection. Additionally a comparison of the results obtained is discussed.

7.3.1 Evaluation process

The validity of the evaluation process can be divided into the following categories: personal profile creation and environmental contexts of the learning environment, group selection, and the examination of scripts. Each of these categories is discussed further in the following subsections.

7.3.1.1 Environmental contexts

Thirty nine students took part in the evaluation process of the learning component. The environmental contexts of the learning environment include any previous experience within the desired domain, the physical environment within the learning environment and any other restrictions that may influence the learning experience. The following bullet points explore these characteristics of the learning environment:

- Initially all the students completed a survey to determine any previous experiential learning in relation to UML. The surveys showed that no student had any previous learning experience with UML content.

- The students then completed all the tests as discussed in chapter six to determine their cognitive ability and pedagogic preference. The tests were carried out in a studio classroom environment, where each student had ample room and access to their own computer for the duration of the experiments. After the students were divided into two cohorts, the students that remained within the studio classroom using the automated component had access to their own computer for the remainder of the evaluation process. The other cohort of students that were taking part in a traditional environment were in a classroom with no access to computers to reduce the possibility of interference.

- The cohort of students that were taken part with the automated learning component had an additional two classroom attendants present that were advised to ensure that the monitor remained off during the examination of the learning content after the learning experience had concluded.

The following section investigates the group selection protocol that was used to divide the groups into two categories.

7.3.1.2 Group Selection

Group selection is a critical component of the evaluation process in order to ensure that both groups consist of an even distribution of the cognitive traits that were identified in chapter three as an ideal profile for an adaptive learning environment. The groups were randomly selected to participate in a learning experience once the cognitive ability and pedagogic preference of the learner had been calculated. Table 7.4 illustrates the categories of learners within each group. The classification of learners identifies the range of results obtained by all learners participating in the evaluation process. The key influencial traits that the learners are described are: working memory capacity, the readability level and the VARK score.

The following bullet points investigate the groups of learners identifies by their personal profile characteristics:

- The classification of learners participating in a working memory test is seen in Table 7.5. It can be clearly seen in Table 7.4 that both groups had a good spread of learners, however there exists a grouping within the automated component group with a low WMC category. It is envisaged that this grouping could have a negative impact on the learning potential of the cohort if suitable adapted content was not found by the automated component.

Traits	**Category**	**Traditional Lecture**	**Automated Environment**
WMC	Very High	4	1
	High	5	5
	Medium	3	4
	Low	5	10
	Very Low	3	1
Readability	Very High	0	0
	High	0	0
	Medium	7	4
	Low	11	12
	Very Low	2	3
VARK	K Very Strong	7	3
	V Very Strong	0	1
	A Very Strong	2	2
	R Very Strong	0	1
	KA	3	2
	RK	0	1
	VRK	0	1
	VAK	0	1
	VARK	8	7

Table 7.4: This table illustrates the categories of learners within the groups selected to evaluate the learning component.

Category	**WMC**	**Readability**
Very High	4.1+	0 –>29
High	3.6 –>4	30 –>49
Medium	3.1 – >3.5	50 –>69
Low	2.6 –>3	70 –>89
Very Low	2 –>2.5	90+

Table 7.5: This table illustrates the categories of learners for working memory capacity and readability.

- Table 7.4 illustrates the categories of learners identified after participating in a readability test. Throughout the readability test learners were informed of the freedom of language used throughout the test; no record was kept of the actual text that was created only the score of the readability test was stored. It can be clearly seen in Table 7.4 that both groups are comparable with little differences between the groups.
- It can be seen in Table 7.4 that there exists a wide spread of categories of learners within the VARK section from strong single preferences through to multi-modal preferences including all four VARK traits (Visual, Aural, Read-Write, and Kinestic). The single most substantial grouping within both groups is VARK.

The following section investigates the examination process for evaluating the scripts of the learners taken part in both the automated course and the traditional course.

7.3.1.3 Examination process

The two cohorts of learners were subjected to personal profile tests, completed some instructional course and then completed a short examination on the learning material.

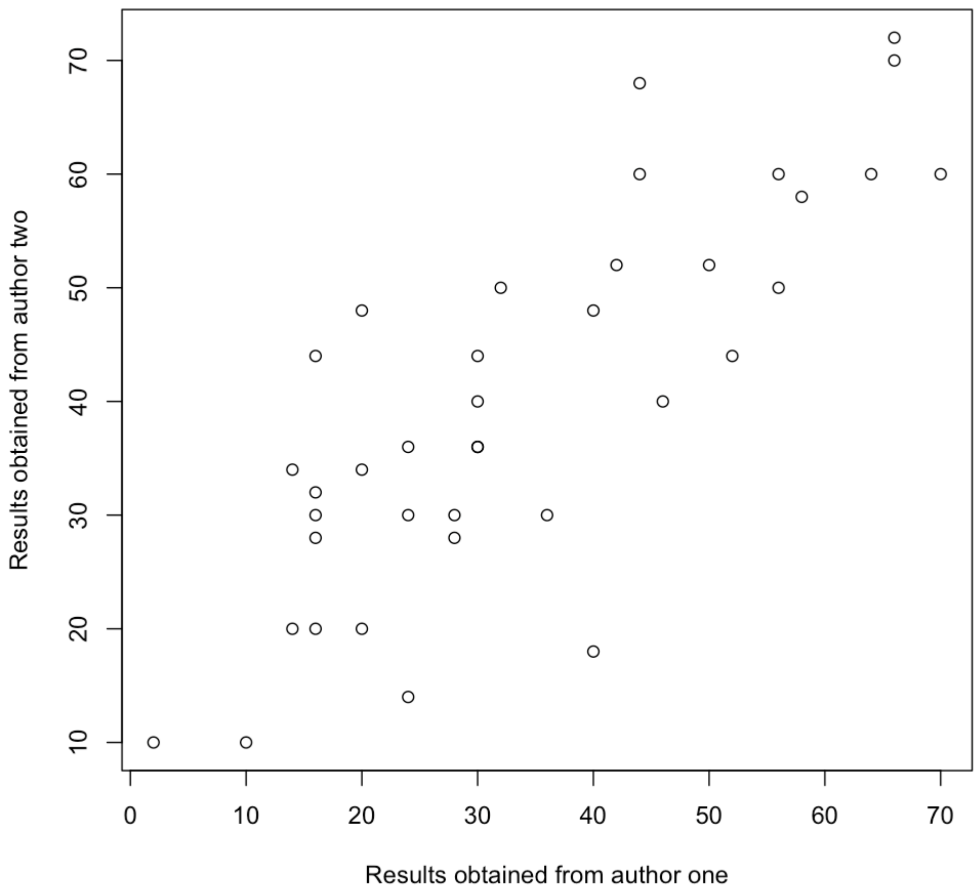


Figure 7.2: Comparing the results obtained from two independent examiners

The examinations were corrected by two independent examiners with no knowledge of which type of learner completed the answer sheet to ensure that the examination process was not reflective of a single examiners interpretation of the examination scripts. In addition a blind marking process was implemented also to ensure that an examiner could not determine the results of the other examiner. A correlation between the two sets of results was carried out to ensure the consistency and validity of the results obtained, which yielded a correlation of 0.806, as seen in Figure 7.2. Since there existed a strong correlation between both sets of results no further investigation was conducted to determine the validity of the results obtained. Further investigation of the results obtained by the learners uses the average of both results determined by the examiners.

7.3.2 Summary

In Summary the environmental contexts for the evaluation of the learning component were designed to ensure that no interference occurred disrupting the learning experience of the individual learners. In particular learners were given enough time and access to their own machine within a studio classroom environment to complete the personal profile tests. Learners were randomly divided into two groups of learners. These groups were analysed to ensure that the groups were comparable in terms of the personal profile traits of each learner. This section also discussed the examination process involved within the learning component evaluation to ensure that an examiner was consistent. This process consisted of a double marking blind process were each author was not aware of the category of the learner or the mark obtained by the learner. The following section investigates the potential of the learning component as a suitable learning instrument to replace / supplement a traditional learning approach. In particular the following section is involved with a comparison of the results obtained by the learners using the learning component against traditional learners within a classroom environment.

7.4 Learning Component Performance

This section is involved with determining the effectiveness of the learning component against a traditional lecturing experience. Firstly, this section compares the performance of learners using the learning component against learners within a traditional environment. The section also investigates the effectiveness of the *fitness function* in identifying a suitable Minimum Expected Learning Experience (MELE) threshold discussed in chapter five and analyses the linear correlation between the MELE and the actual results of a learner.

7.4.1 Learning Component against a Traditional Lecture

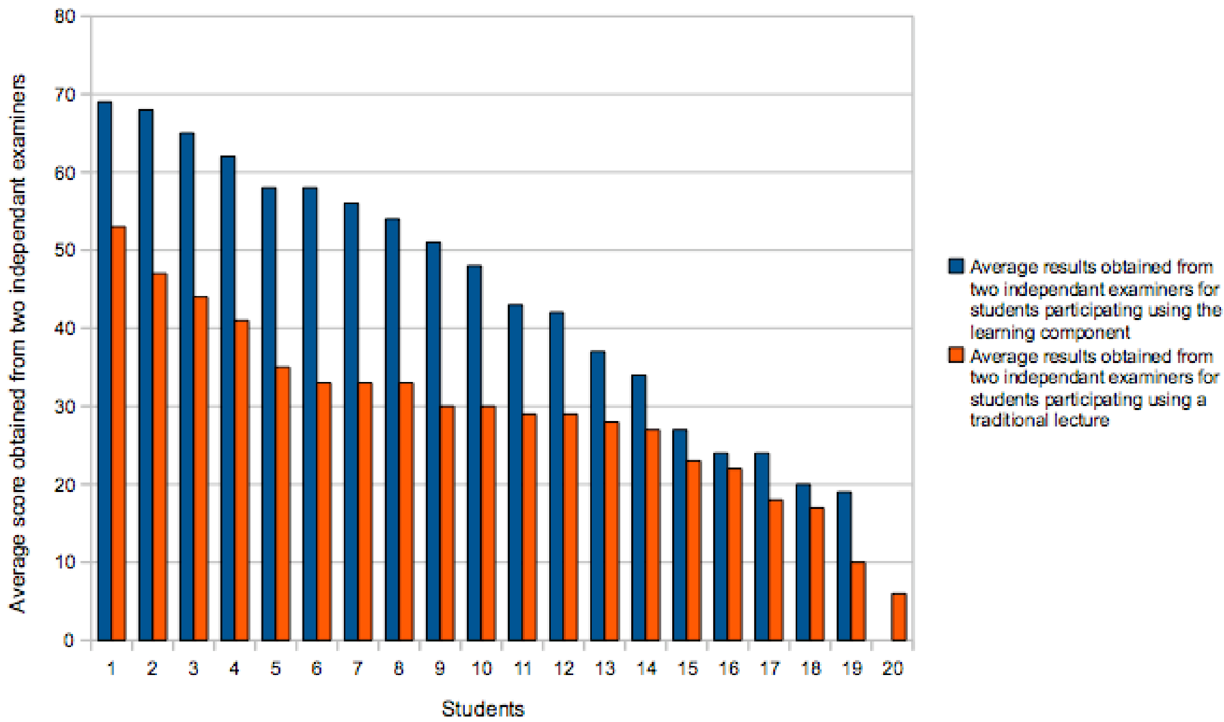


Figure 7.3: Comparison of the results obtained by students interacting with the learning component against students within a typical lecturing environment

Thirty nine students participated in the evaluation of the learning component as discussed earlier. Figure 7.3 illustrates the average results obtained after two independent examiners corrected the UML quiz after all learning experiences concluded. It can be clearly seen in Figure 7.3 that the learners interacting with the learning component outperformed the learners that were subjected to a traditional lecture. The students participating using the adaptive component out performed the students participating in a traditional lecture on average by 15.71 %. Appendix C and E give examples of automated course output from learners interacting with the learning component. Appendix C shows a course that was developed for a learner that has a medium Working Memory Capacity, and weak visual preference. It can be clearly seen that there exists very few external links to the core learning experience reducing the possibility of interference during the learning experience. Additionally the visual constructs are present but not dominating the instructional content. In contrast, Appendix E shows a course that was developed for a learner with High Working Memory Capacity and strong visual preference. It can be clearly seen that there are additional external links with additional content including strong visual constructs.

The following subsections investigate the effectiveness of the *fitness function* and also the significance of each of the traits within the personal profile to determine the most significant trait associated with the improved performance of the learners interacting with the learning component.

7.4.2 Correlation between the MELE and the actual result

The effectiveness of the *fitness function* is a measure of the correlation between the MELE for each learner and the actual result obtained after completing the learning experience. Throughout the evolutionary process the minimum expected learning experience (MELE) was initial set to seventy percent. However, due to the small database that exists for the testing phase (due to financial constraints), as described in chapter six a degrading element was incorporated into the evolutionary component to ensure that each evolutionary process would produce an optimal course for each individual learner interacting with the learning component. Figure 7.4 shows the results obtained from an examination by the learners using the learning component against the MELE for each learner.

The correlation between the MELE and the actual results obtained was 0.274 which implies a weak positive correlation. Further investigation identified two possible outliers, when removed yielded a correlation of 0.57. The potential outliers were not removed as both possible outliers where within two standard deviations of the mean of the results. The following subsection discusses a covariance analysis that was conducted to determine the significance of the difference between both groups and in particular to identify the most significant trait from the personal profile.

7.4.3 Covariance analysis

A covariance analysis was conducted to determine the significance of the differences between both groups and in particular identify the most influential traits used in the creation of the instructional content.

