Chapter Two | Literature Review

2.01 Introduction

E-learning can be defined as being:

The delivery of a learning, training or education program by electronic means. E-learning involves the use of a computer or electronic device (e.g. a mobile phone) in some way to provide training, educational or learning material. (Stockley, 2007)

E-learning can be provided on internet websites, on closed networks, via CD-ROM, DVD, hand-held personal digital assistants (PDAs) or mobile phones. In short, it can be provided on a greater variety of equipment than just a computer connected to the internet (Stockley, 2007). E-learning can employ text, video, audio, animations and virtual environments. The learning can be synchronous – where the learner has to be ‘online’ at a particular time, or asynchronous – where it can be done at anytime, anywhere; self-paced interactive learning integrating support via online bulletin boards, chat rooms, e-mail or instant messaging. It can also come in the form of knowledge databases, where users click through information that is retrieved from a database and is only mildly interactive (Obringer, 2007).

Networked e-learning environments are suited to multiple methods of communication and contact between participants. E-learning courseware is also suited to simulating real world situations, phenomena and procedures. The interactive nature (for example: ‘click and explore’, reveal, build, move, simulate) of e-learning provides the opportunity to ‘try out’ situations prior to real life implementation - e-learning courseware therefore lends itself to active learning and experimentation. There is a strong imperative for e-learning to address the needs of the online generation, ‘digital natives’ who prefer to multitask, use non-linear means to access information and who have a heavy reliance on information and communication technology (ICTs) for their social communication and active learning needs (Litchfield et al., 2007; Prensky, 2007). E-learning courseware that inhibits student-to-student communication and tutorial support lacks a fundamental building block in the learning process. The irony of e-learning is that new media provide multiple opportunities for human-to-human communication and human-computer interaction, yet these interactions are often designed out.

The design process for e-learning courseware is complex. In the design of online environments in general, Spillers has identified eight distinct design disciplines (Spillers, 2007). These include:
- User interface design
- Information architecture
- Interaction design (HCI)
- Graphic design
- Interactive design (e.g, Flash)
- Emotion design
- User-centred design
- User experience design (Stockley, 2007)

Each one of these design disciplines may involve one or more people. Added into this mix are the multiple stakeholders involved in any design project. The stakeholders in the design of e-learning courseware may include:

- Graphic Designer
- Programmer or Interactive Designer
- Content Producer
- Information Architect
- Senior Producer
- Executive Producer
- Client
- Subject Matter Experts
- Writers
- Course Designer
- Tutors/Lecturers/Educators (end users)
- Students (end users)

Each one of these stakeholders forms and shapes the end product, and may demand changes or revisions at any stage.

The issue here is how to embed theories of teaching and learning into e-learning without unduly increasing the complexity of the design process. This involves identifying and addressing learning outcomes, incorporating different learner characteristics (such as learning styles) and using theories on how people learn to address the pedagogical approaches, strategies and tactics (Goodyear, 2005) that could be embodied in the user interface, information architecture/navigation and content layout of e-learning courseware. But first, I will examine why the current approaches to the design of e-learning courseware are insufficient.

The design of e-learning courseware requires a different approach to those employed by the developers of the majority of interactive online and screen-based applications. Commercial
website developers, for example, often measure success by the number of page impressions or how fast the customers proceed to the checkout (Van Duyne, Landay, & Hong, 2003). This is a transactional view of interaction, which does not account for the needs of online learners.

For e-learning the criteria for the success of a piece of courseware needs to be refocused onto pedagogy, allowing the students time for reflection, consideration, contemplation and interaction with the learning materials. A satisfying learning experience includes observation, reflecting on those observations, constructing new knowledge, being able to create theories based on this new knowledge and putting those theories into practice (Kolb, 1984). The learning experience should involve socialising with other students and interacting with peers, scholars and tutors, in an environment that is easy to understand and use, which encourages interaction – with the materials themselves and with other people. This socialisation, according to both experiential learning and social constructivism, validates what has been learned. Students compare their learning with the real world, and socialisation forms part of this validation.

Current opinions of e-learning interface and information architecture are that they are disappointing, have poor content, contain inauthentic learning, emphasise form over substance, have a lack of standards, are boring or ‘shovel ware’ - paper-based material mounted directly on the Web (Chee, 2004; Teo & Gay, 2006). Learning Management Systems (LMS) such as WebCT sit behind learning portals, allowing access to e-learning courseware. However, these systems are limited, not only in how courseware is able to be designed, but also in the level of interaction of the final learning experience.

Students need more than what has been referred to as a ‘spray and pray’ approach, (Chee, 2004) which is fragmented and does not engender the retention of knowledge. Too much emphasis has been placed on access to learning content, to the detriment of learning. Enabling access to materials is not education (Chee, 2004). Poor instructional design is one of the key problems with learning from the Web (Frizell, 2003b).

Psychological and pedagogical theories agree that knowledge that is not put into practice or used is quickly forgotten (Derntl & Motschnig-Pitrik, 2004; Wenger, 1998). Deep and persistent learning occurs when the experience takes into consideration the individual’s needs, styles, interests and incorporates social learning. E-learning courseware ideally would incorporate these different aspects of learning, responding to student's actions – activities as opposed to assessment. To integrate these requirements, the overall design of the e-learning course (architecture, look and feel, interface design, content design) is as important as that of the individual learning modules themselves, and these two aspects of learning design have to go ‘hand in hand’ (Derntl & Motschnig-Pitrik, 2004).
The technologies to implement e-learning courseware have been available for some time. Web pages, Flash, Shockwave, PDF and even the QuickTime formats allow designers to create highly interactive courseware, which not only allow ‘trying out’ problems via simulation, but also allow the exploration of ‘real world’ situations.

To date there has been a lack of widespread pedagogical innovation in e-learning design (Dillon, 2004). The innovation has been in improving access to materials and widening participation in learning, in making content more comprehensible – but the improvement in quality has been a product of creativity, in improving the way ‘social actors involved in the educational situation work with the technology, rather than with the technology itself’ (Dillon, 2004, p. 143).

