Creating a Role for Performance Support Systems in Teacher Education

MARTYN WILD
Edith Cowan University, Perth, Australia

ABSTRACT This article describes an approach to the design of interactive hypermedia in tertiary education that is focused in a perspective which contextualises Performance Support Systems (PSSs) as cognitive tools. The software described is specifically aimed at student teachers and is conceptualised to provide cognitive support in the performance of a complex task – the planning of lessons. In providing this support, the software needs to possess a range of characteristics that are closely related to what we know about cognitive processes, particularly in relation to the performance of non-simplistic tasks in the context of real-life or authentic situations. Whilst this article is intended to present a view as to why PSSs might be of interest to tertiary educators, particularly with reference to the notion of computers as cognitive tools, it does so by considering a specific case – the design of the Lesson Planning System (LPS). Furthermore, although this article originates in a wider research and development project containing a substantial evaluative element, it is the nature, purpose and conceptualisation of use of the LPS that is largely discussed here.

Introduction

The rationale for the development of new approaches to the design of educational software lies in the paucity of effective instructional design models or approaches for the development of instructional hypermedia products in education, particularly tertiary education. The nature of the research reported in this paper is to hypothesise the value of using a model of instruction, based primarily in the theory of cognitive tools and in the design methodology of (electronic) Performance Support Systems (PSSs), but also taking appropriate account of other cognitivist principles in
instructional design, such as mental models, situated cognition and authentic learning.

The concern to provide learners with powerful cognitive tools with which to think has eclipsed the importance of providing similar tools for teachers, particularly for novice or beginning teachers. This paper describes a research project which seeks to redress the balance, and provides an example of how PSSs can be used by teachers to express and extend their thinking in a complex domain.

A basic premise to the development of the Lesson Planning System (LPS) is that it provides a structured environment within which student and beginning teachers are able to design lesson plans for immediate implementation and also receive instructional support in the design process. By engaging novices in the process of designing materials that impact directly on their teaching, it is intended to provide for deeper processing of a complex task, resulting in a more complete understanding of the domain. Indeed, it might be suggested that this is essentially the role and purpose of all cognitive tools (Jonassen, 1994; Jonassen, 1995).

Performance Support Systems

A Performance Support System is interactive software that is intended both to train and support the novice user in the performance of tasks. Raybould describes a PSS as a “computer-based system that improves worker productivity by providing on-the-job access to integrated information, advice and learning experiences” (Raybould, 1990).

There exist slightly different perspectives of PSSs, each moulded by small shifts in emphasis; for example, Barker & Banerji stress the problem-centred focus of PSSs (Barker & Banerji, 1995), whilst McGraw characterises PSSs in terms of their facilities, noting their integration of AI technologies, hypermedia and computer-based training (McGraw, 1994). PSSs can also be described in terms of the uses made of them – that is, in addition to their role in instructing and supporting novices, they might be used by those more experienced in the focus tasks to increase efficiency and quality of output, for example by serving as amplifiers of experience and knowledge (Gery, 1991). Traditionally, however, PSSs have been characterised by their structure and the software resources they provide; that is, they comprise hypermedia reference and instructional sequences, together with open-ended and/or closed software tools, and context-sensitive supporting information (Gery, 1995b). In some cases, they include intelligent tools, such as an intelligent adviser or coach (Carr, 1992; Leighton, 1996; Winslow & Bramer, 1994).

Such software have been developed in training situations in medicine (e.g. medical diagnostic systems), engineering (e.g. computer-assisted design systems) and management (e.g. decision support systems). More recently, the
The concept of performance support has been applied to mainstream and generic software tools, such as Microsoft Excel (version 7) and Microsoft Word (version 7), specifically in the form of ‘wizards’—step-by-step, task-related, procedural tutors. Furthermore, the nature of supporting functions in current and future designs of PSSs has been reconceptualised by Gery (Gery, 1995a) to allow for increasingly diverse applications and types of performance support functions of PSSs. In this latter context, we are witnessing the use of different names applied to describe essentially the same concept—for example, “integrated performance support” (Winslow & Bramer, 1994); as well as increasing interest in applying intelligent tutoring modules, such as those advocated by McGraw (1994) in the form of intelligent advisors or coaches. Leighton has conceptualised this latest development in PSS design as part of a wider movement to replace people resources in more general systems development (Leighton, 1996).