Table 7.6 details the results obtained by the covariance analysis. It can be clearly seen that there exists a strong significance of 0.00208 (probability of error) between the groups, however it can also be seen that the traits that were selected are not that significant in the difference between the results, i.e. the percentage error that exists with the independent variables ranges from 40% up to 90 %. This analysis further supports the weak positive correlation between the minimum expected

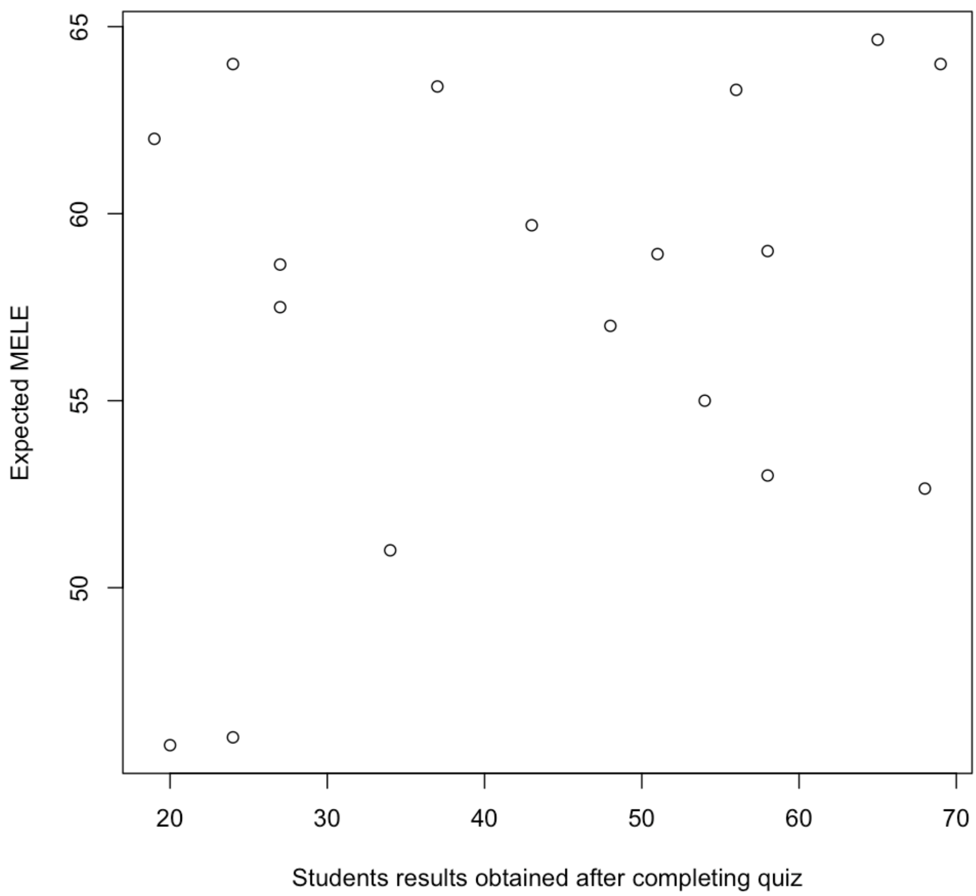


Figure 7.4: Comparing the results obtained from two independent examiners against the minimum expected learning experience, calculated by the evolutionary process

Traits	Values
Readability	0.915
VARK	0.403
WMC	0.44481
Groups	0.00208

Table 7.6: This table illustrates the percentage probability error associated with each trait selected.

learning experience and the actual results obtained. In conclusion, the significant difference cannot be determined by the selected traits, however there exists an extra factor that has not been identified. The content was modified to suit the traits of the individuals, however the extra factor could be involved with motivational issues in using a new learning component or some other factor.

7.4.4 Summary

This section investigated the effectiveness of the learning component against a traditional lecturing experience. This section showed that there existed a significant difference between learners using the learning component and learners interacting with a traditional lecturing experience. It was found that students participating using the adaptive component out performed the students participating in a traditional lecture on average by 15.71 %. In addition, this section discussed a comprehensive analysis to determine the effectiveness of the *fitness function* (used within the evolutionary algorithm when evolving instructional content) in determining the minimum expected learning experience and the introduction of a degrading factor to ensure course creation on each evolutionary iteration. Furthermore, an analysis was conducted to determine the most significant trait in identifying suitable content. In conclusion a significant difference was found between the groups as seen in Figure 7.3, however this difference could not be explained by the traits identified within the personal profile.

7.5 Conclusion

This chapter discussed the evaluation of the learning component. This evaluation, firstly, investigating the consistency of authors to generate instructional content. The investigation determined if a suitable author was found for a suitable learner, using the metrics described within the personal profile as discussed in chapter four, would that author be able to create mathemagenic content for the learner across multiple domains. It was found that an author could not create consistent (in terms of cognitive metrics found within the instructional content) instructional content within the context of a short course on UML. Furthermore it was found that an author does not create content suited to their own personal profile, so matching an author to a suitable learner using the metrics described within the personal profile would not be recommended. Secondly, the chapter evaluated the environmental contexts of the learning environment to ensure that no external influences interfered with the learning experiences. In particular learners were given enough time and access to their own machine within a studio classroom environment to complete the personal profile tests. Learners were randomly divided into two groups of learners. These groups were analysed to ensure that the groups were comparable in terms of the personal profile traits of each learner. This chapter also discussed the examination process involved within the learning component evaluation to ensure that an examiner was consistent. This process consisted of a double marking blind process were each author was not aware of the category of the learner or the mark obtained by the learner. Finally, the chapter discussed an evaluation of the learning component. In particular, the chapter determined the effectiveness of the learning component against a traditional lecturing experience. This chapter showed that there existed a significant difference between learners using the learning component and learners interacting with a traditional lecturing experience. It was found that students participating using the adaptive component outperformed the students participating in a traditional lecture on average by 15.71 %. In addition, a comprehensive analysis was discussed to determine the effectiveness of the *fitness function* in determining the minimum expected learning experience. Furthermore, an analysis was conducted to determine the most significant trait in identifying suitable content. In conclusion a significant difference was found between the groups as seen in Figure 7.3, however this difference could not be explained by the traits identified within the personal profile.

Chapter 8

Conclusion and Future Work

It is well documented that the traditional protocol for higher education does not suit each learner. The rhetorical method of lecturing, while presupposing certain domain knowledge and experience, is a very inefficient method of imparting knowledge. Additionally, delivering instructional content in a typical classroom environment creates an infeasible task for a lecturer to adapt content to suit the needs of each learner within the classroom environment. An ideal solution would be to have a one-to-one system, where an instructor generates mathemagenic content for each learner, taking into consideration the cognitive ability and pedagogic preference of the learner. Obviously this is not an ideal situation considering the high increase of learners into higher education. One solution is for higher education to partially traverse into an online learning environment with an element of suitable adaptive content. This chapter is involved with discussing the conclusions from the research conducted to design, build and evaluate a learning component to automatically generate instructional content suited to the cognitive ability and pedagogical preference of a learner, thus increasing the potential learning experiences gained from online instruction. In addition, the chapter discusses the learning component as a framework for higher education and identifies possible extensions to enable the migration of higher education into an online learning environment.

8.1 Conclusions

This research investigated the following research question;

Is it possible to construct an automated learning component that generates instructional content suited to the cognitive ability and pedagogical preference of a learner, independent of domain and ensuring that no meaning is lost from adaptive strategies?

This thesis discussed two contributions to the field of technology enhanced learning, describing a learning component (content analyser and selection model). Firstly the thesis investigated the environmental contexts of a learning environment and identified a suitable personal profile that included the cognitive traits and pedagogic preference of a learner that could be mapped to measurable cognitive metrics within instructional content. The personal profile that was identified, in Chapter Three, was used in the creation of a model-driven approach to metadata creation using the traits within the profile. The thesis introduced a content analyser that bridges the perceived gap between the inconsistencies found within instructional content repositories and metadata standards. The content analyser successfully migrates instructional content from various formats into SCORM compliant content with additional metadata files associated with the cognitive metrics found within the instructional content. Secondly, the thesis introduced a Selection model (centered on the use of a Genetic Algorithm) for content generation, enabling an author to set a minimum expected learning experience, and modifying the weighting factors for the identified traits. The thesis discusses a protocol for creating suitable instructional content to enhance the evolutionary process. The GA uses the metadata that the Content Analyser generated when construction new course material and does not rely on the author of instructional content to generate metadata consequently avoiding the traditional problems associated with metadata creation (as discussed in Chapter 3). A detailed analysis was discussed to create an optimal evolutionary strategy evolving instructional content to suit an individual's cognitive ability and pedagogical preference.

The learning component created instructional content for third level students. This category of student was seen as an ideal category based on the expected growth in third level student numbers as discussed in Chapter two. Additionally, third level students were seen as an ideal category as third level students are established learners and should be able to manage their own learning experience. However, the learning component is not limited to the category of third level student and could easily be used in a commercial environment or at earlier stages of learning. The learning component creates instructional content in a consistent way evolving with the instructional content metadata designed by the CA, adapting to the personal profile of a learner. This framework for content generation bridges the perceived inconsistencies found within a traditional lecturing environment / traditional adaptive hypermedia system and the cognitive ability and pedagogical preference of an author of instructional content. Chapter Seven detailed the inconsistencies found within instructional content between concepts generated by the same author. This further suggests that a learner participating within a traditional lecturing experience is at a disadvantage in terms of content adaption to enhance the learning experience. Additionally this suggests that content within a traditional AHS may not be consistent in terms of cognitive metrics within the instructional content. Due to financial constraints a number of content authors generated a limited amount of instructional content. This content was analysed in terms of the structure and suitability for the content analyser and not the academic quality of the content. Each author generated content according to the module descriptor for a short course on UML. The GA evolves better when there is a large repository (i.e. more possibilities for creating content). If a suitable repository is not available the GA will evolve to the maximum fitness for any given learner as suitable strategies have been included to avoid the GA arriving at a local minimum, however this does not imply that the GA will evolve to the MELE set by the author of the specification. The degrading factor (MELE -2 for every 2,000 epochs) was introduced to ensure that a suitable course was constructed for each learner.

Chapter seven also discussed in detail an evaluation of the learning component. Thirty nine students participated in the evaluation process of the learning component. All students were first year computing students that had no previous experiential learning involved with UML. The learning component out-performed a traditional lecturing approach by 16% on average when delivering an introductory learning unit on UML with the first year students. In addition, a correlation was calculated between the minimum expected learning experience and the actual outcome of an examination after a learning experience had concluded. There exists a weak positive correlation (0.27) between the Minimum Expected Learning Experience (MELE) and the actual outcome obtained by a learner interacting with the learning environment. However, further investigation showed that eliminating two potential outliers resulted in a stronger correlation between the MELE and actual outcome of 0.54. The potential outliers were not eliminated as they were within two standard deviations of the mean. A covariance analysis yielded a strong significance of 0.00208 between the groups, however this difference could not be explained by the traits within the personal profile. Consequently there must exist some external factors that influence this significant difference, for example, motivation whereby students interacting with a novel environment could have been more motivated that students sitting in a typical lecturing environment.

One of the limitations associated with the experiment is the initial time required for creating the learners profile. This could be avoided if traits were chosen that had suitable manifestations that would be identified automatically when a learner is interacting with a learning environment. Working Memory is one such trait as described in Chapter three that has a number of manifestations associated with the interactions of learner within a learning environment. If the GA was going to produce a course to a learner without a profile, the GA could use statistics associated with the age and sex of the learner and then after multiple learning experiences fine tune the traits by modifying the weighting factors and producing specific courses to target individual traits. In conclusion, it is possible to automatically create suitable content conforming to a single referencing standard identifying metadata associated with cognitive metrics found within the instructional content. Additionally, it is feasible to automatically generate instructional content adapted to the cognitive ability and pedagogical preference of a learner in real-time and repackage that content to suit the SCORM standard. Using the learning component yielded an average increase of 16 % per learner throughout a learning experience against a typical learner within a traditional learning environment in the case study described in Chapter 7. However, further investigation is required to determine additional traits that could be included to increase the correlation between the MELE and the actual outcome. In addition, these results cannot be generalised to any group of students participating in any domain area. The framework is designed as a modular architecture that can be adapted to generate instructional content using any pedagogic strategy and is not bound by the parameters of the proposed personal profile. The following section identifies some future work that is now possible due to the modular framework of the learning component.

8.2 Future Work

This chapter concludes with a number of possible extensions to the research conducted to strengthen the fitness function and utilize the created framework for content adaptation.

8.2.1 Enhancing the learning component

The results outlined in Chapter seven demonstrate that it is possible to generate a course adapting content to the individual cognitive traits and pedagogic preference of a learner. However, the correlation between the MELE and the actual outcome is 0.27. If more cognitive traits were incorporated into the evolutionary process it could increase the correlation between the MELE and the actual outcome. In particular, the associate learning skill of a learner should be included to allow adaptation to the domain content and previous learning experience. To enable this level of adaptation each learning experience should be documented and saved. Over time the system would be able to incorporate the associate learning skill of a learner as a metric within the evolutionary component and using context sensitive metadata describing the domain content, identify suitable content adapting to previously discussed cognitive metrics within the instructional content. The modular design of the learning component would enable easy integration of multiple cognitive traits as metrics for the evolutionary component. Additionally, an investigation of suitable motivation strategies should be carried out to ensure that the learner using the learning component are engaged with the instructional content for the duration of the learning experience without impacting on the potential learning experience.

8.2.2 Utilizing SCORM to create a rich client experience

The content produced by the learning component is SCORM compliant. Creating a rich interactive client side experience tracked using the SCORM data model elements would enable the automatic monitoring and adaptation protocol to change the current environment and instructional content depending on the learners interactions. This type of protocol would enhance the engagement of a learner producing suitable interference when appropriate.

8.2.3 Avoiding black box problems

The content produced is based on reconstructing content from suitable repositories. It is clearly evident that when the instructional space is small the learning component evolves at a slower evolution rate than when the learning space is large. This is due to the inconsistencies of the instructional space and the limited mutations available to the learning component to break out of a local minimum. Increasing the instructional space would avoid this problem. The instructional space could be increased by collaborating with multiple institutions or creating a web crawler to identify suitable content. However, as discussed by Norm Freissen [2] there exists huge inconsistencies in metadata production for instructional content. An additional component should be included to allow learners to communicate with the learning component and identify inconsistencies found within instructional content.

8.2.4 A Flexible Framework for fine tuning

The framework that was produced requires a suitable repository to allow the evolutionary strategy generate instructional content. It was recommended that the MELE be set at seventy percent at an Internal conference on Learning to allow the learner exceed beyond the MELE. This recommendation suited the evolution strategy associated with GA (i.e. GAs perform really fast for the initial evolution but require a significant amount of time in identifying the ideal solution, depending on the problem). The framework is a modular framework and is designed to evolve content to suit the metadata produced by the content analyser. The analyser could easily be extended to generate suitable metadata associated any trait that was identified by an author. The fitness function within the GA would need to be updated to reflect this modification. The specification allows an author to define the MELE and also the weights associated with each trait. The weights give the different traits higher / lower importance throughout the evolutionary process. An author of a specification could make modifications to these weights and reduce the importance of a trait that the author wanted the learner to get some experience with. For example, if a learner was identified as having low working memory capacity, the model could generate a course to suit a learner with higher working memory and train the learner to cope with a large instructional space.

8.2.5 Turing test validation

The framework that was created allows for multiple pedagogic metrics to be identified within instructional content to create suitable granular learning object repositories. These repositories are then harvested to create instructional content. The repositories from the evaluation process for the framework were created from instructional authors participating in the process, however if a suitable search strategy included crawling the web and other learning object repositories prior to processing the data through the Content Analyser there would exist large amounts of elements with suitable metadata. Increasing the number of objects within the repository enhances the potential for the evolutionary strategy. A Turing test could be used to identify whether a course constructed from the framework covered the learning outcomes, which would be validated by content experts. This process could move to producing a flexible framework for on-demand content generation suited to the cognitive ability and pedagogic preference of any learner.

Appendix A

UML Use-Case Module Descriptor

Aims and Objectives

- To introduce the fundamental theory and elements involved in UML Use Case design.
- To develop functional requirements for a given task and to identify the actors associated with the functional requirements.