The solitary experience for students who are left alone to work through content can be demotivating – a degree of guidance, handholding and mentoring is best done through contact with human teachers and fellow students. Social learning does not occur naturally in e-learning situations and must be explicitly introduced by the lecturer or tutor, and this functionality needs to be available to them in the application design (Berge, 1995). If the social dimension is considered when designing e-learning experiences, students will be better able to construct their knowledge through the additional interaction and learning opportunities provided with the contact of fellow students and tutors.

Standard features of learning technology such as bulletin boards, forum, chat, 3D virtual environments and advanced features such as simulations, intelligent agents and others have been technically well defined. However, much remains to be done in the reengineering of the learning process to exploit the technology so that it goes beyond representation and instead redefines the learning experience (Derntl & Motschnig-Pitrik, 2004). By embedding pedagogical theories into the design of e-learning courseware it may be possible to improve the learning experience.

Two theories of learning explain why the lack of interaction impairs learning experiences. Constructivist theory states that each individual constructs his or her knowledge from practical experience acquired by multiple interactions with materials, objects, procedures and phenomena of the world (Piaget, 1999; Solomonidou & Kolokotronis, 2004; Vygotsky, 1978). Learners, therefore, learn through interaction with the learning materials and by subsequently practising what they learn using ‘real world’ examples that reflect how the processes and systems of the world work. Social constructivists also claim that the interaction between the participants in the learning process is core to the construction of knowledge, and this is as important as the interactions with the learning materials themselves. Participants include not only the students’ peers, but also the teacher, the
community of scholars and like-minded students. Fellow students act not only as learners, but also as authors, critics and scholars, and this valuable interaction is important in the construction of knowledge (Dalgarino, 2002).

Experiential learning theory states that students form theories about their learning and consolidate their knowledge through active experimentation – learning through experience. This is best done using ‘real world’ examples, with a focus on practical application (or ‘hands on’) learning (Kolb, 1984). E-learning is suited to active experimentation as screen-based learning can be highly interactive. However, many e-learning courses succeed in presenting usable learning materials, but fail to consolidate the learner’s knowledge. According to the Kolb learning cycle (Figure 2-3), this consolidation occurs after the student has formulated new theories based upon their experiences. Without testing the new knowledge against the ‘real world’ knowledge construction is less likely to solidify. Experiential learning theory is discussed further in Section 2.05.

These two theories complement each other: they encompass how knowledge is constructed, consolidated and is put into practice, while taking the social aspects of learning into consideration. This research will use these theories as a pedagogical philosophy (Goodyear, 2005), however the method and theoretical framework underlying the development of design for pedagogy patterns is generalisable: the focus is on embedding pedagogical theories into the design of e-learning courseware, irrespective of the particular teaching and learning theory or theories employed.

2.02 Where Usability is King

Current practices in e-learning place the focus of the design process on usability, (Van Duyne et al., 2003) rather than learning (Retalis, Georgiakakis, & Dimitriadis, 2006). Usability is the idea that interactive applications should be easy to use and in order to do this, developers have to take into consideration the psychological, ergonomic, social and organisational factors that determine how people work. They also have to take into consideration group working, user interaction and how the media is integrated with the entire user experience of the application. However, there is often a gap between what is known about interface design and design practice (Shield & Kukulska-Hulme, 2006).

Developers have attempted to overcome this gap by placing the user at the centre of the design process, where ease of use, technical performance, content and satisfaction are the key issues (Van Duyne et al., 2003). User-centred design is a process of iterative design, which is cyclical: design, prototype and evaluate (repeat N times until usability goals are satisfied). Usability revisions may necessitate multiple changes to the look and feel, interface, user
interaction, information architecture or content. The evaluation of the prototype involves empirical measurement of the user interaction firstly by early use of paper-based designs and further into development by using simulations and prototypes. The user's performance (e.g. how easily they navigate through a system) and reactions are observed, recorded, measured and analysed (Preece, Rogers, & Sharp, 2002). This human-computer interaction (HCI) usability process has become an industry standard.

Usability does not address issues such as poor instructional design. Poor instructional design is that which does not enable students to achieve the desired pedagogical goals and objectives (LeLoup & Pontierio, 2010). The materials may well be usable in terms of navigation and ease of use, but lack the context of a teaching and learning environment. They may be usable in terms of human-computer interaction, but may not be usable in terms of achieving the desired learning outcomes, that is, achieving students' pedagogical goals and objectives for a specific topic. The problem with the current practice of prioritising usability over pedagogically-oriented environments is that the learning experience of the entire e-learning environment may be overlooked (Preece et al., 2002). This is likely to affect the development of the learning environment, compromising how learning materials are incorporated, the user-interface, modes of communication, collaboration spaces and so on. The environments may well be usable, but do they achieve the pedagogical goals required? The e-learning courseware as a whole not only needs to be usable, it must also be ‘learner centred’.

Usability does not take into account:

- Theories of how people learn
- Social interactions between students
- Learning outcomes
- Collaboration
- Communication between students and tutor
- ‘Real world’ problem solving
- Simulations
- Interactive learning materials

E-learning courseware, no matter how sophisticated the technology used, should enhance the learning and interaction at the cognitive, behavioural and physiological levels (Teo & Gay, 2006). In essence, there is a lack of focus on pedagogical methodology, of the overall learning experience in the design of e-learning courseware. One solution to this is to rely on educational theories to drive the design of e-learning courseware. How then does one embed
pedagogical theories into the design of e-learning courseware? Pattern languages may be able to provide a solution.

2.03 Design Patterns and Pattern Languages

Designing e-learning courseware is a particularly complex task. Design is recognised as being a 'wicked' problem (Rittel & Webber, 1973) – one that has numerous stakeholders with conflicting perspectives and that cannot be accurately modelled or addressed by the techniques of science and engineering. Wicked problems have multiple solutions that integrate multiple facets. One way of dealing with a wicked problem is the use of design patterns and pattern languages.