However, these later developments in the design, application and theory of PSSs have not altered their main purpose, which is, quite simply, to facilitate satisfactory or improved performance of a task by someone with limited experience and training in such a task, by providing so-called ‘just-in-time’ resources (i.e. instructional and performance resources). Moreover, PSSs, as well as the supporting functions found in more sophisticated mainstream generic software tools or applications, are more often applied to simple tasks (in the sense of the task being well-defined, well-understood and procedural in nature), rather than complex tasks. In applying PSSs to complex tasks, it is argued that both instruction and performance support functions need to provide for higher-order learning, and particularly for transfer of knowledge. Again, this is fundamentally different from the traditional nature and purpose of PSSs, which are concerned with tasks characterised by training in systems use, whether in a business or software engineering sense (Raybould, 1995). It is worth noting that the uses and types of PSSs are likely to diversify in their future manifestations, when the nature of any particular PSS will be defined largely in terms of its target application, its users and the related domain.

In this light, a PSS has been developed for use by first-year Education students. This PSS is intended to facilitate the development of students’ learning and their performance skills in the area of lesson planning. The basic premise underlying the development of the LPS is that it provides a structured environment within which student and beginning teachers are able to design lesson plans for immediate implementation and also receive instructional support in the design process. Moreover, the LPS can also be conceptualised as providing an environment for cognitive apprenticeship, where the user might engage in learning knowledge and skills that reflect the way in which that learning will be useful in real, professional, life—in other words, where learning is anchored in authentic cognitive activity.
(Brown et al., 1989; Collins, 1989; Merriam, 1993; Resnick, 1987; Cognition and Technology Group at Vanderbilt, 1990).

To date, instructional materials based on interactive technologies have tended to focus on only the instructional aspect of task performance (Brown, 1991; Jih & Reeves, 1992). It is contended that students’ use of the LPS will facilitate the transfer of effective cognitive strategies from the point at which they are learning about the lesson planning task, to the point where they are successfully performing that task, thereby minimising the distinction between ‘learning’ and ‘doing’, and improving students’ lesson planning performance.

Cognitive Tools

Cognitive tools are computer-based applications that are normally used as productivity software. However, these applications may also function as knowledge representation formalisms that require learners to think critically when using them, to represent content being studied or what they already know about a subject (Jonassen, 1995).

In an extensive discussion of the value of cognitive tools, Jonassen describes how conventional applications, such as spreadsheets, databases, expert systems etc., might become intellectual partners and serve to expand and amplify the thinking of learners, engaging students as knowledge constructors rather than information processors (Jonassen, 1995). In one sense, Jonassen is telling us nothing that is particularly new - Briggs, Nichol, Dean, and others of the ‘Prolog education community’ (Nichol et al., 1988) have long sought to provide learners with a range of cognitive tools for the representation and exploration of knowledge (Briggs et al., 1990; Dean, 1990; Nichol et al., 1988). Furthermore, various research and development teams have similarly been involved with the provision of cognitive tools to engage students in a range of modelling environments, so that they might represent and manipulate knowledge according to various formalisms (Cox & Webb, 1994; Mellar et al., 1994; Webb, 1994). Whilst the learning theories underpinning our understanding of the value of such cognitive tools are reasonably robust and cognitivist-based, the use and value of cognitive tools has, of late, been shown to owe much to mental models theories, particularly to that of Johnson-Laird (Johnson-Laird, 1983; Wild, 1996b).

There is a sense in which the use of applications software as cognitive tools takes us beyond the intended uses of such software, so that they can be seen to be functional outside of their original design. This is also true of PSSs – as applications software, these can be used by students as cognitive tools to express and extend their thinking in a complex domain. However, unlike applications software, PSSs are task-specific; they also possess a series of functions and resources to simultaneously engage the user in both
the performance and learning of this task – in this sense, their use blurs the
distinction between task performance and task learning, and can be
conceptualised as offering cognitive apprenticeship to the user. Such a
conceptualisation is derived from an understanding and application of
situating cognitive activity in authentic contexts; in this sense, it is
suggested that learning and doing, or performing, do not exist
independently of the activity in which they occur (Resnick, 1987), and that
learning takes place in situ (Brown et al, 1989; Collins, 1989; Cognition and

The implications of this for the design and use of the LPS are that
novice student teachers can be expected, in their use of the LPS, to improve
their performance skills in lesson planning and also to increase their
understanding, their cognitive learning, in this domain. Thus, it could be
hypothesised the students will not only improve their actual performance,
but also their learning, in this task.

Practical Considerations in Designing the LPS
Brown (1996) defines an electronic performance support system as “a
software environment that provides a context within which work is done.
Everything needed to do the job – information, software, expert advice and
guidance, and learning experiences – is integrated and available, resulting in
improved worker productivity and minimal support and intervention by
others” (Brown, 1996, p. 5).

There are a number of considerations to make in designing the LPS so
that it might function as a PSS in the manner described by Brown (1996)
and others, who reflect a consensus in much of the literature that seeks to
describe the components of a PSS (Gery, 1991; Gery, 1995a, 1995b; Ladd,
1993; Raybould, 1990; Raybould, 1995). These considerations include:

- electronic support for job tasks;
- integration of performance and support functions;
- support on demand;
- appropriate use of technology.