Learning Outcomes

On completion of this module, Students will:

- LO1: Have gained a specialised knowledge of the elements involved with a UML 2.0 Use Case diagram.
- LO2: Understand the principals involved in creating UML Use Case models and the actors associate with the models.

Content

1. **UML Use-Case**
    - Why use UML Use-Case diagrams
    - Use Case
    - Use Case diagram
2. **Actors**
    - Primary actors
    - Secondary actors
    - Time actor
    - <<systems>>actor
3. **Functional Requirements**

- Requirements definitions

4. **Relationships**

    - Between Use Cases
    - Extends
    - Includes
    - Binary association between an actor and a Use Case

5. **Generalisation**

    - Between actors
    - Between Use Cases

Teaching and Learning Methods

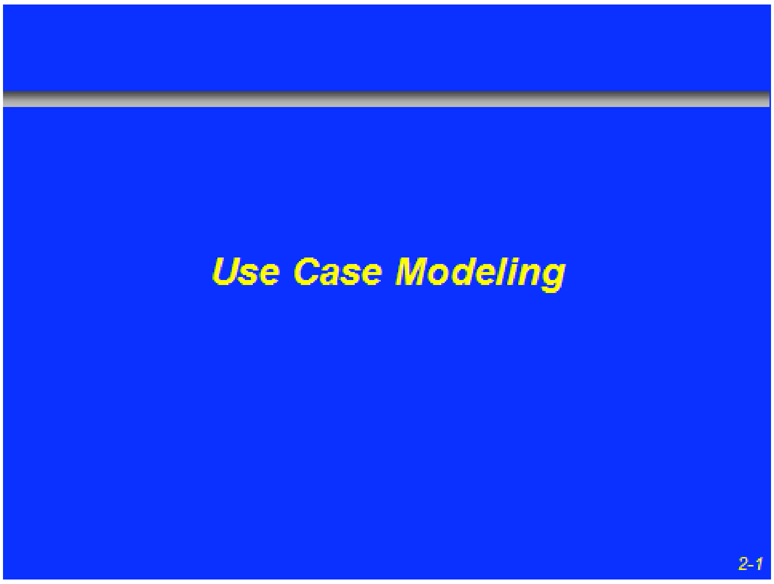
Twenty minute lecture or using learning component

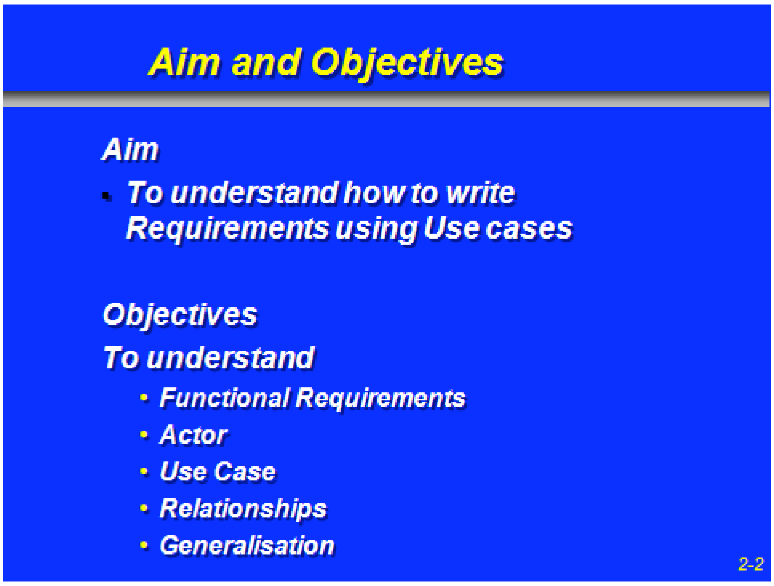
Assessments and Marking Schemes

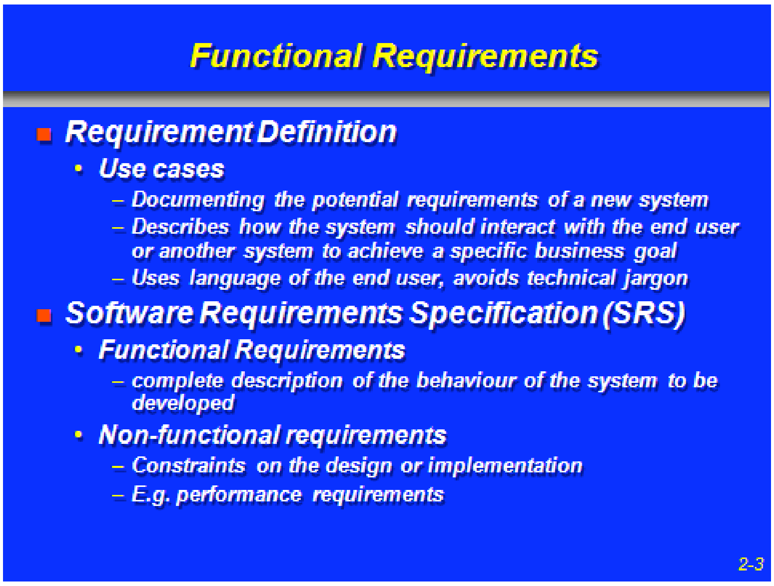
Terminal Examination 100 %

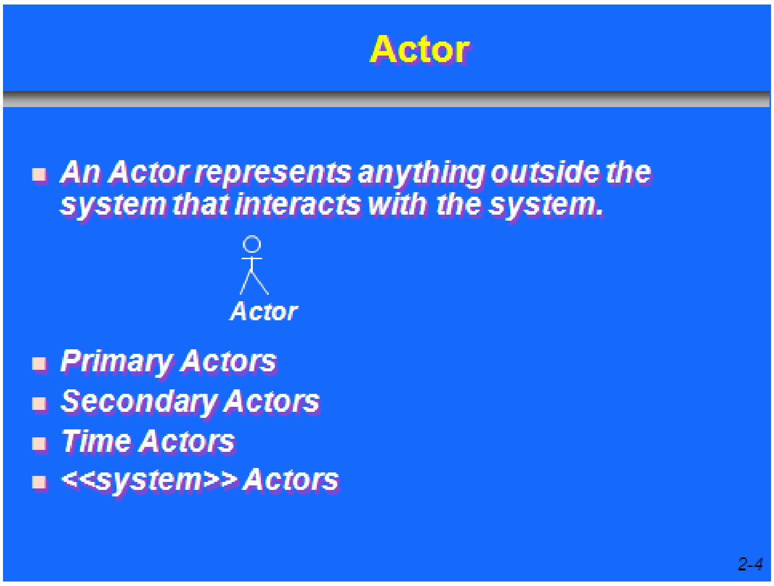
Appendix B

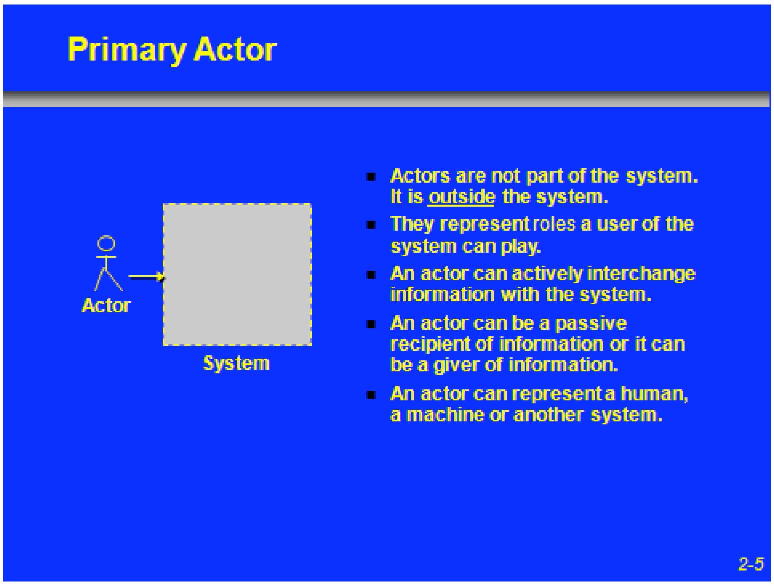
Traditional Slides for UML Course

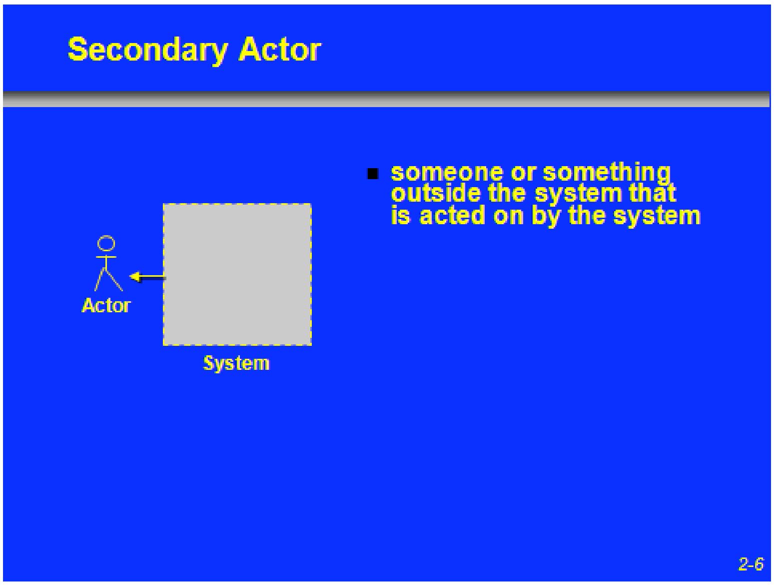

Use Case Modeling
2-1


Aim and Objectives
Aim
To understand how to write Requirements using Use cases
Objectives
To understand
• Functional Requirements
• Actor
• Use Case
• Relationships
• Generalisation
2-2


Functional Requirements
Requirement Definition
Use cases
– Documenting the potential requirements of a new system
– Describes how the system should interact with the end user or another system to achieve a specific business goal
– Uses language of the end user, avoids technical jargon
Software Requirements Specification (SRS)
Functional Requirements
– complete description of the behaviour of the system to be developed
Non-functional requirements
– Constraints on the design or implementation
– E.g. performance requirements
2-3

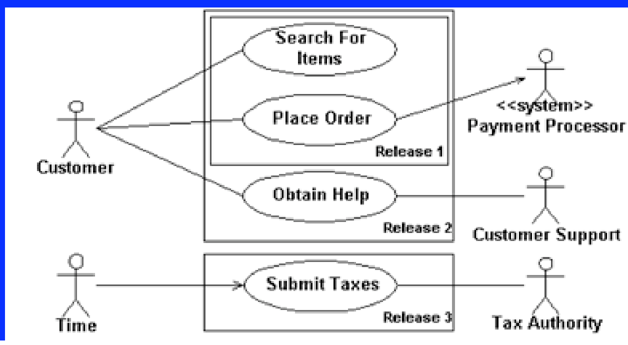

Actor
An Actor represents anything outside the system that interacts with the system.
Actor
Primary Actors
Secondary Actors
Time Actors
<<system>> Actors
2-4


Primary Actor
Actor
System
Actors are not part of the system. It is outside the system.
They represent roles a user of the system can play.
An actor can actively interchange information with the system.
An actor can be a passive recipient of information or it can be a giver of information.
An actor can represent a human, a machine or another system.
2-5


Secondary Actor
Actor
System
someone or something outside the system that is acted on by the system
2-6

Time Actor and <<systems>> actor

- For the online shopping introduce an Actor Called "Time" to Initiate Scheduled Events.
- Use <<system>> to indicate System Actors

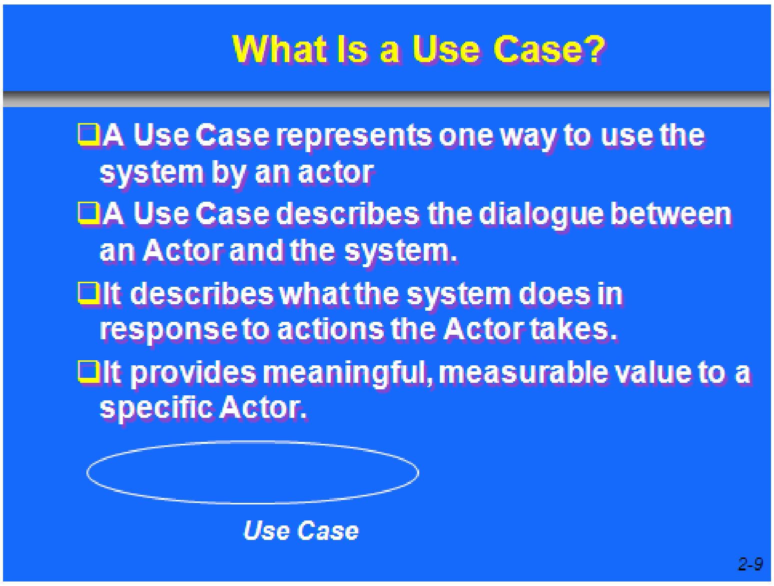


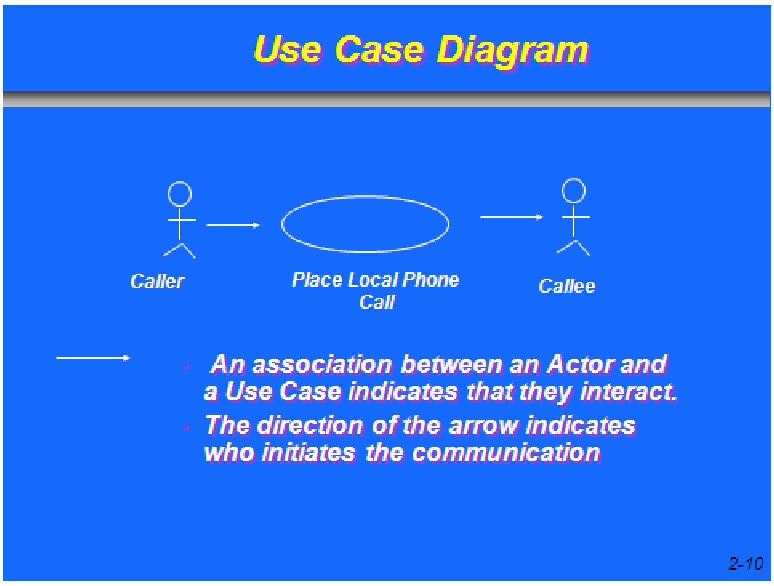
2-7

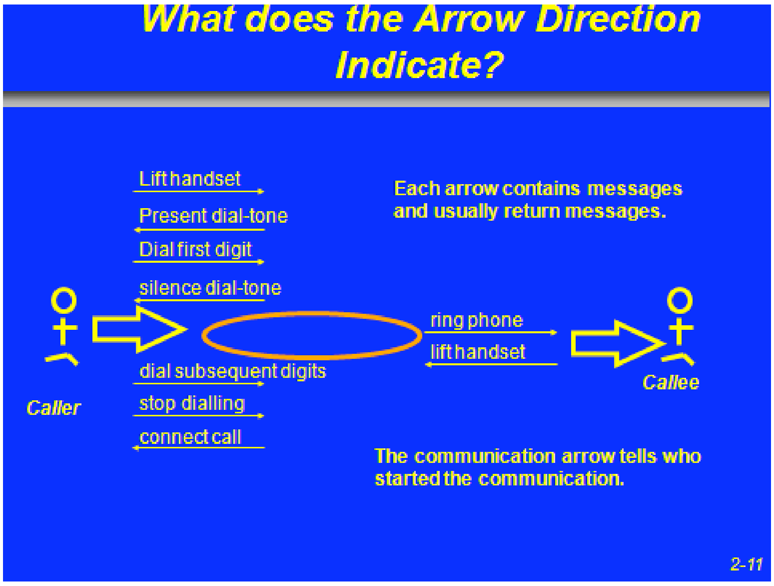
Use Cases – Why?