A pattern language is comprised of design patterns. Each design pattern is a text document that forms a core module in the pattern language, but is structured so that it links to other patterns in an integrated way (Alexander et al., 1977; Derntl & Motschnig-Pitrik, 2004). Each design pattern contains smaller elements, or sections, some of which provide linkages to patterns at a higher level of abstraction (upwards), or to patterns that provide more detail (downwards). In this way the user can work ‘top-down’ or ‘bottom-up’ if they so choose. A pattern language is specific to a particular design domain, and is made up of an entire set of integrated design patterns.

The core of each design pattern is a proven solution to a common problem within a specified context (Bloodworth, 2008). When using a pattern language to aid in a design process, the user chooses the patterns that are most relevant to the specific set of problems they wish to solve and follows the linkages between the patterns as appropriate. In this way, the same pattern language can be used in multiple design situations. Each set of solutions will be specific to the design task in hand, depending on the set of patterns chosen by the user. By breaking a set of design problems down into smaller integrated components, a pattern language provides a shared vocabulary for designers to capture and transmit the design process (Chan, 2003).

Designing using patterns began with a series of papers written by Christopher Alexander in the 1960s, culminating in the publication of the book A Pattern Language in 1977 (Alexander et al.).

Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice. (Alexander et al., 1977, p. x)
Originally a generic problem/solution set, patterns were developed as a way of describing and solving architectural design problems. They have since been developed as a language of design. Each design domain requires its own pattern language, its own way of solving problems. Since their inception, pattern languages have been applied to a number of domains, from their originating field of architecture to areas such as software design and object orientated programming, to name a few (Bloodworth, 2008). Design patterns provide concise and accurate communication for designers, and guidance for novices (Frizell, 2003a). When written in an accessible way that enables novice designers to connect to new problems, design patterns have been shown to be an effective tool in teaching and enabling software design (Clancy & Linn, 1999).

Patterns focus on modelling workflows and processes that are completely abstracted from specific content (Derntl & Motschnig-Pitrik, 2004). One of the most comprehensive sources for online website patterns is to be found in The Design of Sites, which organises the patterns in genres, which in turn references lower level patterns (Van Duyne et al., 2003). However, Van Duyne’s work focuses on customer based e-commerce sites and although some of the patterns are applicable to e-learning, it does not provide complete solutions for e-learning developers. Pattern languages have been developed for some e-learning solutions, such as Learning Management Systems and blended learning approaches, (Avgeriou et al., 2003; Chan, 2003; Derntl & Motschnig-Pitrik, 2004; Frizell, 2003a) but what is lacking is an approach that embeds pedagogical theories into the patterns and therefore the design process itself.

Although Alexander’s pattern language dealt with architectural design and town planning, the structure of large-to-small elements in design can be mapped to the structure of e-learning design (Figure 4-1). This mapping can form the foundation of a strategy for the creation of a pattern language for e-learning. By commencing with broader environments, such as online portals then drilling down through a subject area, domain or course, then ultimately to individual topics/activities, pages and interface elements, the design of entire online environments can be managed.

In the design of an e-learning system there are more aspects to consider than just the one-to-one mapping of an e-learning application to Alexander’s language of physical space; there are pedagogical techniques to consider as well. Goodyear (2004) states that:

Thinking in similar ways about the design space of networked learning, one can advance some tentative proposals about an equivalent pattern language. What would be the largest pattern, equivalent to Alexander’s ‘Independent region’? I suspect it would be a course, or Programme of study. This is the largest entity which can be designed. At smaller scale levels there are the building blocks of a course, however one labels them in one’s own system or institution – Study unit, Module, etc. Then
there are the kinds of pedagogical technique catalogued by Paulsen: Discussion group, Debate, etc. Within these are smaller pedagogical tactics (tasks), smaller organizational forms, as well as the tools and artefacts with which we populate the learning space. (Goodyear, 2004, p. 344)

As Gooyear and Alexander both point out, the utility of a pattern language is not in the individual patterns themselves, but in the links and relationships between the patterns. An individual pattern is difficult to evaluate on its own. It must be seen in context of the patterns that support and surround it (Alexander et al., 1977). Patterns contain sections that link them to the patterns that sit at higher levels of abstraction in the pattern language and the sections that are smaller elements that sit below. Each pattern contains a specific problem formed in the context of a problem question, which is solved by the solution (Figure 2-1). Each problem must be examined in detail, trying to capture a generic and recurrent problem/solution set through the lens of pedagogical theory, and at the same time the language must be structured so that it encompasses larger and smaller patterns and the complex relationships between them. Writing a pattern language is done on two levels – looking at the detail of the pattern itself and its relationship to those that surround it – a bottom-up and top-down approach at the same time (Alexander et al., 1977; Goodyear, 2004).

Figure 2-1 Alexander’s pattern structure (Alexander et al., 1977).
This research study will examine the first of these levels: embedding pedagogy into the pattern itself. The writing of an entire pattern language and linkages between the patterns will be beyond the scope of this research.

The problem with the classic Alexandrian pattern structure (Figure 2-1) is that when attempting to map pedagogical theories into an Alexandrian pattern, the focus moves away from design onto pedagogy. The pattern becomes an entirely pedagogical one. If the focus is maintained entirely on design there remains little scope within the pattern structure for including pedagogy. It becomes an either/or proposition. Although Alexander’s pattern structure is not necessarily the immediate cause of this, introducing pedagogy into the pattern adds an additional layer of complexity when creating the pattern document. One solution to this is to modify Alexander’s pattern structure. This will be addressed in Chapter Four where Alexander’s pattern structure is adapted to include a pedagogical framework (Goodyear, 2005).