The first consideration is that it should provide electronic support for the
task of lesson planning. In the LPS, such support is provided in the form of
explanatory help (e.g. explanations as to how to go about the procedural
aspects of planning lessons); demonstrations and descriptions (e.g. of how
experts go about the task of lesson planning); customised templates (e.g. of
lesson plans); and a number of databases (e.g. of verbs to use in writing
lesson objectives). In these ways, support in the LPS is both conceptualised
and implemented to provide an instructional framework for use in the task
of creating a lesson plan, comprising: descriptive or declarative information
(e.g. “a lesson plan consists of learning objectives, processes and evaluation
...”); explanatory information (e.g. “it is necessary to evaluate a lesson to
determine how we might improve later lesson plans, and to measure the level of success in this one ...”); and procedural information (e.g. “to create a lesson plan you need to complete four steps - describe your learning objectives, work out the best way of meeting these objectives ...”).

Secondly, the LPS needs to be integrated in the work environment so that the task and the PSS are tightly linked. In the LPS, this is achieved by users being able to move freely between both performance and support functions within the LPS operating environment. A more tightly integrated PSS might provide for partial or fully automated error trapping, to detect inconsistencies in any of the data being provided in lesson plan designs. For example, it might be that there are identifiable inconsistencies between (types of) lesson objectives and lesson evaluation strategies devised in a single lesson plan. However, such detection would necessitate the application of software intelligence in the manner described by Self (1990) to model some form of cognitive reflection, and it is currently not feasible to design technology applications to facilitate complex cognitive functions such as reflection, directly - this requires the development of sophisticated intelligent tutoring systems (Self, 1990).

Thirdly, any support provided in a PSS needs to be made accessible at the time of need - a concept often referred to as ‘just-in-time’ support (Brown, 1996; Geber, 1991). In the LPS, such support is provided as information directly related to the task being undertaken and in a format expected by the student. For example, this might be a sequence of instructions to support the completion of a procedure; or it might take the form of a database of possible objectives for selection and placement in a lesson plan.

Finally, the appropriate use of technology is provided for in the LPS so that its suite of functions may operate on a standard desktop or laptop computer (Apple Macintosh, running operating system 7.1 or greater, with 4 mg of RAM). The technologies in the LPS are presently focused on hypermedia-driven informational support and performance tools. It is likely that further enhancements of the LPS might be towards offering greater information currency where the user may link, on-line, to a wider range of relevant information using distributed information networks available via the Internet. It may also offer multimedia information.

The Context of Design for the LPS

At the core of the LPS is a model of lesson planning required by Edith Cowan University and wider afield. This model includes essential components of lesson planning such as writing learning objectives, developing learning experiences and planning evaluation (Barry & King, 1993). Each component is supported by activities that instruct the user about the task (e.g. provision of information relating to reasons why
objectives are necessary, criteria for quality objectives) and which also assist
the user in performing the task (e.g. provision of a database of verbs to
assist in writing quality learning objectives). A set of software tools is
available to support each activity. For example, one of these is a tool
designed to engage students’ reflective thinking and aimed at providing
them with the ability to evaluate the effectiveness of their completed lesson
plan. This tool functions by prompting students to analyse and reflect upon
the appropriateness of evaluation processes set in relation to lesson
objectives.

It has been suggested that cognitive processes such as reflection can
now be provided for in a computer by applying the use of intelligent
advisors or coaches (Carr, 1992; Leighton, 1996; Winslow & Bramer, 1994).
However, despite inroads into the development of these technologies, as
described by Self (1990), it apparently remains not possible to design
technology applications to facilitate reflection directly. Indeed, where one of
the expected outcomes of PSS use is student learning (such as in the LPS),
it is probably not desirable to conduct high-level cognitive activity in the
computer in place of this cognitive activity in the learner. Furthermore,
present PSS-related technologies can mediate and encourage reflection in
the student in several ways, such as providing a communication link
between learners, providing tools for knowledge and outcome
representation during activities (Hedberg et al, 1994), or simply displaying a
record of the learner’s activities (Schauble et al, 1993).

The lesson planning process is viewed in the LPS as an exercise in
problem solving. An important factor in solving problems is domain-specific
comprehension, where Glaser has suggested that one of the features
distinguishing a novice from an expert is the incompleteness of the novice’s
knowledge base, rather than limitations in their processing capabilities
(Glaser, 1984) and that the transition from novice to expert performance is
largely provided for by the acquisition of a suitable knowledge base (Glaser,
1982).