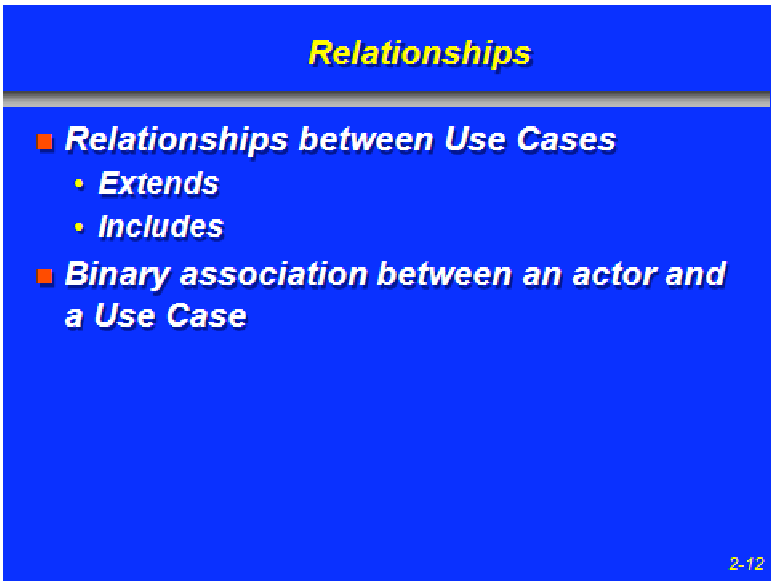
- Use cases diagrams are used to:
  - document the business process
  - identify possible collaborative business areas
  - separate business processes into functional system areas
  - serve as requirements documentation for system development, as they are defined in a non-implementation/easy-to-read manner.
  - categorize requirements (e.g., state of implementation, shipyard, functional system, etc.)
  - rank requirements (e.g., level of importance, risk, level of interoperability, etc.)
  - identify the effects of functional changes on implementation systems, or implementation changes on functional capabilities

2-8


What Is a Use Case?
A Use Case represents one way to use the system by an actor
A Use Case describes the dialogue between an Actor and the system.
It describes what the system does in response to actions the Actor takes.
It provides meaningful, measurable value to a specific Actor.
Use Case
2-9

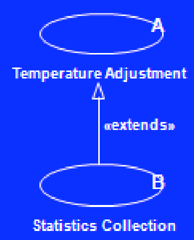

Use Case Diagram
Caller
Place Local Phone Call
Callee
An association between an Actor and a Use Case indicates that they interact.
The direction of the arrow indicates who initiates the communication
2-10


What does the Arrow Direction Indicate?
Lift handset
Present dial-tone
Dial first digit
silence dial-tone
dial subsequent digits
stop dialling
connect call
Caller
ring phone
lift handset
Callee
Each arrow contains messages and usually return messages.
The communication arrow tells who started the communication.
2-11


Relationships
Relationships between Use Cases
Extends
Includes
Binary association between an actor and a Use Case
2-12

Why use the "Extends" Relationship ?

- To model optional behavior in a separate use case
- To simplify a complex flow of events
- To model behavior that is to be developed separately, possibly in late iterations
- To model behavior that require different post conditions. (optional)



2-13

The Extends Relationship

- The extended use case A must be complete:
  - Does not know of the extending use case B, that is, it is A is independent of B
  - When instantiated, the instance may perform the behavior described for B as well as that of A
- The relationship:
  - Has a probe that tells where in A to extend
  - Has a condition that tells when to extend
  - The direction goes from the extending use case B, to the extended use cases A
- The extending use case, B, knows which use case to extend

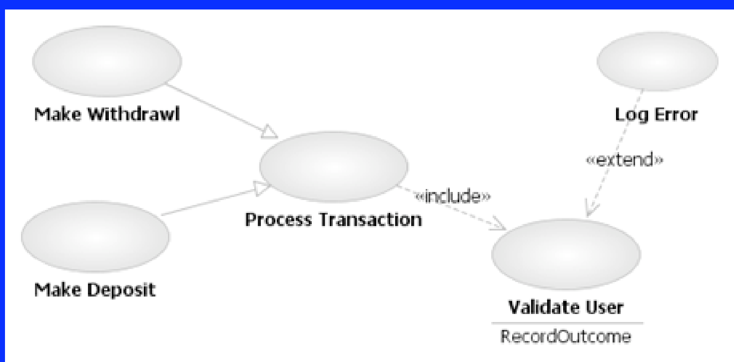
2-14

Includes

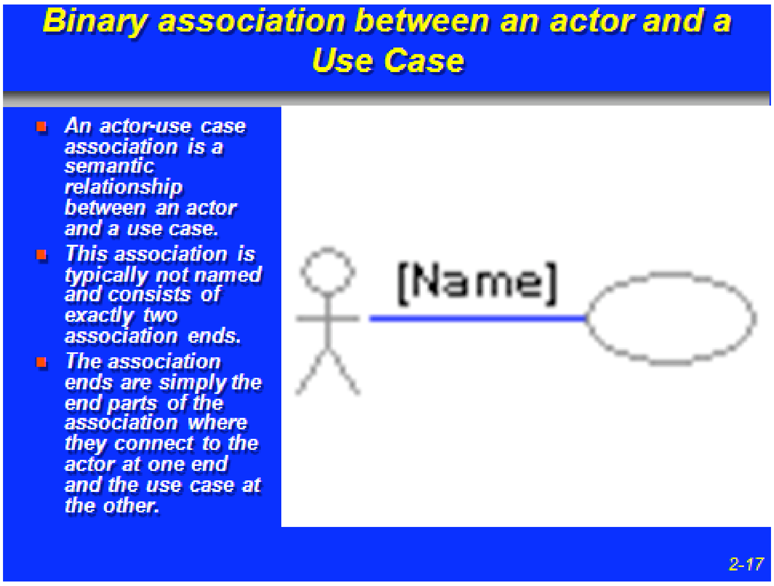
- *A given use case may include another.*
- *"Include is a Directed Relationship between two use cases, implying that the behaviour of the included use case is inserted into the behaviour of the including use case".*

2-15

Track Order

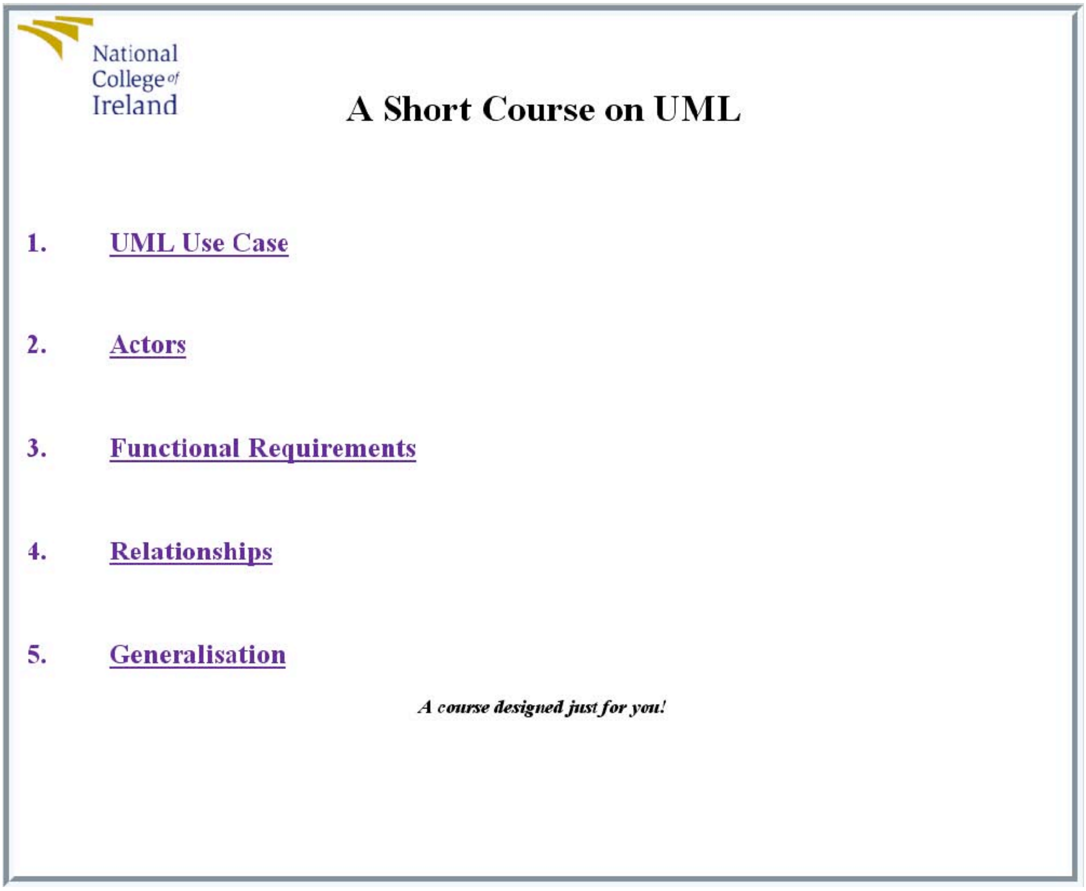


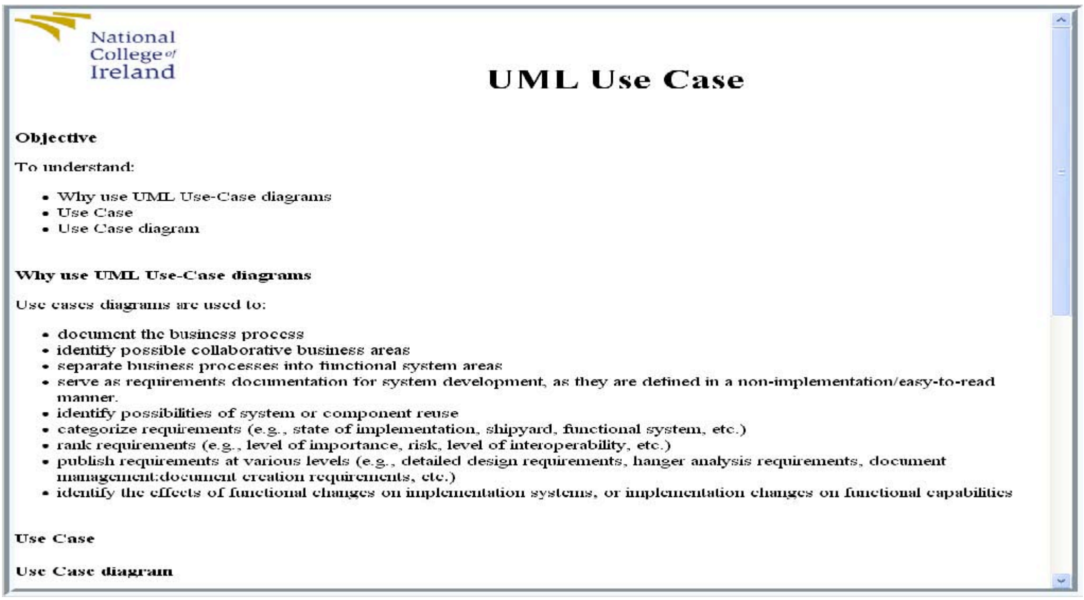
2-16

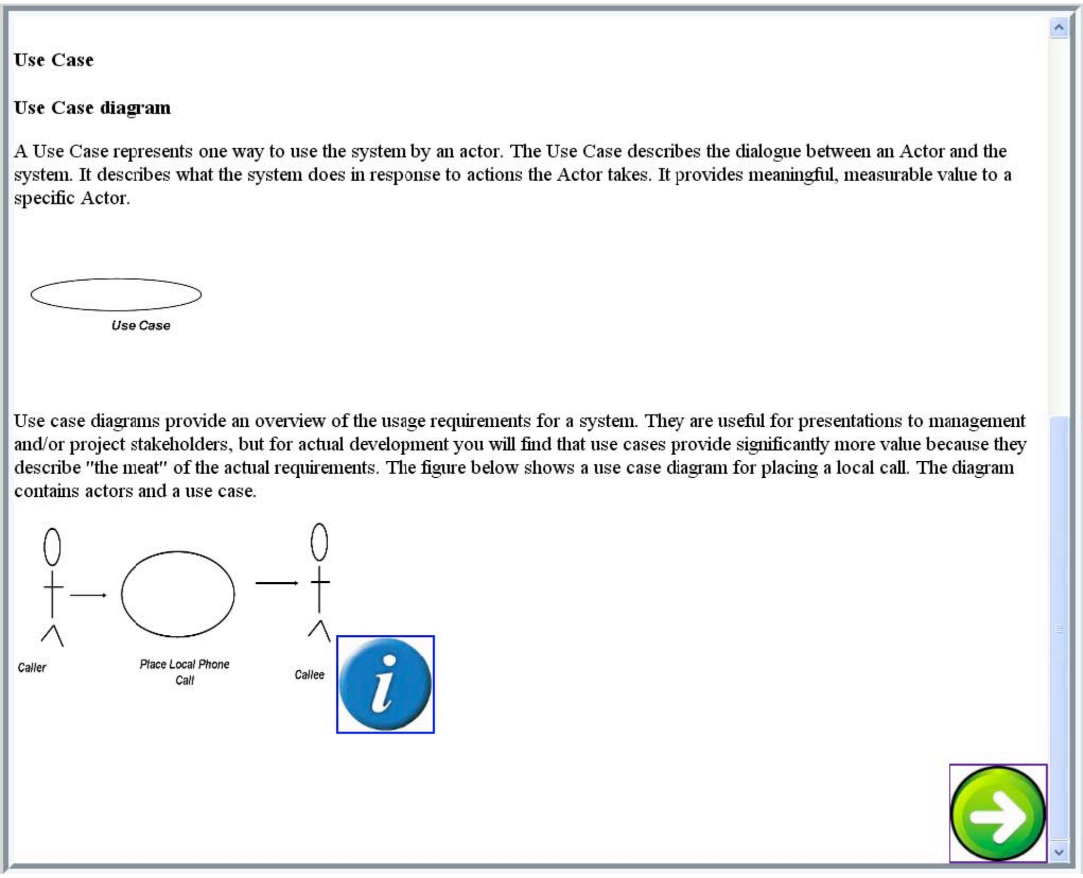

Binary association between an actor and a Use Case
An actor-use case association is a semantic relationship between an actor and a use case.
This association is typically not named and consists of exactly two association ends.
The association ends are simply the end parts of the association where they connect to the actor at one end and the use case at the other.
[Name]
2-17

Appendix C

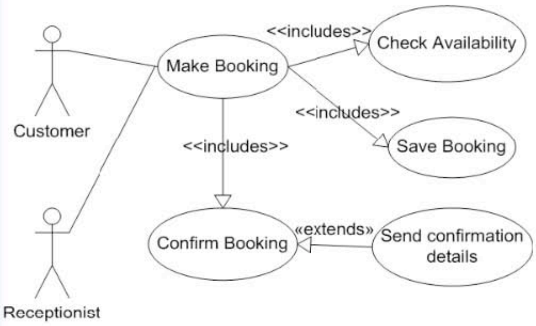
Course for learner with low WMC


National College of Ireland
A Short Course on UML
1. UML Use Case
2. Actors
3. Functional Requirements
4. Relationships
5. Generalisation
A course designed just for you!