Design patterns can augment e-learning courseware design by:

- Providing the designer/teacher with a set of e-learning courseware design ideas
- Creating a structure so that it is easy to understand the relation between each pattern
- Providing a clear articulation between the design problem and solution by providing an underlying principle that bridges research-based evidence with the designer’s experiential knowledge of design
- Encoding the designer’s experiential knowledge in a way that integrates with the iterative design process over the entire production process (Goodyear, 2004)

Some problems with e-learning patterns are that, unlike architectural patterns, it is difficult to locate them. One attempt to create a resource for e-learning practitioners is the Design Principles Database (Kali & Linn, 2011), which is an integrated database of design patterns and principles. There are a number of papers available but at the time of writing only one published refereed book (Goodyear & Retalis, 2010). Another problem is validation. There are few empirical studies in e-learning design patterns and very few published results (Garzotto & Retalis, 2007). Technology-Enhanced Learning (Goodyear & Retalis, 2010) contains a number of papers on e-learning patterns, but only two contain pattern evaluation.
When considering a pattern language for e-learning there are higher level pedagogical issues to manage as well. Goodyear (2005) describes these as the ‘pedagogical framework’ (Figure 2-2) that defines the educational problem space. He conceptualises the educational problem space as having two layers of tasks - the pedagogical framework and the educational setting. These interact with each other to form the organisational space that informs the learning outcomes.

The pedagogical framework Goodyear sees as:

- pedagogical philosophy (a set of beliefs on how we think people learn, what knowledge consists of, how we think people should be treated, etc.)
- high level pedagogy (broad approaches such as problem based learning, cognitive apprenticeship, collaborative knowledge building)
- pedagogical strategy (directly related to action, e.g. the use of an online debate) and
- pedagogical tactics (the detailed methods we use to set tasks for students, encourage their participation, offer guidance and feedback, etc.) (Goodyear, 2004)

The importance of an online team having the same pedagogical philosophy cannot be understated. If members of a team approach the creation of an educational e-learning course with deep philosophical differences, this ‘can lead to fatal divergence in the day to day operational work’ (Goodyear, 2005, p. 84). From this flows the high level pedagogy, the approach taken when designing a particular approach, for example, ‘problem based learning’ and from these the strategy and tactics flow.

For a pattern language used for teaching and learning to maintain consistency and cohesion, it is important that it contain a pedagogical philosophy. This philosophy guides the pedagogical techniques used within the pattern language and aids the pattern writers in
refining the scope of the entire language. It helps the writers to clarify the approaches they can use in defining individual pattern solutions.

It does not matter which particular learning theory is used, but it is necessary for a pattern language for e-learning courseware to contain a pedagogical philosophy. This philosophy guides the form and structure of the pattern language, and informs the types of pedagogical practices and activities that can be encapsulated within the pattern solutions. By allowing the designers’ insights into the pedagogy required when enacting the pattern, this in turn would influence the design process to include pedagogy. In turn, if educators were also to understand how the resulting courseware was intended to be used, and what teaching strategies were envisaged by the designers, the courseware could be put into use in the way that it was intended.

This research creates a method that allows a pedagogical philosophy to be embedded into design patterns. The pedagogical philosophy uses both experiential learning and constructivist learning theories as a basis. It takes as its foundation Kolb’s learning cycle, and adds a social dimension to it by incorporating constructivist theories on knowledge creation and social learning (see 2.04 Theories of Learning). The patterns have been created looking through the lens of constructivist and experiential learning. This will inform the high level pedagogy of each learning solution and the strategic and pedagogical tactics will be created in accordance with these learning and teaching theories. This will be outlined in Chapter Four: A Method for Creating a Design for Pedagogy Pattern.

Pedagogical patterns certainly exist that contain a pedagogical philosophy (Goodyear, 2005), and indeed Garzotto and Retalis (2007) have proposed a taxonomic approach in the development of design patterns for pedagogy, in which an entire pattern language may be made up of patterns that address:

- human actors
- the didactics of subject matters
- pedagogical strategies
- learning objects
- learning tools

While this taxonomy takes a pedagogical approach to the overall pattern language, specific pedagogical theories and teaching strategies are not embedded into each individual pattern. Without an overall pedagogical philosophy unifying the tactics, strategies and higher-level pedagogies, as well as informing each pattern, the results could be inconsistent.
There are further considerations that need to be taken into account when creating patterns that embed pedagogy within them – design for pedagogy patterns for e-learning. Patterns also need to be well-written.

### 2.03.01 The Well-Written Pattern

A pattern will not be usable by the target audience if it is not written well. There are general principles that can be followed by pattern writers to create patterns that are accessible and easy to use. If these principles are put into practice, the task of the pattern writer can be facilitated, and the users of the pattern will find them easy to put into practice. Design patterns and pattern languages have been used successfully in the domain of software development (Larman, 2002; Pree, 1995). Although software development patterns address a different domain to those that would be integrated into e-learning courseware, lessons learned from their use could be used to help create a well-written design for pedagogy pattern for e-learning. This section examines research literature from the domain of software development patterns and draws from it the principles or properties of the well-written pattern.

One of the problems with pattern writing is that it is a complex activity. Some of the characteristics that make pattern writing more complex than other types of writing are:

- Needing to fulfil a specification (each pattern requires a specific structure and addresses a specific design problem within a domain)
- Multiple stakeholders (users of the pattern)
- Multiple possible outcomes (each pattern is enacted by each user in a different way) (Alexander et al., 1977)

Pattern writing has become the subject of a pattern language - *A Pattern Language for Pattern Writing* (Meszaros & Doble, 1997). This pattern language deals with issues such as using ‘skippable sections’, ‘problem/solution summary’, ‘running examples’ (i.e. case studies), ‘relationship to other patterns’, ‘evocative pattern names’ and so on (Meszaros & Doble, 1997, p. 527). That a pattern language has been written for pattern writing acknowledges the complexity of the task.