A knowledge base consists of both descriptive and heuristic
components – descriptive knowledge is the shared knowledge of experts and
practitioners that is usually found in textbooks, while the heuristic
component includes the knowledge of good practice and judgement
constructed over years of experience. It is suggested that the description of
expert performance should include two related aspects: the information
structures and declarative knowledge that are required for performance, and
the cognitive strategies and procedural knowledge that are required by the
task. It is in these structures that the instructional and the tool-based
knowledge is provided for in the LPS. Figure 1 demonstrates how separate
windows were used to achieve the integration of support functions in the
LPS.
Lesson planning is an essential cognitive skill for teachers. Effective lesson planners possess declarative knowledge about themselves as planners, about the task of lesson planning and about ways of going about the task. They also possess domain-specific knowledge such as the criteria for creating instructional objectives, the most appropriate strategies to achieve particular objectives and the range and relevance of evaluation techniques. They know how to plan lessons in the appropriate way, what is required of them when planning a lesson, and they know when and why to perform particular aspects of lesson planning. In addition to this knowledge, they have the skills to regulate their own performance, checking and monitoring to ensure they are meeting certain criteria. They also possess the skills and knowledge to allow themselves to correct errors. The LPS is intended to provide a set of scaffolds and structures by which the novice lesson planner can bring to bear these same expert skills in performance in lesson planning in a way that is typical of experienced teachers.

Theoretical Issues in the Design of the LPS

There are a number of theoretical considerations that have been made in the design of the LPS. These are described below.
There is an assumption made in the design of the LPS that there is implicit value in the development of information or knowledge in hypermedia structures. That is, much of the cognitive value of hypermedia is directly attributable to the structure governing its application – a semantic or associative network of interlinked information, distributed across a range of media (i.e. sound, graphic, animation, or video). For example, hypermedia information structures allow for the chunking of information, a feature that, in the light of information-processing theories about working memory, might be seen to support the cognitive processing of knowledge (Biggs & Moore, 1993). There have also been suggestions that in providing for browsing and thematic exploration, hypermedia information facilitates higher-order cognitive processes such as transfer and knowledge application (Jacobson & Spiro, 1995; Oliver et al, 1996); whilst at a more conceptual level, there has always been a case made for hypertext mirroring the ways in which much of human thinking occurs – by association rather than linearly or procedurally (Burton et al, 1995; Bush, 1945; Minsky, 1975).

However, we need to remember hypermedia, or hypertext, is itself only a delivery medium for information or knowledge (Clark & Craig, 1992; Clark, 1994). Hypertext does not possess a single or normative information structure – hypertext documents are created to conform or fit to a structure imposed by their authors. At one extreme, this structure might be highly ordered, supported by a constrained and sequential set of links; whilst at another extreme, the hypertext may be nonsequential and supported only by referential links (see Figure 2). In many cases, a coherent hypertext document, such as a World Wide Web site, might comprise a mix of these
structures. It is, then, the nature and application of these structures that
determine the effectiveness of engagement with knowledge carried in
hypermedia or hypertext. Furthermore, to maximise engagement the
knowledge needs to conform to a structure that best fits or suits both the
type of knowledge being conveyed and the objectives set by the author for
the types of interactions a user should have with it. As Jonassen (1990)
points out:

Few designers of hypertext believe that hypertext knowledge bases
should be unstructured and totally nonsequential so that users would
have no guidance about the information they access. Even Nelson (1981)
concedes that totally nonsequential hypertext can be disorderly and
could lead to “idiosyncratic and exceptional forms of connections”.
Nonsequential hypertext also results in navigation problems (getting lost
in hyperspace), as well as integration and synthesis problems. (p. 85)

This is the context, then, in which any hypermedia or hypertext information
structure needs to be designed. Concern has to be taken to represent the
various knowledge types in appropriate structures and to build into these
structures sufficient scope for the desired learner-material interactions.

Oliver (1996) characterises the application of hypermedia structures to
learning environments as a continuum, where at one extreme the
hypermedia structure of interlinked information is a linear one with
information nodes connected in a specified and hierarchical fashion; whilst
at the other extreme, information nodes are associated through a referential
structure (Oliver & Omari, 1996). Thus, if we superimposed issues of learner
control over Oliver’s continuum of hypermedia structures, at the former
level learners would have only minimal control – that is, they would be led
through sequences of highly structured information. However, at the latter
level, learners would be free to choose their access of information, limited
only by the number of referential links engineered between information
nodes. Indeed, Oliver and others have extended the association by aligning
this continuum of hypermedia structures with one describing levels of
knowledge acquisition or cognitive activity (Jonassen et al, 1993; Oliver,
1996). So, for example, where learners are intended to acquire low-level
knowledge (i.e. factual statements, rules, procedures) or engage in low-level
cognitive activity (rehearsing, identifying, matching), this is best achieved in
a linear, highly organised hypermedia structure; and where learners are
intended to acquire higher-level knowledge (i.e. abstraction, transferability,
understanding) or engage in higher-order cognitive activity (i.e. reflecting,
predicting, imagining), this is best achieved in an unstructured or referential
hypermedia framework.