National College of Ireland
UML Use Case
Objective
To understand:
• Why use UML Use-Case diagrams
• Use Case
• Use Case diagram
Why use UML Use-Case diagrams
Use cases diagrams are used to:
• document the business process
• identify possible collaborative business areas
• separate business processes into functional system areas
• serve as requirements documentation for system development, as they are defined in a non-implementation/easy-to-read manner.
• identify possibilities of system or component reuse
• categorize requirements (e.g., state of implementation, shipyard, functional system, etc.)
• rank requirements (e.g., level of importance, risk, level of interoperability, etc.)
• publish requirements at various levels (e.g., detailed design requirements, hanger analysis requirements, document management:document creation requirements, etc.)
• identify the effects of functional changes on implementation systems, or implementation changes on functional capabilities
Use Case
Use Case diagram


Use Case
Use Case diagram
A Use Case represents one way to use the system by an actor. The Use Case describes the dialogue between an Actor and the system. It describes what the system does in response to actions the Actor takes. It provides meaningful, measurable value to a specific Actor.
Use Case
Use case diagrams provide an overview of the usage requirements for a system. They are useful for presentations to management and/or project stakeholders, but for actual development you will find that use cases provide significantly more value because they describe "the meat" of the actual requirements. The figure below shows a use case diagram for placing a local call. The diagram contains actors and a use case.
Caller
Place Local Phone Call
Callee

National College of Ireland



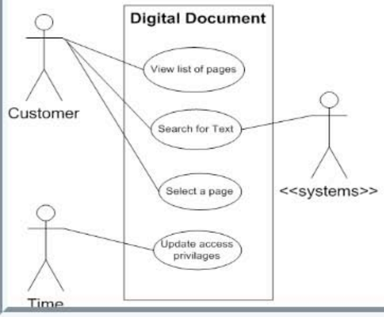
The above figure gives a diagrammatical representation of a Use Case diagram for an online hotel reservation system. The diagram shows both the receptionist and the customer following the same process of making a booking, with the system then checking availability and confirming the booking. In the initial stages of the requirements capture process, this would be sufficient detail. As the overall development continues, then each of the Use Cases in this diagram (for example "Send Confirmation Details") would be documented in more detail.

Actors

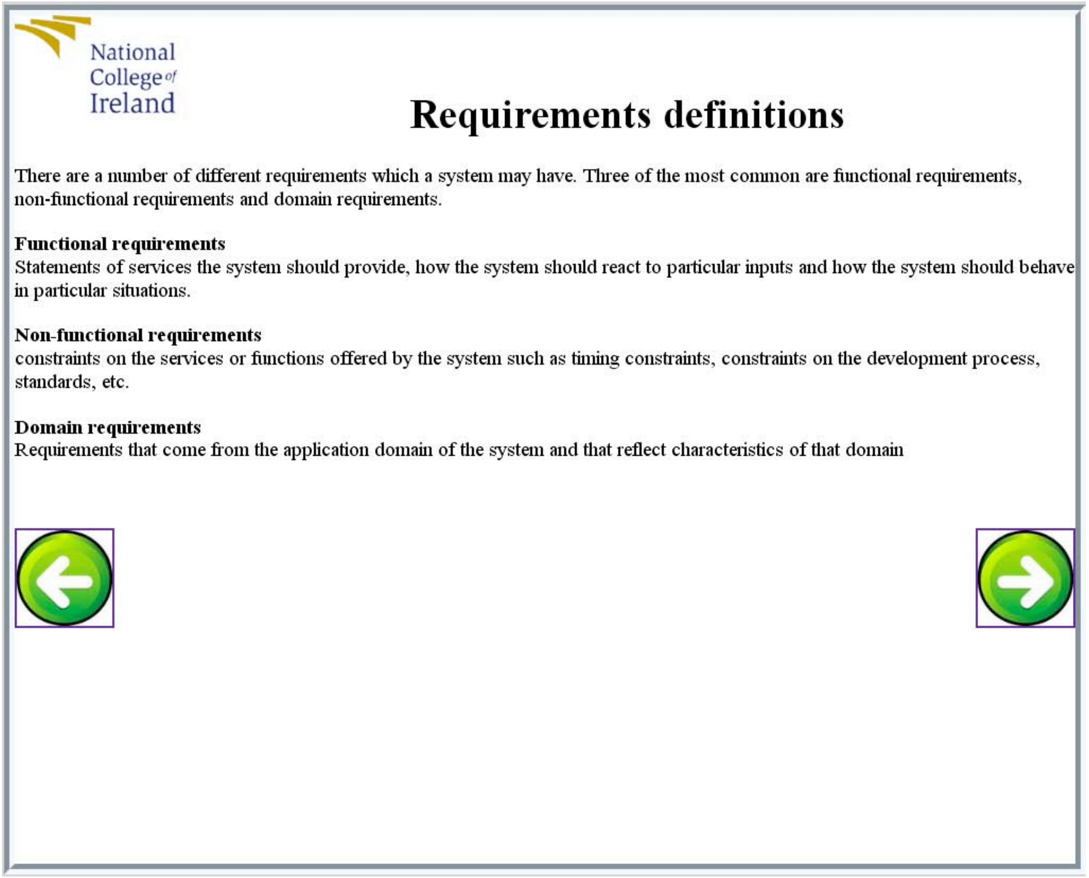
In UML Design Actors are people or processes that interact with your system. Actors are classified as: *Primary actor*, *Secondary actor*, *Time actor* and a *Systems actor*.

- *Primary actor*: represents the principal individual interacting with your system. Primary actors are primarily located at the top left corner of a UML Use Case diagram (in western cultures people read from left to right, top to bottom).
- *Secondary actor*: this actor is used to represent another person interacting with the system. Secondary actors would typically be involved with the operations of the system.
- *Time actor*: the Time actor is used to reflect a process that occurs periodically.
- *Systems actor*: represents an automated response to a customer driving interaction.

The figure above gives an example of an actor. The actor is drawn as a stick man with a name associated with the actors roll.



The figure above gives an example of a digitised document system. The main actor interacting with the system is a customer. The customer can view the list of all the pages, Search pages for specified text and select a page. The Systems actor uses data mining strategies to locate the desired information. The Time actor periodically updates the access privileges on each account.


National College of Ireland
Requirements definitions
There are a number of different requirements which a system may have. Three of the most common are functional requirements, non-functional requirements and domain requirements.
Functional requirements
Statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations.
Non-functional requirements
constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, etc.
Domain requirements
Requirements that come from the application domain of the system and that reflect characteristics of that domain

National College of Ireland

Relationships

A relationship is a connection between elements. Relationships are used to add structure and behaviour to the elements within the system.

Relationships Between Use Cases

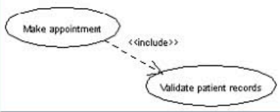
The *include* relationship is used when several use cases need to complete the same sequence of actions. This relationships allows the incorporation of reusability and consistent behaviour into the system. When a use case is depicted as using the functionality of another use case in a diagram, this relationship between the use cases is named as an include relationship. Literally speaking, in an include relationship, a use case includes the functionality described in the other use case as a part of its business process flow. An include relationship is depicted with a directed arrow having a dotted shaft. The tip of the arrowhead points to the parent use case and the child use case is connected at the base of the arrow.

Above is a representation of the include relationship.

The *extend* relationship is seldom used. In an extend relationship between two use cases, the child use case adds to the existing functionality and characteristics of the parent use case. An extend relationship is depicted with a directed arrow having a dotted shaft, similar to the include relationship. The tip of the arrowhead points to the parent use case and the child use case is connected at the base of the arrow. This relationship is used to indicate a given use case may extend to another use case, usually indicates use cases that are optional to the base use case. This is generally used when a use case becomes complex and it is split out in order to keep things simple.

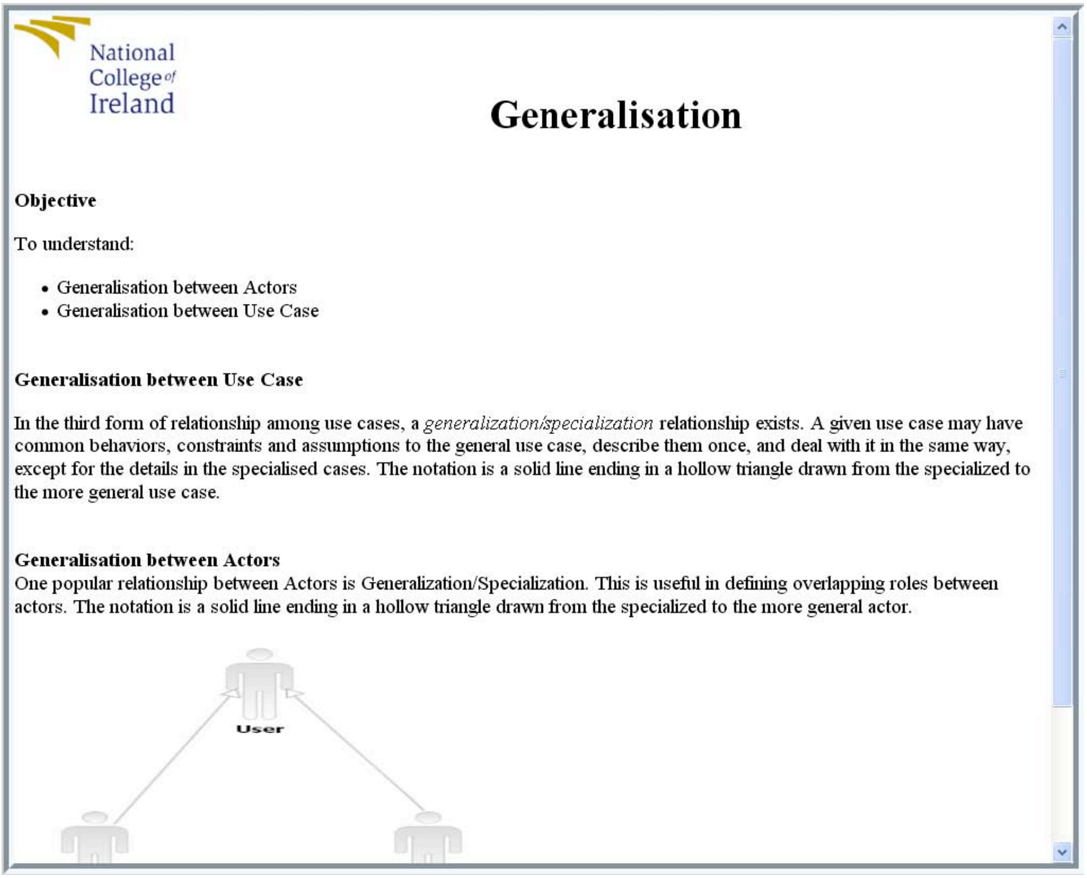
Binary association between an Actor and a Use Case

An *association* exists when an actor is involved in an interaction described by a use case. Binary associations are represented by a line with two ends. An example of a binary association is 'A department offers courses'



Above is a representation of the include relationship.

Above is a representation of the extend relationship.


National College of Ireland
Generalisation
Objective
To understand:
• Generalisation between Actors
• Generalisation between Use Case
Generalisation between Use Case
In the third form of relationship among use cases, a generalization/specialization relationship exists. A given use case may have common behaviors, constraints and assumptions to the general use case, describe them once, and deal with it in the same way, except for the details in the specialised cases. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general use case.
Generalisation between Actors
One popular relationship between Actors is Generalization/Specialization. This is useful in defining overlapping roles between actors. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general actor.
User


Customer
Administrator


National
College of
Ireland
Thank you for participating, please turn off your screen and start the quiz.
A course designed just for you!

Appendix D

Quiz and Marking Scheme

Answer all questions. Award 5 marks for each question.

1. What are UML Use Cases used for?

   Award marks for the learners understanding of a UML USE-Case. In particular, award marks for the learners ability to give suitable examples of where to use UML Use Case diagrams.

2. Differentiate between a Primary actor and a systems actor.

   Award marks for the learners understanding of the differences between a primary and a systems actor.

3. List two functional requirements of a library computer system?

   Award marks for each appropriate functional requirement that is listed.

4. What are the different relationships that exist between Use Cases?

   Award marks for the learners understanding of the different relationships that can exist between Use Cases.

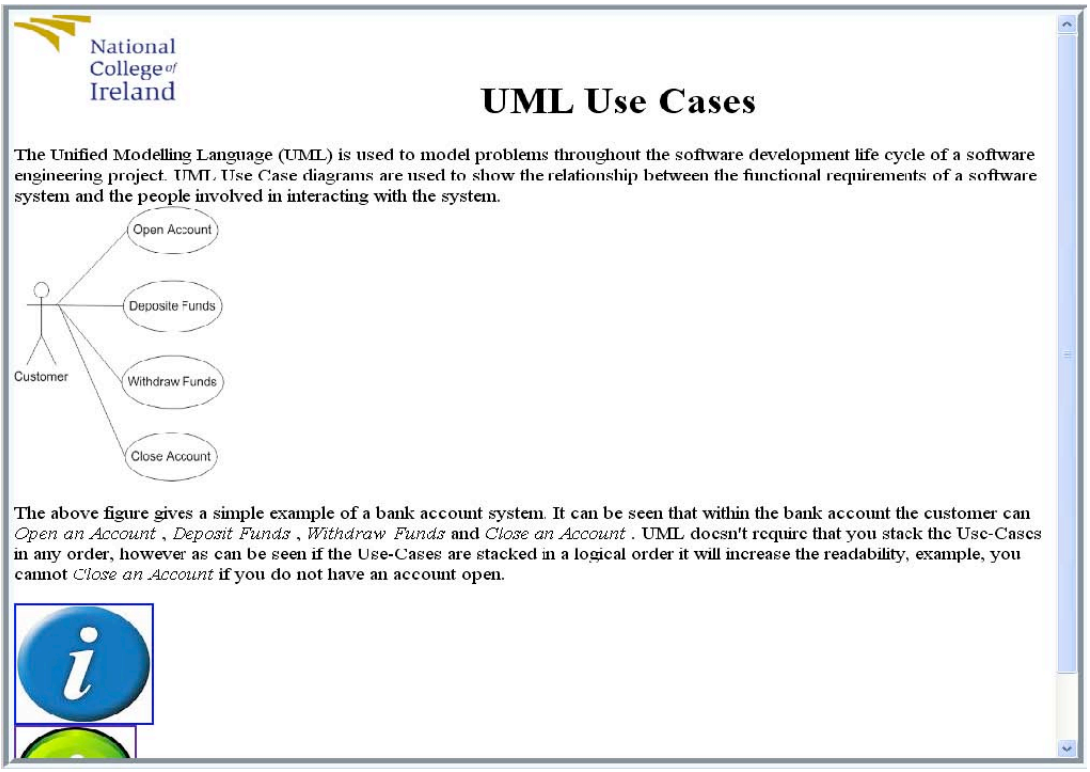
5. Give an example, using UML notation of a generalisation between two actors.