Although pattern languages have been posited as a solution methodology to a class of design problems, there are also problems with their use. One is in how the patterns are written, and the level of expertise at which they are aimed. Novice users may find expert patterns inaccessible. Expert patterns assume a level of knowledge and use domain-specific terminology which requires further explanation if used by novices.

In one study of a class using information technology patterns, two groups of students were given two sets of patterns addressing the same problem. One group was given a pattern
written using *descriptive* rules that described what the patterns did, while another group was given a pattern written using *constructive* rules that described how the pattern should be used. Students given the *constructive* rules created significantly more correct results, 18 out of 20 when compared to 12 out of 20 for those using the descriptive rules (Clancy & Linn, 1999; Shackelford & Badre, 1993).

Studies taken from pattern use in software development have revealed that using narrative case studies can make patterns easier to use. Case studies allow users to understand the context of the pattern, and how it can be enacted within that context. In one study, a case-based pedagogical approach was tested in ten high school Pascal programming courses (Clancy & Linn, 1999). A case study is a fully worked out solution to a programming problem with narrative explanation of the development process along with study questions designed to help students consider alternative approaches, similarities and differences between code segments and multiple representations of programming concepts. All classes were given an introductory case study. Then, classes were randomly assigned to one of three groups: working with the case study alone, solving the problem first then working with the case study, solving the problem first then being given a solution programme to study. The students were then tested to assess their understanding of programming, without relying on anything contained in the narrative. The resulting scores showed that students who used the complete narrative significantly outperformed those who did not. The use of narrative case studies appeared to help the students distinguish between the design patterns and to understand how to use them (Clancy & Linn, 1999). If a fully worked narrative case study is included within the pattern document, this should help the users understand the context of the pattern's use.

Pattern languages exist that outline the software development process, which make object-orientated programming and software design less complicated (Shalloway & Trott, 2005). The design patterns in these pattern languages differ from case-based patterns, as fully worked out narrative examples are not provided. Also, they could not be said to be written in a *constructive* way, as the patterns do not outline a set of *rules* describing how the pattern should be used. The goal of these patterns is to provide related code snippets that the programmers then adapt and use directly in their own computer programs.

These patterns take the same basic approach as Alexander, working from larger methods and functions and then drilling down to smaller object-orientated code snippets that programmers can use (Shalloway & Trott, 2005). In this approach, the structure and relationships between the patterns are tightly integrated. The modular structure of pattern languages lends itself to object-orientated programming, as pattern languages themselves are object-orientated. Shalloway and Trott (2005) have outlined a methodology for creating a
pattern language for object-oriented programming, however, the structure and methodology of creating a pattern language for object-oriented computer programming does not necessarily map to the needs and methodologies of e-learning courseware design. In a broad sense the top-down, bottom-up approach is generalisable to all pattern languages, but the specific methods to do this for programming languages are not. The intended audience for these patterns is software developers, writing code in the Java, C++ or C# programming languages. These design patterns are not intended for website or flash-based interaction design (Shalloway & Trott, 2005, p. 233). Although such literature is useful in helping the writer to ‘think in patterns’ (Shalloway & Trott, 2005, p. 233) and defining the relationship between interconnecting patterns, its methodology is closely tied to programming, and does not generalise to the domain of e-learning. However, the tight integration and relationships between the patterns used in this approach are worthy of note.

How to extract a design pattern from prior solutions can also add to the complexity of the pattern writing task. Some examples in literature on pattern languages for computer-based interaction place the emphasis on a customer-centred design approach using focus groups and usability testing to aid the pattern writing process (Van Duyne et al., 2003). This is a common and important process in software specification and development, and the authors suggest that the design patterns are a ‘powerful conceptual framework for building compelling, effective and easy-to-use Web sites’ (Van Duyne et al., 2003, p. 19). Using the customer-centred approach for pattern writing, a series of focus groups are formed to help in the pattern definition process. Users are asked to help define the pattern problems that need to be addressed by the pattern language, and then suggest possible solutions to these problems. Once the focus group research has been completed, the research is handed over to the pattern writers who then write the pattern. The method taken by writers to accomplish the pattern writing is not outlined.

A user-centred approach is also taken by Schümmer and Lukosch (2007) in their patterns for computer-based interaction. Once the pattern problem is defined in this approach, it is given to a pattern writer who completes the pattern. The process of design pattern writing is not outlined, nor is a defined method presented.

In summary, incorporating these principles into the writing of a pedagogical pattern for e-learning could be achieved by:

- Using constructive rules (Shackelford & Badre, 1993)
- Using a narrative case study (Clancy & Linn, 1999)
- Making some sections skippable (Meszaros & Doble, 1997)
- Creating a structure that includes tight relationship to other patterns, (Shalloway & Trott, 2005)
According to the above research, using these principles would result in a pattern language that is usable and cohesive in its overall pedagogical approach.

There appears to be no method in the current literature for pattern writers to follow to achieve the writing of the pattern, nor how to embed pedagogical theories into the pattern. Nor has there been any reported evaluation of the design patterns to determine if their use facilitates the design process for e-learning courseware. This research gap is significant and will be addressed by this research.

In the next sections, two learning theories will be examined. These theories appear to be the most relevant to e-learning design, and will be used to form the pedagogical philosophy of the design pattern language that will be developed in this research.

### 2.04 Theories of Learning

E-learning courseware should not only be examined in terms of a teaching and learning resource, but also in terms of teaching strategies, how the materials are to be used within the learning context. Instructional strategies need therefore to be informed not only by theories of learning, but also the pedagogies that apply to those theories and how they impact upon instructional design and practice (Adams et al., 1996).