Thus, the most notable, if not the most distinguishing feature of
interactive hypermedia software in terms of its educational significance, is
this facility to provide for non-hierarchical representations. Interestingly, it
was those working with knowledge representation tools who, looking for a
theoretical framework in approaches to learning, initially suggested that computer-based semantic representation of knowledge perhaps best mirrored the behaviour of certain higher-order cognitive activities (Nichol, 1988; Nichol et al, 1988) – a suggestion that finds a basis in Minsky’s theory of cognitive frame representation (Minsky, 1975) and in mental models theories (Gentner & Stevens, 1983; Glaser, 1984; Johnson-Laird, 1983, 1993; Wild, 1996b). Of course, even if one accepts this premise, it does not automatically follow that using hypermedia structures for knowledge representation will result in better cognitive representations on the part of learners.

Modelling

The LPS is a cognitive tool that encourages problem solving through modelling, that is, the building and exploring of qualitative models. In this sense, users of the LPS are encouraged to create models of lesson plans and to explore, test and refine those models.

Modelling is an essential component of cognitive activity, of thinking, and for Craik, the originator of the concept of mental models, thinking is concerned with the organisation and functioning of mental processes and representations (Craik, 1943; Johnson-Laird, 1993). It follows that cognitive tools must necessarily provide for modelling activity; that is, they must provide the means by which learners can construct, manipulate and evaluate representations of knowledge. The modelling environment needs to be accurate and structural but not necessarily complete, enabling learners to move from their own mental representations of lesson planning to the conceptual model of that process required by an expert. In this process, novices will be able to construct a deeper understanding of a complex domain.

It is generally agreed that although a modelling environment should not be complete, it is important that it remains functional; that is, it must provide the learner with some expert knowledge and facilitate learner predictions (London Mental Models Group, 1988; Mellar et al, 1994; Wild, 1996b). It is the incompleteness of the model that provides the opportunity for construction, reflection and change. In this sense, the LPS provides an environment for learners to externalise their own understanding of the lesson planning process, to identify inaccuracies or insufficiencies in their thinking and to reflect on their cognitive models without expressing a commitment to any one in particular.

Indeed, it is known that mental reasoning (propositional, relational and quantified reasoning) involves the construction and evaluation of a number of possible models to suit particular interpretations of premises to an event, before making a final inference or conclusion (Johnson-Laird, 1983; Johnson-Laird & Byrne, 1991). Since the limitation to inferential processing

279
is the capacity of working memory, the greater the number of models needed for an inference, the harder that inference will be (Sweller & Chandler, 1994). Furthermore, learners will sometimes fail to construct all possible models for a given event – if they arrive at a conclusion that fits their available beliefs, they will tend not to search for others, with the consequence of overlooking the correct conclusion (Johnson-Laird, 1993; Johnson-Laird & Byrne, 1991). Also in this context, learners may construct mental models based on seemingly analogous experiences which may compound the construction of misconceived models (Jih & Reeves, 1992). Thus, by providing cognitive tools in the computer, it is possible to provide the necessary means for learners (in this case, student teachers) to externalise their thinking and consequently create strong and accurate models that otherwise might prove elusive.

**Cognitive Load**

The greater the availability and accessibility of information within a given computer environment, the more likely users will flounder as a result of excessive cognitive load or cognitive overload and consequently fail to learn. According to Jih & Reeves, learners using a hypermedia system must cope with and integrate three types of cognitive load: the content of the information, the structure of the programme and the response strategies available (Jih & Reeves, 1992). How learners cope with such a load depends largely on the human-computer interface. For example, cognitive load can be reduced by:

- reducing the number of options at any one point in the programme;
- by encouraging users to externalise their thinking, by use, for example, of text annotations and place-marking;
- by ‘hiding’ programme options not likely to be needed by most users;
- by providing strong visual cues to aid navigation;
- by reducing the number of hypermedia links between information nodes (Oren, 1990).

The means by which users deal with the cognitive load imposed by the LPS are expected to be largely a function of their conception of the lesson planning task, as well as that of the software interface. Certainly, software features such as on-line help (i.e. help, for example, in planning the task) and dynamic structure maps (i.e. maps to show a user’s position in the hypermedia environment at any one point), are included in the design of the LPS to encourage learners to build strong conceptualisations, or mental models (Jih & Reeves, 1992).
Learner Control

Learner control is a reference to that dimension in computer use that describes the level of control exercised by the learner when interacting with a given software item. Despite the fact that learner control has been one of the most heavily researched dimensions of computer-based education in recent years (Steinberg, 1989), Reeves has pointed out that many of the research studies are flawed both in their theoretical and methodological bases (Reeves, 1993). It seems to be popularly assumed that the greater the control exercised by the learner (as opposed to that exercised by the software) within a given software environment, the greater the level of learning will be. This assumption is undoubtedly a product of cognitivist learning perspectives, and is closely related to the following fundamental premises:

- learners are active processors of information;
- knowledge is more likely to be successfully constructed when learners have control over the learning process (Rowe, 1993).