   This question is involved with understanding of using a generalisation between two actors. Award marks for a suitable example using UML notation illustrating a generalisation between two actors.

Appendix E

Course for learner with high WMC


National
College of
Ireland
A Short Course on UML
1. UML Use Case
2. Actors
3. Functional Requirements
4. Relationships
5. Generalisation
A course designed just for you!


National
College of
Ireland
UML Use Cases
The Unified Modelling Language (UML) is used to model problems throughout the software development life cycle of a software engineering project. UML Use Case diagrams are used to show the relationship between the functional requirements of a software system and the people involved in interacting with the system.
Open Account
Deposite Funds
Withdraw Funds
Close Account
Customer
The above figure gives a simple example of a bank account system. It can be seen that within the bank account the customer can Open an Account , Deposit Funds , Withdraw Funds and Close an Account . UML doesn't require that you stack the Use-Cases in any order, however as can be seen if the Use-Cases are stacked in a logical order it will increase the readability, example, you cannot Close an Account if you do not have an account open.

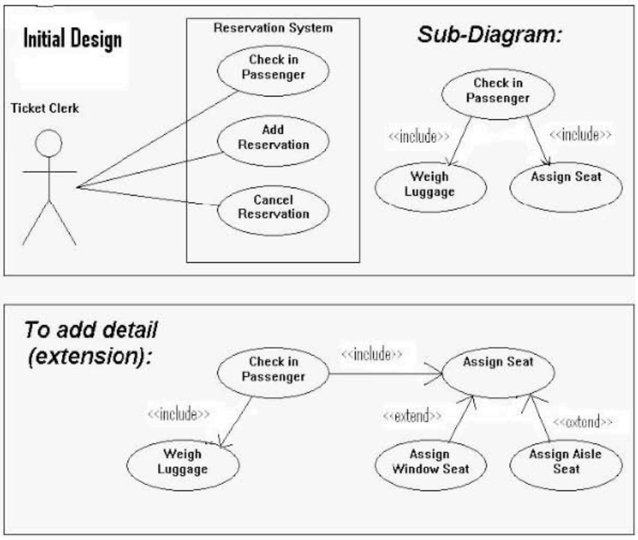
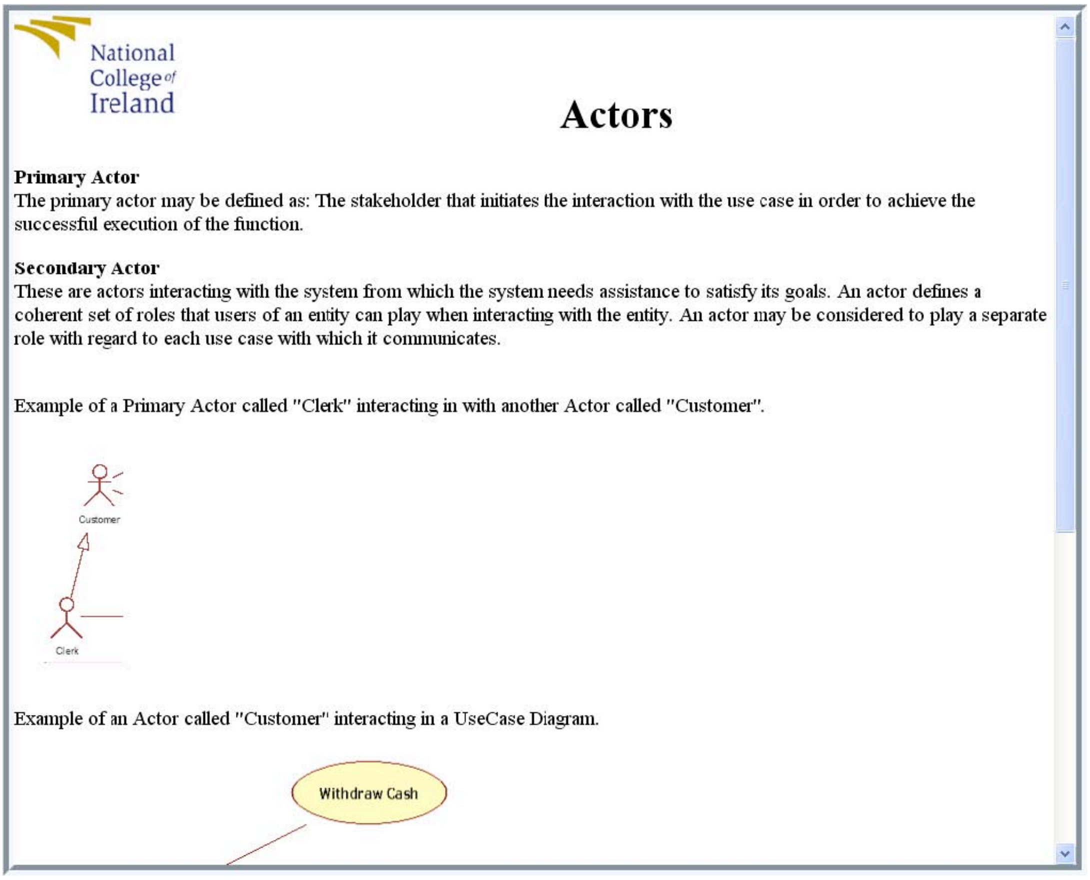
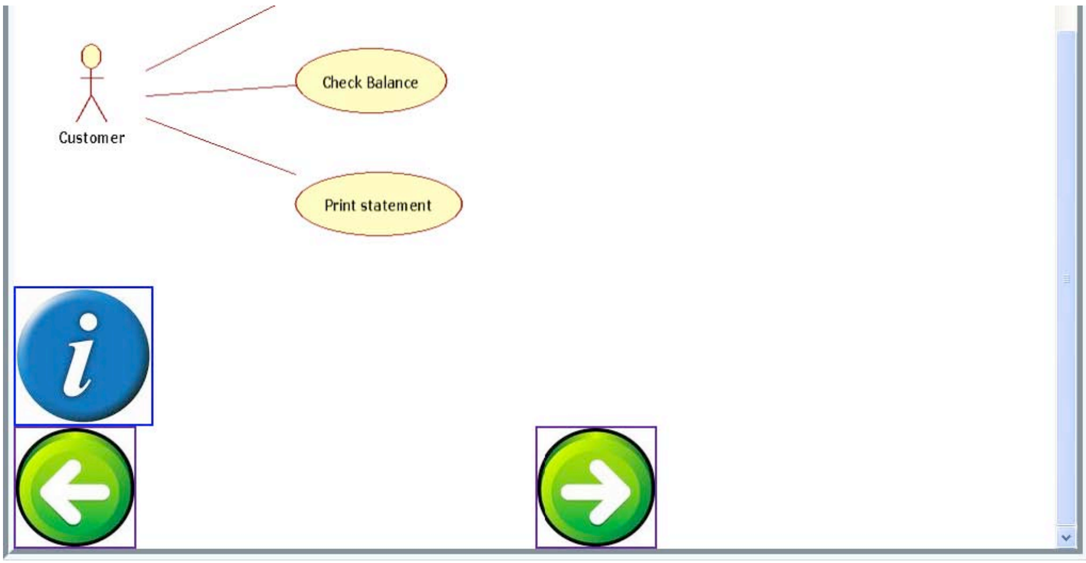


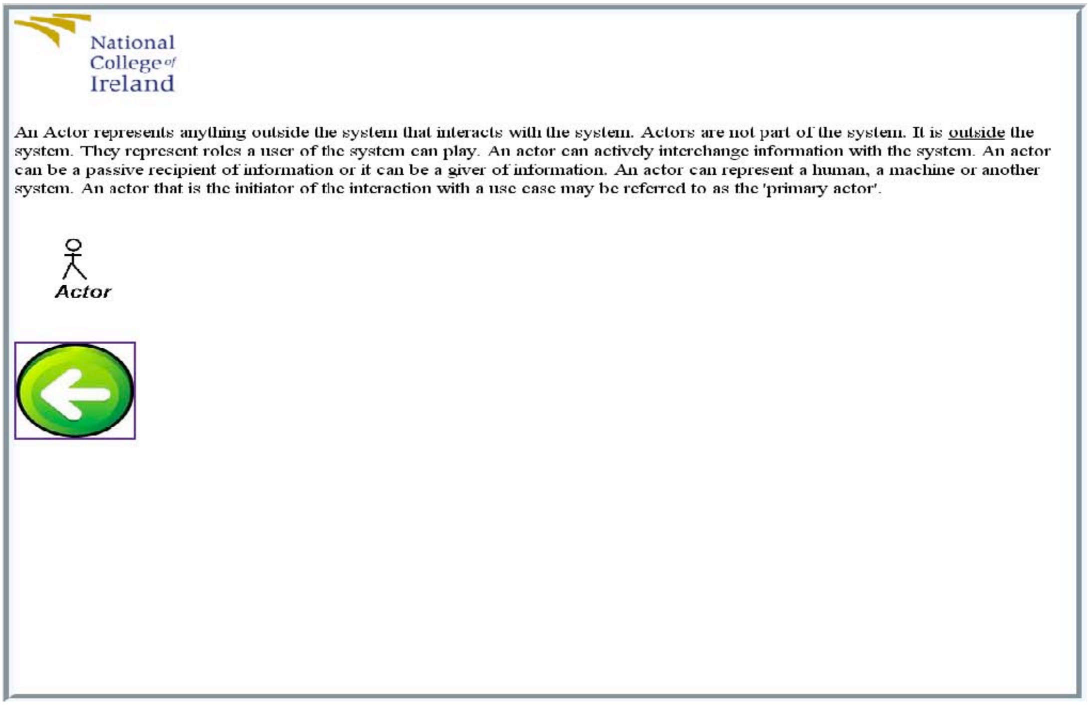
Figure 1. Example of a Use Case Disgram

Figure 1. shows a simple example of a Use Case Diagram. The include arrow is drawn from a use case X to another use case Y to indicate that the process of doing X always involves doing Y at least once (although it may involve doing it many times, "at least once" is the only relationship guaranteed by this symbol.) The extend arrow is drawn from a use case X to a use case Y to show that process X is a special case of the more general process Y. Suppose you wanted to add detail to the diagram shown above, representing an airline reservation system. Specifically, what you would like to show is that not all of the seats aboard the airplane are exactly alike (some window and some aisle seats), . But of course, they cannot just be given their preference right away,

because the seat they want might not be available. Therefore, the process of assigning a window seat involves checking for the availability of window seats, whereas the process of assigning an aisle seat involves checking for the availability of aisle seats. But even though these processes are different, they are quite similar in a number of other ways, so it doesn't make sense to ignore their similarities. Fortunately, UML lets us have both: we write that assigning these two types of seats are different processes, but they are similar in that both processes extend a common, more general process (assigning seats).


National College of Ireland
Actors
Primary Actor
The primary actor may be defined as: The stakeholder that initiates the interaction with the use case in order to achieve the successful execution of the function.
Secondary Actor
These are actors interacting with the system from which the system needs assistance to satisfy its goals. An actor defines a coherent set of roles that users of an entity can play when interacting with the entity. An actor may be considered to play a separate role with regard to each use case with which it communicates.
Example of a Primary Actor called "Clerk" interacting in with another Actor called "Customer".
Customer
Clerk
Example of an Actor called "Customer" interacting in a UseCase Diagram.
Withdraw Cash

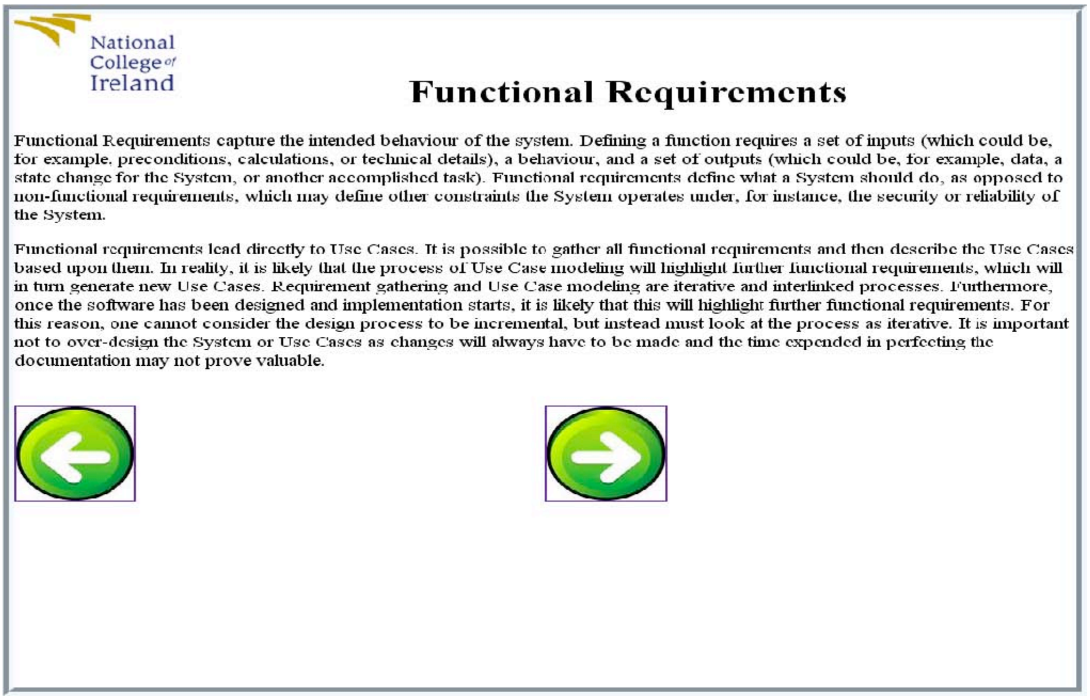

Customer
Check Balance
Print statement



National College of Ireland

An Actor represents anything outside the system that interacts with the system. Actors are not part of the system. It is outside the system. They represent roles a user of the system can play. An actor can actively interchange information with the system. An actor can be a passive recipient of information or it can be a giver of information. An actor can represent a human, a machine or another system. An actor that is the initiator of the interaction with a use case may be referred to as the 'primary actor'.