Pedagogic theories loosely fall into two approaches:

- A psychological approach that looks at the internal mental processes and how these change as a student learns
- A social and contextual approach that investigates how learning is made possible through other people and the student's connection with the learning materials as well as the world outside (Adams et al., 1996)

Two learning theories will now be discussed in order to reveal the approach each takes to interaction, 'real world' problem solving and how it is thought that learners learn, situated within the learning environment and including social interaction. These sections will reveal how each theory could inform the e-learning design process. It is possible to investigate how each learning theory informs and relates to design principles and these will be examined at the end of each section. Once the design principles that are informed by each learning theory are established, they can then be embedded into design for pedagogy patterns, which will be expanded upon further in Chapter Three: Approach and Methods.
2.05 Experiential Learning

Experiential learning theory takes as its basic tenet that people learn from their experience. The theory draws primarily from the intellectual background of Jean Piaget, John Dewey, David Kolb and Kurt Lewin (Dewey, 1938; Kolb, 1984; Lewin, 1951; Piaget, 1951, 1970a, 1970b, 1978). A reaction against behaviourism, experiential learning theory offers the foundation of an approach to education and learning as a lifelong process and is based in the intellectual traditions of social psychology, philosophy and cognitive psychology. The experiential learning model links the basic competencies of job demands to their corresponding educational objectives, emphasising the critical linkages between the classroom and the ‘real world’ with experiential learning methods (Kolb, 1984, p. 4).

As John Dewey put it ‘there is an intimate and necessary relation between the processes of actual experience and education’ (Dewey, 1938, pp. 19‐20). Experiential learning theory examines and strengthens the linkages between education, work and personal development (Kolb, 1984, p. 4).

Based on the work of Dewey, Lewin and Piaget, Kolb’s experiential learning theory takes a different philosophical view to that of more traditional educational and behaviourist approaches. Traditional approaches state that there are mental atoms, or simple ideas that are immutable – they are constant fixed elements of thought. Experiential learning theory assumes that learning is a ‘process whereby knowledge is created through the transformation of experience’ (Kolb, 1984, p. 38). Ideas are constantly formed and re-formed. The emphasis is in the process of learning, not necessarily the memorising of a body of knowledge. Experience is always modifying thoughts, and that is why no two thoughts are ever the same – experience always intervenes.

Learning is, by its very nature, a ‘tension and conflict filled process’ (Kolb, 1984, p. 30). Kolb believes that effective learners require four different kinds of abilities; concrete experience, reflective observation, abstract conceptualisation and active experimentation (Figure 2-3). He defines concrete experience as the learners’ ability to involve themselves fully and openly in new experiences. Reflective observation is the ability of learners to reflect on their own experiences from many perspectives. The ability to integrate these observations into logically sound theories Kolb calls abstract conceptualisation. Learners have to be able to use these theories to solve problems and make decisions – this Kolb calls active experimentation.
One aspect of learning that Kolb admits is under explored in research is the interaction between learners and their environment. He states that ‘learning involves transactions between the person and the environment’ (Kolb, 1984, p. 34). However, its implications have been largely ignored in research, which instead focuses on a person-centred psychological view of learning. The wider world environment, according to Kolb, is seen to shape behaviour but his contemporary researchers disregarded how behaviour shapes the environment (Kolb, 1984, p. 34). This person-centred view also decontextualises learning – experiential learning laboratory situations bore so little resemblance to real learning conditions that the results could not be extended to the subjects in their natural environment (Kolb, 1984).

In experiential learning theory, the relationship between learner and the environment, according to Kolb, is symbolised in the dual meaning of the word *experience*. One meaning is subjective and personal, as in ‘I have experienced great joy’ - the other is objective, as in ‘She has 20 years’ experience in her job’. These two meanings interact in complex ways – every experience is subjective and at the same time objective (Kolb, 1984). The word *interaction*, according to Dewey, assigns equal rights to both objective and subjective meanings of experience (Dewey, 1938). Therefore, a learning experience cannot be divorced from its environment – the layers of meaning are intertwined.

Kolb furthers this argument by stating that there are two distinct modes of grasping experience – comprehension and apprehension. Apprehension is the environment’s input into a person’s sense of situatedness – the feel of a chair on a person’s back, the blue glow of a
desk lamp, the colour of the room. The colours, textures and sounds of a person’s
environment are apprehended – this is how our bodies perceive reality, through sight,
hearing, sound, taste and touch. It is effortless – apprehended with:

no need for rational enquiry or analytical confirmation... In this sense, concepts and
the associated mode of knowing called comprehension seem secondary and
somewhat arbitrary ways of knowing. Through comprehension we introduce order
into what would otherwise be a seamless, unpredictable flow of apprehended
sensations, but at the price of shaping (distorting) and forever changing that flow.
(Kolb, 1984, p. 43)

Comprehension of experience can be communicated in a way that transcends apprehension,
which disappears as soon as you change your situation, for example, leaving the room with
the chair and lamp, the former is replaced by new apprehensions. The comprehension of the
experience allows you to transmit that knowledge in the form of a model of that situation, to
others (Kolb, 1984). And it is here that understanding of the importance of the situatedness
or the environment that encapsulates the learning begins. The comprehension of learning
materials is the way that the student orders that which he or she has apprehended. And for
that, environment is key.

A ‘trying out’ or experience-based approach is at the core of Kolb’s experiential learning
theory, and if encapsulated in a collaborative learning space provides the ‘real world’ context
outlined in social constructivist theory. The ‘what-if’ ability available in e-learning is the
difference between e-learning and classroom-based learning and, according to both
experiential and constructivist learning theories, affords additional opportunities in the
consolidation of knowledge.

E-learning provides the opportunity to cater for learners who transform their knowledge via
intention (the more reflective type of learner) and for those who transform via extension,
(those who jump into a learning activity and are able to cope with mistakes). Those who
transform their learning via intention need time to contemplate the theories and information
given, to be certain of their thinking before they commence a learning activity. Because much
of e-learning is not as subject to time constraints as classroom-based learning, these learners
are able to spend the time they need to assimilate the knowledge enough to be confident
when undertaking a learning activity. Similarly those who transform via extension – the type
of learner who likes to ‘jump in’ to a learning activity can do so as the interactive nature of
e-learning allows them to skip ahead to engage in the interactive activity before returning
additional materials as needed.
2.05.01 Design Principles from Experiential Learning Theory

Experiential Learning theory has at its core the ideas that learners develop through ‘hands on’ exercises and using ‘real world experience’. Drawing from Kolb’s learning cycle and ideas outlined in his book (Kolb, 1984), it is possible to extract design principles from his theories. Interaction design principles arising out of experiential learning theory that can be applied to e-learning are described in Table 2-1.