However, what evidence we do have about learner control is at best contradictory and at worst negative (Reeves, 1993; Steinberg, 1989). In particular, Oliver draws attention to research that suggests that unskilled learners fare especially badly in terms of performance outcomes when the degree of learner control is high and external control (e.g. control by the programme) is low (Oliver, 1994).

The LPS provides for significant learner control over a range of learning processes, including task perception, information retrieval and processing, problem-solving, knowledge construction, revision, reflection and cognitive modelling.

Transfer

The development of the LPS supposes that students, by using the LPS, will come to understand the processes involved and be able to plan lessons effectively, both through their use of the LPS and also by other means (e.g. pen and paper). A significant finding in the transfer of learning research is that where there are common factors in the content or procedures in carrying out two tasks, transfer is more likely (Child, 1981). To facilitate transfer of learning, or as in this case, transfer of learned performance, the metaphor that guides the design of the human-computer interface is provided by traditional lesson planning; the LPS environment in which students plan their lessons makes use of identical terms and elements to those encountered in the paper and pen approach (Barry & King, 1993). It is expected that students will undertake the performance aspects of the lesson planning task using similar methods, whether they are working with
the LPS or pen and paper. Furthermore, the amount and type of human-computer interaction expected by use of the LPS (for both the performance and supporting functions of the LPS) are intended to approximate to that between learner, lecturer and other supports (e.g. information sources) in a traditional context.

**Stages and Decisions in the Design and Development of the LPS**

There were three distinct stages in the design and development of the LPS:

- Identification of desirable features in the LPS.
- Development of these features as components in a coherent (instructional design) model.
- Evaluation of the LPS to determine the behaviour of the features.

**Stage 1: identification of desirable features in the LPS**

This was achieved by two approaches: the first involved using relevant literatures to help predict the cognitive processes that are necessary for the completion of a complex task, such as lesson planning, and then outlining the nature of the software tools and information resources that might best support these processes. The second approach focused on a round-table discussion using three experts in lesson planning (i.e. lecturers who are currently teaching lesson planning to Edith Cowan University undergraduates), to collectively determine the most efficient ways and means of creating lesson plans, together with identification of the shortcomings in lesson plans presently constructed by first-year (novice) undergraduates at Edith Cowan University. This second approach resulted in the identification of a set of procedures by which effective lesson plans might be created by novices in this domain, together with the information resources necessary to support their creation. This meeting was scheduled for approximately one hour and data was collected in the form of a transcription of an audio tape recording of the meeting.

These two sets of data have been used to determine the features or components of the LPS necessary for or best suited to the task of lesson planning by novice students in this domain. Extrapolations from the data revealed a preferred model of the lesson planning process that needed representation in the LPS. This model can be seen as a process that focuses on five questions. They are given below, together with statements which explain their meaning and, in some cases, their original context in the experts' discussion.

**What Background Facts Need to be Considered in Planning this Learning Experience?** The student teacher needs to determine the context to the
topic or theme to be taught, as well as the abilities, needs, interests, skills and understandings that students will bring to this lesson.

What Should the Students Learn as a Result of this Learning Experience? The student teacher should identify a wider goal (i.e. expressed perhaps as an intent or aspiration) as well as specific objectives for the lesson. Objectives are best stated as what learners should be able to do, or do better, as a result of having worked through this lesson (Rowntree, 1990). The experts’ discussion revealed a range of possible objective types, from general through to specific, but favoured the need for beginning or novice student teachers to describe objectives written in terms of observable learner behaviours or learner performance, under specified conditions and within stated parameters. For example: “Working in a group of three, describe three different ways you can get to school” – this objective specifies conditions (i.e. working in a group of three), the expected behaviour (i.e. describe different ways you can get to school) and parameters of that behaviour (i.e. three different ways).

The experts’ discussion also revealed a discrepancy between statements of learning objectives as descriptions of observable learner behaviours or performance, and learning objectives given as descriptions of ‘outcome statements’. Outcome statements, however, were still taken to be descriptions of behaviours, but on a predetermined continuum of development, often originating in national or state curriculum statements.