Actor

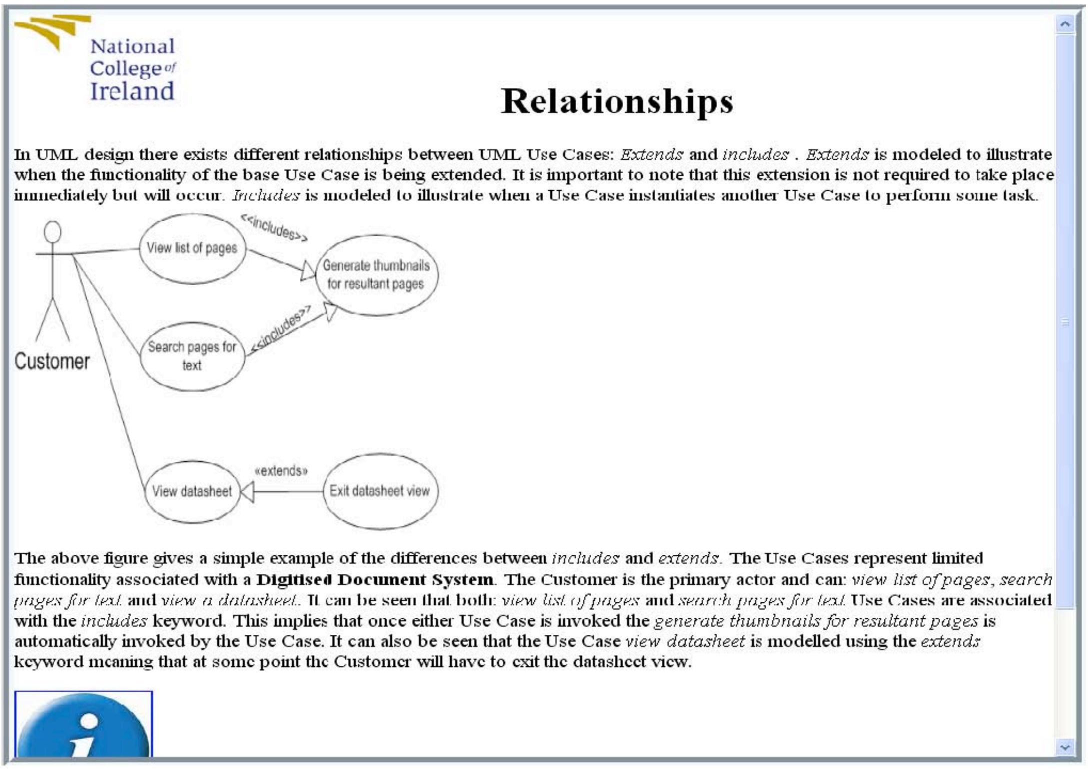


National College of Ireland

Functional Requirements

Functional Requirements capture the intended behaviour of the system. Defining a function requires a set of inputs (which could be, for example, preconditions, calculations, or technical details), a behaviour, and a set of outputs (which could be, for example, data, a state change for the System, or another accomplished task). Functional requirements define what a System should do, as opposed to non-functional requirements, which may define other constraints the System operates under, for instance, the security or reliability of the System.

Functional requirements lead directly to Use Cases. It is possible to gather all functional requirements and then describe the Use Cases based upon them. In reality, it is likely that the process of Use Case modeling will highlight further functional requirements, which will in turn generate new Use Cases. Requirement gathering and Use Case modeling are iterative and interlinked processes. Furthermore, once the software has been designed and implementation starts, it is likely that this will highlight further functional requirements. For this reason, one cannot consider the design process to be incremental, but instead must look at the process as iterative. It is important not to over-design the System or Use Cases as changes will always have to be made and the time expended in perfecting the documentation may not prove valuable.

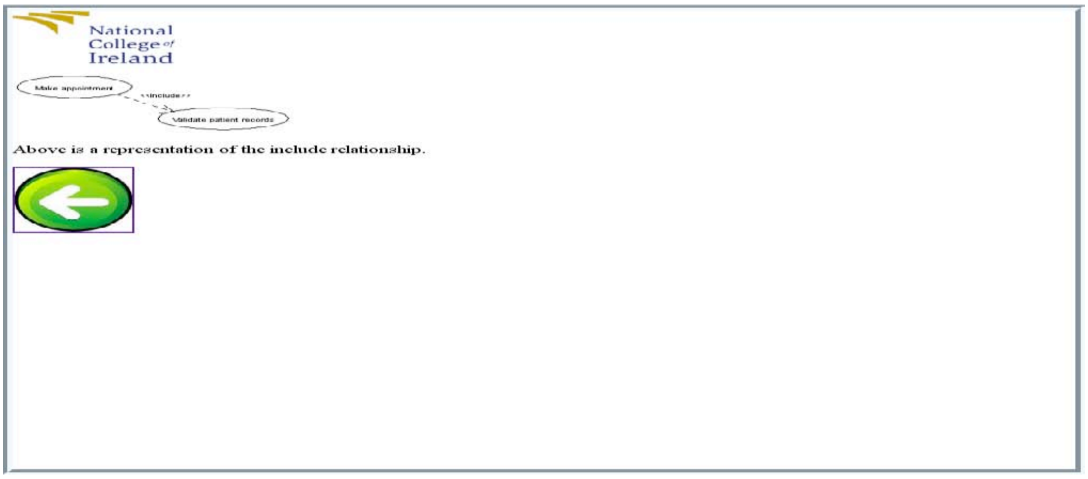


National College of Ireland

Relationships

In UML design there exists different relationships between UML Use Cases: *Extends* and *includes* . *Extends* is modeled to illustrate when the functionality of the base Use Case is being extended. It is important to note that this extension is not required to take place immediately but will occur. *Includes* is modeled to illustrate when a Use Case instantiates another Use Case to perform some task.

The above figure gives a simple example of the differences between *includes* and *extends*. The Use Cases represent limited functionality associated with a **Digitised Document System**. The Customer is the primary actor and can: *view list of pages*, *search pages for text* and *view a datasheet*. It can be seen that both: *view list of pages* and *search pages for text* Use Cases are associated with the *includes* keyword. This implies that once either Use Case is invoked the *generate thumbnails for resultant pages* is automatically invoked by the Use Case. It can also be seen that the Use Case *view datasheet* is modelled using the *extends* keyword meaning that at some point the Customer will have to exit the datasheet view.



National College of Ireland

Above is a representation of the include relationship.

Generalisation

We have already seen that there are several types of relationships that may appear on a use case diagram: An association between an actor and a use case, and An association between two use cases. These associations are depicted with lines connecting two modeling elements with an optional open-headed arrowhead on one end of the line. However if we use a closed arrow head then the relationship is called a generalisation. We say that Generalisations are depicted as a close-headed arrow with the arrow pointing towards the more general modeling element.

Figure 4. Use Case Relationships with Generalisation

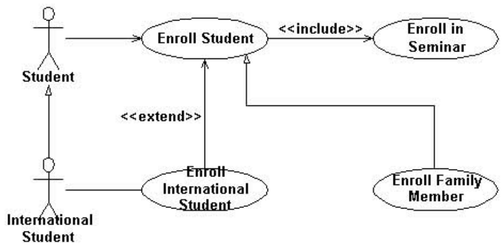
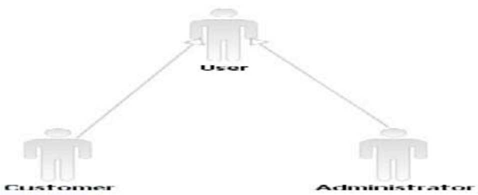
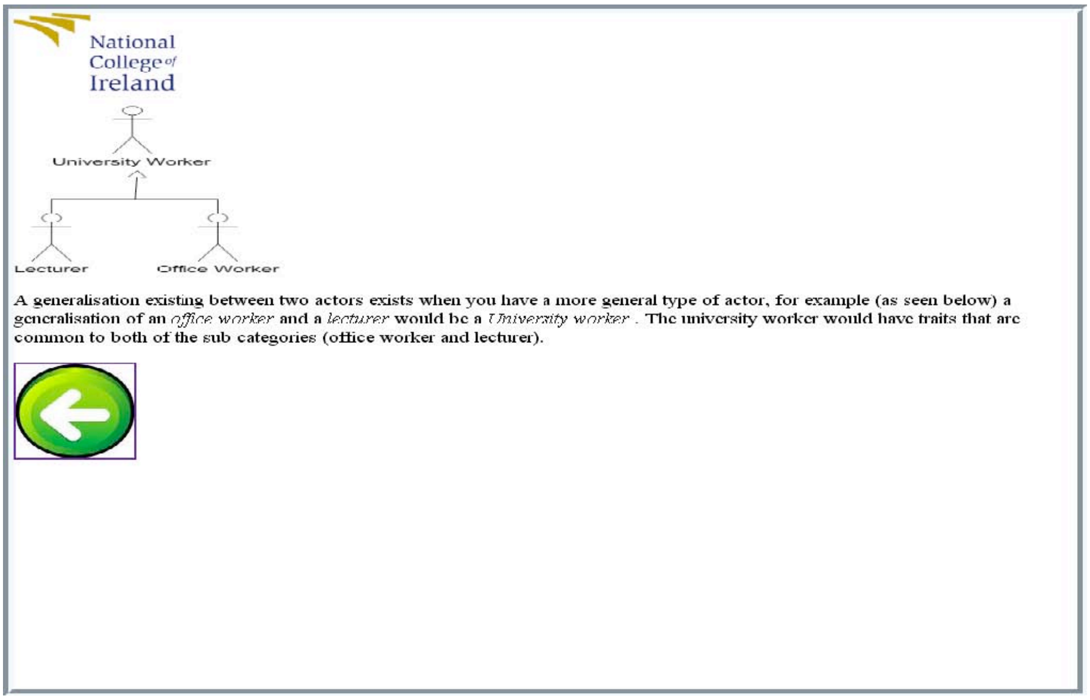


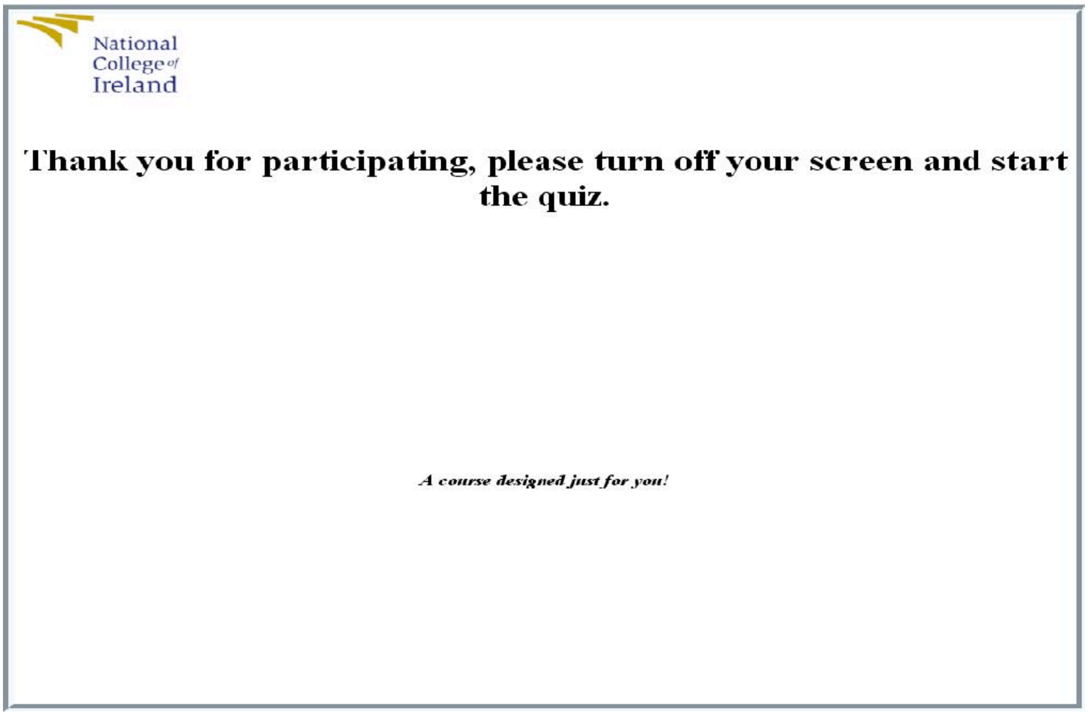
Figure 4. Use Case Relationships with Generalisation

Between actors
Apply the “Is Like” Rule to Use Case Generalisation (see Figure 4). Here we see that the Use Case Enroll Family Member “Is Like” Enroll Student.
Between Use Cases
Apply the “Is Like” Rule to Actor Generalisation (see Figure 4). In this case an International Student “Is Like” a Student.

Generalisation between Actors
One popular relationship between Actors is Generalization/Specialization. This is useful in defining overlapping roles between actors. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general actor.




National College of Ireland
University Worker
Lecturer
Office Worker
A generalisation existing between two actors exists when you have a more general type of actor, for example (as seen below) a generalisation of an *office worker* and a *lecturer* would be a *University worker* . The university worker would have traits that are common to both of the sub categories (office worker and lecturer).


National College of Ireland
Thank you for participating, please turn off your screen and start the quiz.
A course designed just for you!

Cited References

[1] N. Flemming, "Vark." Available Online, viewed March 20th 2010. `"http://www.vark-learn.com/english/index.asp"`.

[2] N. Freisen, A. Roberts, and S. Fisher, "Cancore: Metadata for learning objects," *Canadian Journal of Learning and Technology*, vol. 28, pp. 43–53, autumn 2002.

[3] G. Miller, "The magical number seven plus or minus two: Some limits on our capacity for processing information.," *Psychological Review*, vol. 63, issue 2, pp. 81 – 97, 1956.

[4] A. D. Baddely and G. J. Hitch, "Working memory.," *The psychology of learning and motivation: advances in research and theory*, vol. 8, pp. 47 – 89, 1974.

[5] N. Cowan, "Working memory capacity.," *New York NY: Psychology Press*, 2005.

[6] A. Ericsson and W. Kintsch, "Long-term working memory.," *Psychological Review*, vol. 102, issue 2, pp. 211 – 245, 1995.

[7] P. Brusilovsky, "Developing adaptive educational hypermedia systems: From design models to authoring tools," *Authoring tools for Advanced Technology Learning Environments*, pp. 40 – 50, 2003.

[8] A. Patel, Kinshuk, and D. Russell, "Intelligent tutoring tools for cognitive skill acquisition in life long learning," *Educational Technology and Society*, vol. 3, issue 1, pp. 107 – 113, 2000.

[9] V. Dijk and T. A. Kintsch, "Strategies of discourse comprehension.," *New York: Academic press*, 1983.