<table>
<thead>
<tr>
<th>Experiential Learning Principle</th>
<th>Design Principle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create interactive learning exercises that incorporate all four different kinds of abilities.</td>
<td>Provide materials that form concrete experience (e.g. correct examples or demonstrations), encourage reflection, allow students to form new theories and test those theories using ‘hands on’ exercises (Kolb, 1984).</td>
</tr>
<tr>
<td>Use visual cues to allow apprehension.</td>
<td>Use visual cues within the courseware to allow apprehension. Rich images with relevant objects placed in an environment add this extra layer of meaning (Kolb, 1984).</td>
</tr>
<tr>
<td>Create exercises where learners can ‘try out’ their new knowledge.</td>
<td>Use highly interactive ‘hands on’ exercises, such as simulations, ‘click and learn’, or ‘cause and effect’ interactives.</td>
</tr>
<tr>
<td>Cater for both reflective users and those who like to jump straight in to the exercises.</td>
<td>Allow ease of switching (e.g., through navigation) between learning exercises.</td>
</tr>
</tbody>
</table>

2.06 Constructivism and Social Constructivism

Taking an epistemological perspective, constructivism is concerned with how we know and how we develop meaning. Constructivist theory states that learning is an active process where the learner makes links through the materials and prior knowledge (Vygotsky, 1978). It can be summarised simply by the statements ‘knowledge is constructed in the mind of the learner’ (Bodner, 1986, p. 873), and that ‘coming to know is an adaptive process that organises an individual’s “experiential world”’ (Hendry, 1996, p. 19). It has been suggested that the terms ‘weak’ constructivism be applied to the first statement and the term ‘radical’ constructivism if both hypotheses are employed. Radical constructivism is different to solipsism in that interactions with the world are necessary for new knowledge construction (Hendry, 1996, p. 20). This process has to be internal to the learner, by integrating knowledge into pre-existing schemes (assimilation) or by changing the existing schemes to fit the new environment (accommodation). In order to be effective, a task must be included in an ‘actual context’ using ‘real world’ examples and a collaborative context. This allows learners to
understand the motivation and the final goal of the task, and also to incorporate other learner’s opinions.

The work of constructivist theorists, primarily Piaget and Vygotsky (Piaget, 1999; Vygotsky, 1978), outlined the two main constructivist learning models, those of individual and social constructivism. Piaget was an epistemologist who studied thought in children to answer the question of how knowledge is acquired. He differentiated between physical, logico-mathematical and social knowledge. Physical knowledge is embodied in the fact that a glass breaks when it is dropped. Logico-mathematical knowledge is made up of the relationships between things, such as comparing how a squash ball and a tennis ball bounces. Social knowledge, is based on social conventions, for example, tennis classes take place on a ‘Saturday’ (Bodner, 1986). Piaget also distinguished between cognitive functions and cognitive structures. Cognitive functions such as organisation and adaptation remain constant throughout development whereas cognitive structures change markedly with increasing age and experience (Bodner, 1986; Piaget, 1999).

Social constructivism states that learners acquire knowledge through collaboration where meaning is seen through multiple perspectives (Young & Collin, 2004). Knowledge can exist in people’s minds as an objective entity which is defined through social interaction (Hendry, 1996, p. 20). The ideas of social constructivism came from Mannheim, and were developed in the works of Berger and Luckmann’s (1966) The Social Construction of Reality and in Lincoln and Guba’s (1985) Naturalistic Enquiry (Creswell, 2009, p. 8). Social constructivism posits that people hold assumptions about the world in which they live and seek understanding of that world. The meanings developed by people are fundamentally subjective – and those meaning are variable, depending on the sum of their experiences and worldview. These meanings lead to a complexity of views of and about the world.

Learning, according to the social constructivist approach, occurs not only with the student’s interaction with the learning materials, but also with the teacher and with the student’s peers, the community of scholars and like-minded students. Social constructivism maintains that knowledge and social interaction go together (Dalgarno, 2002; Young & Collin, 2004). Learners construct their knowledge from these interactions. Fellow students take on a number of roles, they are not only authors and presenters, but also peers, reviewers and active listeners (Dernlt & Motschnig-Pitrik, 2004). Learning through guided discovery promotes active reflection in both student and teacher. In distance education these interactions between student and teacher and fellow students are limited, to the detriment of learning (Fabri & Gerhard, 2000).
Chapter 2: Literature Review

Constructivism has grown in prominence over the past 30 years and is seen as the latest development in cognitivism. It differs from logical positivism in that it states that the world cannot be known directly – it can only be known through the construction imposed on it by an individual’s mind. How we develop meaning, according to constructivism, is internal and individual (Young & Collin, 2004).

The information transmission approach to learning has had a large impact on information technology and how it has developed. This is a form of constructivism that views learning from an information perspective – it states that learning arises out of how data is coded and transformed by people attempting to make sense of their world and their experience of it (Dillon, 2004). The information transmission approach refers to what Kolb calls comprehension as categorisation, and defines the difference between perceptual categorisation and conceptual categorisation. On their own, the coding of perceptual categorisations gives only a fragmentary classification of immediately available sensory inputs. Conceptual categorisation is the coding of sensory data through non-local stimuli coming from memory and experience (Dillon, 2004). Perceptual coding of sensory data invokes prior knowledge. This means that perception categorisation is never wholly free from context, and the prior life experience of the perceiver.