Objectives can be of three different domains – cognitive, affective or psychomotor – and within each of these domains be at a specified hierarchical level of performance (Bloom et al, 1956; Harrow, 1972; Krathwohl et al, 1964). For example, in the cognitive domain there are six levels of performance, from knowledge (lower-order) to evaluation (higher-order) (Bloom et al, 1956). These six levels can be used to organise corresponding levels of verbs that might be used to invoke appropriate descriptions of specific behavioural objectives (for example, see Table I), as with those provided for use in the LPS.

It is clear from the experts’ discussion that it is not always possible or desirable to express learning objectives in behavioural terms – for example, when planning for a learning experience that is entirely creative, or one that is expressive or exploratory and should not have delimiting or restrictive operators on the scope of the experience. However, in the final analysis, it would seem that a prime characteristic of expertise in lesson planning is knowing when to apply learning objectives that are behavioural and when to use non-observable or less precisely stated objectives.

What Knowledge, Concepts or Skills Have to be Covered in the Learning Experience? This refers to the sequencing of instruction, in terms of the underlying structure of the content material of the lesson. There are three clear guidelines to the development of answers to this question:

- start planning with students’ prior knowledge or previous learning;
work from the concrete to the abstract when structuring the content to be learned, particularly for younger children; break the content into discrete yet related ‘chunks’ or smaller parts, to allow more easily for mastery learning (Biggs & Moore, 1993; Wild, 1996a).

What Experiences will Help Students Learn in this Domain or Subject? Here the student teacher is intended to consider the strategies (i.e. the organisation of the learning experiences) that are to be devised to meet the learning objectives. These strategies would be chosen to suit the learners’ ages, abilities, interests, skills and needs.

Table I. Verbs provided in the LPS, based on Bloom’s (1956) cognitive domain.

How Can I Best Know What the Students Learn as a Result of this Learning Experience? Evaluation might occur before instruction (i.e. as a diagnostic tool), during learning (i.e. as formative evaluation) or after
learning (i.e. as summative evaluation). Each type of evaluation or approach to it can be catered for by the use of various techniques – some of which are described below:

- **Diagnostic evaluation**: use of a standardised test (pre-test); use of student observation over a specified period of time.
- **Formative evaluation**: questioning students about their understanding; commenting on students’ work whilst they are completing the lesson; students’ demonstration of their understanding; students conducting self-assessment.
- **Summative evaluation**: marking completed student work; use of a standardised test (post-test); student interviews; profiling students.

**Stage II: components of the LPS**

As with most PSSs, there are two major types of components to the LPS: the first are support tools; the second, instructional sequences or items. In addition, there is a ‘help’ facility, which can be classified as an instructional aid (see Figures 3 and 4 for a view of the interface that provides access to these elements).
Figure 3. The provision for instruction and tool-based support in the LPS.

The primary difference between tool-support and instructional-support components in the LPS is one of operation. For example, tool-support functions provide dynamic access to information, templates and generic tools (i.e. the Work Pad), to allow users to implement information directly or indirectly into their lesson plans. Alongside and in addition to these dynamic tools is the provision for a standard series of other tools (such as ‘save’ and ‘print’) that allow for the manipulation, in various ways, of the students’ work. Conversely, the instructional information is not primarily intended for students to embed into their work, but rather to inform their lesson planning.

The components provided in the LPS, together with their relationship to desirable knowledge types and their corresponding means of representation, are described in Tables II, III and IV. The components represented in the LPS also correspond to those knowledge types suggested by Brown (1996) and Gery (1991) as being desirable in PSS knowledge base development.

<table>
<thead>
<tr>
<th>LPS component</th>
<th>Cognitive act</th>
<th>Knowledge type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructional support</td>
<td>Learning</td>
<td>Declarative (D)</td>
</tr>
<tr>
<td>Knowledge about the task</td>
<td>Knowledge about how to perform the task</td>
<td>Perceptions of self as lesson planner, i.e. How do I best complete this task?</td>
</tr>
<tr>
<td>Facts</td>
<td>Procedures</td>
<td>Task processes</td>
</tr>
<tr>
<td>Principles</td>
<td>Concepts</td>
<td></td>
</tr>
</tbody>
</table>

Table II. Instructional components and knowledge representation in the LPS.

<table>
<thead>
<tr>
<th>LPS component</th>
<th>Cognitive act</th>
<th>Tool</th>
<th>Knowledge type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tool support</td>
<td>Performing</td>
<td>Reflection</td>
<td>M</td>
</tr>
<tr>
<td>Verb database</td>
<td>P</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Example lesson plans</td>
<td>D</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Work Pad</td>
<td>M, D, P</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Example Objectives</td>
<td>D</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Evaluation</td>
<td>P</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table III. Tool components and their representation in the LPS.