[10] N. Cowan, "Evolving conceptions of memory storage, selective attention, and their mutual constraints within the human information processing system," *Psychological Bulletin*, vol. 104, issue 2, pp. 163 – 191, 1988.

[11] Kintsch, A. Patel, and A. Ericsson, "The role of long-term working memory in text comprehension.," *Psychologia*, vol. 42, pp. 186 – 198, 1999.

[12] A. Baddeley, "The episodic buffer: a new component of working memory?," *Trends in Cognitive Sciences*, vol. 4, issue 11, pp. 417 – 423, 2000.

[13] D. Laurillard, *Rethinking University Teaching a framework for the effective use of learning technologies.* London: Published by RoutledgeFalmer, 2002.

[14] A. Guida and H. Tardieu, "Is personalisation a way to operationalise long-term working memory?," *Current Psychology Letters: Behaviour Brain & Cognition*, vol. 1, issue 15, pp. 1 – 15, 2005.

[15] A. M.Owen, K. M. McMillan, A. R. Laird, and E. Bullmore, "N-back working memory paradigm: A meta-analysis of normative functional neuroimaging studies.," *Human Brain Mapping*, vol. 25, pp. 46 – 59, 2005.

[16] J. L. Alty, "Multimedia: What is it and how do we exploit it?," *People and Computers*, vol. 6, pp. 31 – 44, 1991.

[17] P. Brusilovsky, E. Eklund, and E. Schwarz, "Web-based education for all: A tool for development adaptive courseware," *Proceedings of the Seventh International World Wide Web Conference, Computer Networks and ISDN Systems*, vol. 30, pp. 291 – 300, 1998.

[18] A. Patel and Kinshuk, "Discipline attributes and teaching style: Environment contexts of an its design for determining multimedia and virtual reality representations," *Knowledge Transfer*, vol. 1, issue 1, pp. 107 – 113, 1998.

[19] P. D. Bra and L. Calvi, "Aha: a generic adaptive hypermedia system," *Proceedings of the 2nd workshop on Adaptive Hypertext and Hypermedia Hypertext*, pp. 1–10, 1998.

[20] P. Brusilovsky and J. Anderson, "Act-r electronic bookshelf: An adaptive system to support learning act-r on the web," *Proceedings of the 3rd world conference of the WWW Internet and Intranet WebNet98*, vol. 3, pp. 92 – 97, 1998.

[21] A. Patel, D. Russell, and Kinshuk, "Computer assisted assessment in higher education," *Kogan Page*, vol. London, pp. 139 – 147, 2000.

[22] P. Brusilovsky, "Concept-based courseware engineering for large scale web-based education," *Proceedings of WebNet2000, World Conference of the WWW and Internet*, pp. 69 – 74, 2000.

[23] K. VanLehn, "Cognitive skill acquisition," *Annual review of Psychology*, vol. 47.

[24] Kinshuk, R. Oppermann, A. Patel, and A. Kashihara, "Multiple representation approach in multimedia based intelligent educational systems," *Artificial Intelligence in Education*, vol. IOS Press, pp. 259 – 266, 1999.

[25] A. Patel, Kinshuk, and D. Russell., "Implementing cognitive apprenticeship and conversation theory in interactive web-based learning systems.," *Proceedings of Sixth Multi-Conference on Systemic, Cybernetics and Informatics*, pp. 523 – 528, 2002.

[26] G. Pask, "Conversation theory: applications in education and epistemology," *Elsevier*, 1976.

[27] A. Collins, J. S. Brown, and S. E. Newman, "Cognitive apprenticeship: Teaching the craft of reading, writing and mathematics!," *Knowing, learning and Instruction: Essays in Honor of Robert Glaser Hillsdale*, 1989.

[28] H. Sally, Kinshuk, H. Hong, and A. Patel, "Granular approach to adaptivity in problem-based learning environmen," *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT 2002)*, vol. 99, pp. 3–7, 2002.

[29] T. Lin, Kinshuk, and A. Patel, "Cognitive trait model for persistent student modelling," *Proceedings of EdMedia*, vol. AACE, pp. 2144 – 2147, 2003.

[30] Kinshuk, T. Lin, and P. McNab, "Modeling inductive reasoning ability for adaptive virtual learning environment," *Proceedings of the IADIS International Conference on Cognition and Exploratory Learning in a Digital age (CELDA 2004)*, vol. IADIS Press, pp. 343 – 349, 2004.

[31] H. Hong and Kinshuk, "Adaptation to student learning styles in web based educational systems," *Proceedings of ED-MEDIA 2004 World Conference on Educational Multimedia*, vol. USA: AACE, pp. 491 – 496, 2004.

[32] T. Lin and Kinshuk, "Adaptive techniques for supporting working memory capacity," *Proceedings of the International Conference on Computers in Edu-*

cation (ICCE2004), vol. Common Ground Publishing and RMIT University, pp. 1663 – 1667, 2004.

[33] G. Aguilar-Burguete and K. Kaijiri, "Static adaptations of personalization factors," *Interactive Educational Multimedia*, vol. 12, pp. 8 – 16, 2006.

[34] O. Conlon, l. O'Keefe, and S. Tallon, "Combining adaptive hypermedia techniques and ontology reasoning to produce dynamic personalized news services," *Proceedings of the 4th Internalional Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 81 – 90, 2006.

[35] A. Littlejohn, "Reusing resources: a sustainable approach to e-learning," *Kogan page limited*, 2003.

[36] J. E. Gilbert and C. Y. Han, "Adapting instruction in search of a significant difference," *Journal of network and computer applications*, vol. 22, issue 3, pp. 149–160, 1999.

[37] M. Gettenger, "Individual differences in time needed for learning: A review of the literature," *Educational Psychologist*, vol. 19, issue 1, pp. 15 – 29, 1984.

[38] A. Graesser and N. Perso, "Question asking during tutoring," *American Research Journal*, vol. 31, issue 1, pp. 104 – 137, 1994.

[39] B. Bloom, "The two sigma problem: The search for methods of group instruction as effective as one-to-one tutoring.," *Educational Researcher*, vol. 13, issue 6, pp. 4 – 16, 1984.

[40] "Moodle." Available Online, viewed March 20th 2011. `http://www.moodle.org/`.

[41] "Sakai." Available Online, viewed March 20th 2011. `http://sakaiproject.org/`.

[42] "Blackboard." Available Online, viewed March 20th 2011. `http://www.blackboard.com/`.

[43] "Desire2learn." Available Online, viewed March 20th 2011. `http://www.desire2learn.com/`.

[44] A. Patel, Kinshuk, and D. Russell, "A multi-institutional evaluation of intelligent tutoring tools in numeric disiplines.," *Educational Technology and society*, vol. 3, issue 4, pp. 38–40, 2000.

[45] J. S. Brown and R. R. Burton, "Diagnostic models for procedural bugs in basic mathematical skills," *Cognitive Science*, vol. 2, pp. 155–192, apr 1978.

[46] R. R. Burton, "An investigation of computer coaching for informal learning activities," *The International Journal of Man-Machine Studies*, vol. 11, pp. 5–24, jan 1979.

[47] J. R. Carbonell, "Mixed-initiative man-computer instructional dialogues," *BNN Report no. 1971*, 1970.

[48] ADL, "Scorm 2004 3rd edition overview." Available Online, viewed on March 20th 2011. `http://www.adlnet.gov/downloads/DownloadPage.aspx?ID=237`.

[49] K. Maycock and J. G. Keating, "Bridging the gap between adaptive hypermedia systems and the sharable content object reference model.," *Proceedings of the 4th WSEAS conference on E-Activities (E-learning E-Communities E-Commerce E-Management E-Marketing E-Government)*, pp. 69–73, 2005.

[50] "Multimedia educational resource for learning and online teaching." Available Online, viewed March 20th 2011. `http://www.merlot.org/`.

[51] "Jouram." Available Online, viewed March 20th 2011. `http://www.jorum.ac.uk/`.

[52] "National digital learning repository." Available Online, viewed March 20th 2011. `http://www.ndlr.ie/`.

[53] T. Murray, "Design tradeoffs in usability and power for advanced educational software authoring tools," *Educational Technology*, vol. 44, oct.

[54] J. Conklin, "Hypertext: An introduction and survey.," *IEEE Computer*, vol. 20, pp. 17–41, 1987.

[55] O. Conlan, V. Wade, C. Bruen, and M. Green, "Multi-model, metadata driven approach to adaptive hypermedia services for personalized elearning," *Proceedings of the second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 100–111, 2002.

[56] M. Sprect, M. Cravcik, L. Pesin, and R. Klemke, "Authoring adaptive educational hypermedia in winds," *Proceedings of ABIS2001*, pp. 1–8, 2001.

[57] J. Eklund, P. Brusilovsky, and E. Schwarz, "Web-based education for all: A tool for developing adaptive courseware," *Computer Networks and ISDN Systems Proceedings of the seventh international WWW conference*, vol. 30, issues 1-7, pp. 291 – 300, 1998.

[58] J. Eklund and R. Zeiliger, "Navigating the web: Possibilities and practicalities for adaptive navigational support," *Proceedins of Ausweb96*, 1996.

[59] P. Brusilovsky, D. Bra, and G. J. Houben, "Adaptive hypermedia: From systems to framework," *ACM computing surveys*, pp. 40 – 50, 1999.

[60] "Aviation industry cbt (computer-based training) committee." Available Online, viewed March 20th 2011. `http://www.aicc.org/`.

[61] "Innovation adoption learning." Available Online, viewed March 20th 2011. `http://www.imsglobal.org/specifications.html`.

[62] "Ieee learning technology standards committee." Available Online, viewed March 20th 2011. `http://www.ieeeltsc.org:8080/Plone`.

[63] "Ariadne." Available Online, viewed March 20th 2011. `http://www.ariadne-eu.org/`.

[64] C. Staff, "Hypercontext: A framework for adaptive and adaptable hypertext," *PhD Thesis, University of Sussex*, 2001.

[65] B. Martin, "Constraint-based modelling: Representing student knowledge," *New Zealand Journal of Computing*, vol. 7, issue 2, pp. 30 – 38, 1999.

[66] C. Johnson and C. Orwig, "What is a learning style." Available Online, viewed January 2008. `http://www.sil.org/lingualinks/library/Llearning/CJ0625/CJ0676.html`.

[67] "Theory into practice database." Available Online, viewed March 20th 2011. `http://tip.psychology.org`.

[68] D. Kolb and R. Fry, "Towards an applied theory of experiential learning, in theories of group process.," *C.L.Copper London: John Wiley*, pp. 33 – 58, 1975.

[69] H. Gardner, "Multiple intelligences: The theory into practice," *New York: Basic Books*, 1993.

[70] M. R. Felder and K. L. Silverman, "Learning styles and teaching styles in engineering education," *Engr. Education*, vol. 78, issue 7, pp. 674 – 681, 1998.

[71] E. M. Litzinger and B. Osif, "Accommodating diverse learning styles: Designing instruction for electronic information sources," *Library instruction for the 90s. ed. Linda Shirato. Ann Arbor MI: Pierian Press*, pp. 73–81, 1992.

[72] C. K. Briggs and B. Myres, "Myers-briggs type indicator," *Palo Alto CA: Consulting Psychologist Press*, 1977.

[73] M. Sprecht and R. Opperman, "Ace - adaptive courseware environment," *The new review of Hypermedia and Multimedia*, vol. 4, issue 1, pp. 141 – 161, 1998.

[74] J. E. Gilbert and C. Y. Han, "Arthur: A personalised instruction system," *Journal of Computing in Higher Educational Systems*, vol. 14, issue 1, pp. 113–129, 2002.

[75] Kinshuk and T. Lin, "User exploration based adaptation in adaptive learning systems," *International Journal of Information Systems in Education*, vol. 1, issue 1.

[76] K. Maycock, S. Jyothi, and J. G. Keating, "Dynamic profiling to reduce the cognitive load on each learner," *Proceedings of WEBIST 06*, 2006.

[77] N. Cowan, "Attention and memory: An integrated framework.," *New York: Oxford University Press*, 1995.

[78] R. C. Atkinson and R. M. Shiffrin, "Human memory: A proposed system and its control processes," *The Psychology of Learning and Motivation: Advances in Research and THeory*, vol. 2, pp. 89 – 195, 1968.

[79] A. D. Baddely, "Working memory.," *Oxford: Oxford University Press*, 1986.

[80] J. L. Campbell and N. Charness, "Age-related declines in working memory skills: Evidence from a complex calculation task," *Development Psychology*, vol. 26, issue 6, pp. 879 – 888, 1990.

[81] T. A. Salthouse and R. Babcock, "Decomposing adult age differences in working memory," *Development Psychology*, vol. 27, issue 5, pp. 763 – 776, 1991.

[82] H. Haui, "Cognitive styles and memory capacity: effects of concept as a learning method," *Doctorial Thesis Twente University The Netherlands*, 2000.

[83] R. M. Felder, "Learning and teaching styles in engineering education," *Engineering Education*, vol. 78, issue 7, pp. 674 – 681, 1988.

[84] "Carroll human cognitive abilities project." Available Online, viewed March 20th 2011. `http://www.iapsych.com/chcdef.htm`.

[85] "Java open document." Available Online, viewed March 20th 2011. `http://www.artofsolving.com/opensource/jodconverter`.

[86] M. Dorigo and V. Maniezzo, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 26, issue 1, pp. 29–41, 1996.

[87] R. G. Reynolds, "An introduction to cultural algorithms," *Proceedings of the 3rd Annual Conference on Evolutionary Programming*, vol. 99, pp. 131–139, 1994.

[88] R. G. Reynolds and C. Chung, "A self-adaptive approach to representation shifts in cultural algorithms," *Proceedings of IEEE International Conference of Evolutionary Computation*, pp. 94–99, 1996.

[89] M. Mitchell, S. Forrest, and J. Holland, "The royal road for genetic algorithms: Fitness landscapes and ga performance," *Proceedings of the First European Conference on Artificial Life*, pp. 245–254, 1991.

[90] S. Boettcher and A. G. Percus, "Optimisation with extremal dynamics," *Physical Review Letters*, vol. 86, pp. 5211–5214, jun 2001.

[91] R. Battiti, M. Brunato, and F. Mascia, "Reactive search intelligent optimization," *Operations Research / Computer Science Interfaces Series*, vol. 45, 2008.

[92] S. KirkPatrick, "Optimization by simulated annealing: Quantitative studies," *Journal of Statistical Physics*, vol. 34, Numbers 5-6, pp. 975–986, 2005.

[93] T. M. Mitchell, "Machine learning," *McGraw-Hill International Editions*, pp. 249 – 268, 1997.

[94] G. et al., "Lamarckian learning in multi-agent environments," *Proceedings of the Fourth international conference on Genetic ALgorithms.*, pp. 303–310, 1991.

[95] D. E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," *Foundations of Genetic Algorithms*, pp. 69–93, 1991.

[96] R. Flesch, "A new readability yardstick," *Journal of Applied Psychology*, vol. 32, jun.