Examining the constructivist approach raises an important question: if meaning is constructed in the mind of an individual, how do groups of people appear to share knowledge? The key to answering this question is that knowledge must ‘fit’ reality. Knowledge construction is a process in which knowledge is built and continually tested against the environment. Individuals cannot construct just any knowledge. Knowledge therefore must fit a series of constraints that our physical environment provides – concepts, theories, ideas and models are constantly being checked against reality as a result of our experience of the world (Bodner, 1986, p. 875). This is one of the reasons why communication with peers is so important in e-learning, students should be given the opportunity to test newly formed theories with their cohort, to legitimise the learning. Pattern design solutions incorporating the social constructivist approach would accommodate knowledge sharing and legitimisation.

Social constructivists claim that it is not enough to point the students in the direction of readings, interactives or activities. They need mentoring by lecturers or teachers; they need to be able to socialise and interact with their fellow students; they need to be embedded in a learning environment. Students construct their knowledge through their interrelations with the world, of which teachers are a part. This is why constructivism differs from the objectivist
view of learning – knowledge cannot be transmitted by an instructor, it must be constructed in the mind of the learner (Hendry, 1996, p. 24).

Each person has an individual representation of knowledge and learning occurs when a gap occurs in the learner’s knowledge or an inconsistency arises between the learner’s knowledge representation and experience (Dalgarno, 2002). This inconsistency perturbs the learner. Existing theories become unsustainable when experience of the world creates this inconsistency, and new knowledge must be formed to explain the phenomena. This new knowledge must be qualitatively different to the previous knowledge. Acquiring this new sustainable knowledge requires mental effort and time, but is also satisfying and pleasurable. From birth we ‘discover rapidly the unique satisfaction of creating new knowledge or solutions, either separately or socially in interaction with others in their identical process of inquiry’ (Hendry, 1996, p. 23).

Learners serve a cognitive apprenticeship, which should be managed by employing six teaching methods; modelling, coaching, scaffolding, articulation, reflection and exploration. Modelling, coaching and scaffolding lead to the acquisition of cognitive and metacognitive skills through learners observing, then being supported and coached as they put their new skills into practice. Articulation and reflection allows students to think about problem solving and finally, exploration leads to learner autonomy and problem formulation (Chee, 2004).

2.06.01 Design Principles from Constructivism

Based on these constructivist concepts of learning and interaction between the learner and the learning context, a number of general interaction design principles have been identified when applying constructivist approaches to e-learning design and are described in Table 2-2.

Pedagogical interactions via synchronous and asynchronous communications, as proposed in Table 2-2, raise different issues for users. Synchronous interactions cause problems with scheduling, finding a common time when participants are available, and also take considerably more time than anticipated in initial meetings. However, studies show ‘students also indicated that they appreciated the ability to engage in social and relationship building discourse during the synchronous collaborative sessions’ (Bower, 2007, p. 143). This social interaction is important for additional learning opportunities afforded by social constructivist theory. The synchronous approach also appeared advantageous to the teacher, allowing the process of students’ concept formation to be revealed (by inspecting the text chat or session recordings), not just their final product (Bower, 2007).
Chapter 2: Literature Review

Table 2.2 Linking design tools and techniques with constructivist principles

<table>
<thead>
<tr>
<th>Constructivist Pedagogical Principles</th>
<th>Design Principles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Choose materials that allow interactions with ‘real world’ examples.</td>
<td>Use ‘real world’ environments such as virtual laboratories, simulations using real data and live data feeds. Use examples using actual people and their experiences that students can relate to.</td>
</tr>
<tr>
<td>Assign tasks that involve joint construction and social verification of knowledge via collaboration. Provide scaffolding for correct extraction of principles (Linn, Davis, &amp; Eylon, 2004).</td>
<td>Design for collaboration and ease of access to collaborative tools, both on menus and through links within the content. Create synchronous, asynchronous, public and private modes of communication. Allow access to social media.</td>
</tr>
<tr>
<td>Allow a community of scholars to organise. Allow for social interaction and community building as well as on-topic communication.</td>
<td>Use tools such as bulletin boards, chat, wikis, video conferencing and social media.</td>
</tr>
<tr>
<td>Create learning through guided discovery.</td>
<td>Create highly interactive exercises using staged delivery of content (e.g. levels). Allow progress once learning objectives or sequencing is completed. Provide scaffolding with contextual help, links to FAQs and tutorial assistance.</td>
</tr>
<tr>
<td>Facilitate as much tutor to student contact as is practical to allow mentoring, scaffolding and coaching.</td>
<td>Allow student-tutor contact via a public facing tool (such as a forum or wiki) or privately via messaging, Skype (video chat), or e-mail. Create contextual navigation to online student-tutor communication.</td>
</tr>
</tbody>
</table>

The asynchronous nature of Wikis offers the potential to support the development of more reflective and critical thinking. Wikis enable students to add posts in their own time and reflect on the posts made by others. Disadvantages became evident when concepts raised by earlier posting students were the same that later posting students wished also to raise (Bower, 2007). This asynchronous approach lends itself to the reflective approach outlined in experiential learning theory.

When used in conjunction, constructivism and experiential learning theories form a valuable partnership that offers a balance between descriptive and normative theories of learning. Experiential learning describes what Kolb believes happens when a student learns something, whereas the cycle outlined in Chee’s work (2004), based on constructivist theories, describes what should happen when an instructor teaches. When used in conjunction with a pattern language to inform design, the resultant e-learning courseware will encapsulate and integrate the learning theories and will result in an improved e-learning design process.
Embedding pedagogical principles in the design of e-learning courseware would facilitate and validate the design decisions made by the designers. Any learning theory can motivate the structure and content of e-learning design, but must be done in a way that eases the complexity of the design decision making. If these theories are firmly placed within the design process by embedding them within design patterns, the design process can be assisted. Similarly, if a methodology for doing so is standardised and tested, any resultant pattern language can be extensible.
Reference List:


Chapter 2: Literature Review