<table>
<thead>
<tr>
<th>Knowledge type</th>
<th>Instructional support</th>
<th>Instructional support</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Think tool</td>
<td>Help tool</td>
</tr>
<tr>
<td>Declarative</td>
<td>Structure of the lesson</td>
<td>What is a lesson plan?</td>
</tr>
<tr>
<td></td>
<td>Objectives</td>
<td>What is a good objective?</td>
</tr>
<tr>
<td></td>
<td>Methods</td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>Preparation</td>
<td>Using the LPS</td>
</tr>
<tr>
<td></td>
<td>Methods</td>
<td>How do I ensure my evaluation will be effective?</td>
</tr>
<tr>
<td>Metacognitive</td>
<td>Ways of writing the lesson plan</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Evaluating my own performance</td>
<td></td>
</tr>
</tbody>
</table>

Table IV. Tool components and instructional support in the LPS.

Stage III: evaluation of the LPS

As an initial evaluation of its operational effectiveness, the LPS was used by four novice student teachers enrolled full-time in an Education degree, whilst in their first year of study, over a period of four weeks prior to fulfilling a course requirement of two weeks’ teaching practice at local primary schools. Use of the PSS was provided to these students both in and outside the university campus. These students were interviewed individually (for approximately 30 minutes), using a semi-structured set of questions immediately following this four-week period, to establish broad patterns in students’

- patterns of usage;
- perceived result of usage; and
- difficulties in usage.

Patterns of Usage

All students revealed an increasing reliance in their usage of the LPS, on the support tools, particularly the Verb Database, example lesson plans and
the lesson planning template, together with print and save functions that were inherent to the software. Correspondingly, a decreasing amount of time over the four-week trial period was spent in the instructional components of the LPS. The reasons given for this pattern of usage were of two categories: the first set of reasons suggested that students quickly absorbed what was required of them by the system to be able to perform the task competently; or that students seemed to understand quickly the concepts in the domain of lesson planning so they did not consider it necessary to return to the instructional components. Interestingly, one student suggested that his understanding of the use of the system was mistaken, and that he had originally set out, before all else, “to look for the test” in the software, being convinced that there would be a test somewhere in the system!

By asking students to expand on these responses, it appeared that at least three out of the four students became aware, over no more than three or four occasions using of the LPS, that they did not need to know much about
lesson planning to perform the task, only “how to go about using the software”. This suggests that students perceived themselves as able to complete the task competently, without having to learn about aspects of the task – in other words, they used the LPS to learn how to perform the task (procedural knowledge) without spending effort in learning about aspects of the task (declarative knowledge). This also implies that these students quickly applied metacognitive strategies to regulate their usage of the LPS, concentrating on using those system functions that enabled them to competently perform that task of generating a lesson plan, without undue recourse to pushing the boundaries of their knowledge to understand aspects of the task. For example, two students described that they were able to generate a large number of lesson plans (i.e. eight to 14) more quickly, by “always starting writing out a new lesson plan with the previous lesson plan”, and only altering those aspects of each lesson plan that distinguished it from others they had previously planned. Indeed, whilst this approach to the use of the system was undoubtedly efficient, it carried with it the danger of reproducing a series of lesson plans, perhaps in one session, with a minimum of appropriate consideration given to all aspects of each – a danger that one student was aware of: “After about the third lesson, I realised that I hadn’t really thought about how I was going to evaluate what I was teaching ... how each lesson should be evaluated perhaps differently. So I went back and made sure I used different techniques for the lessons.”

In the same vein, all students revealed that after three or four occasions of use, they deliberately by-passed system prompts (that are provided in the Reflection tool, and similar prompts automatically brought into play if the system detects the closure of a lesson plan without the student having accessed the Reflection tool) to force them to reflect on their lesson plans – for example, to consider the appropriateness of their evaluation strategies and how well these strategies match lesson objectives. This suggests that students either quickly internalise these kinds of metacognitive processes, or are unwilling to be ‘forced’ by the software system to practise such processes.

**Perceived Result of Usage**

All the students in this evaluation suggested that they now knew more about lesson planning than before (before they used the LPS), despite having been introduced to lesson planning in a lecture as part of their course of study immediately prior to their role in this study. They also all suggested that this was a direct result of having used the LPS (note: two students in this group had missed the lecture given on lesson planning by their course-unit coordinator, and one other student suggested that she “hadn’t really followed what was being said about lesson planning in this lecture”). Also, three out of the four students suggested that they were now competent
lesson planners and would be able to plan a variety of different types of lessons competently, with or without the use of the LPS. The remaining student said that she would still prefer to have the use of the LPS to plan her lessons, “just in case I need to look at the proper choice of verbs to use” (using the Verb Database).

Difficulties in Usage

It was imperative to all the students that they could both save and print their lesson plans. Indeed, whilst these functions are provided for in the LPS, all students here suggested that it was frustrating that they could not print their lesson plans at the point of need (i.e. in or near the classroom in which the lessons were to be taught; or at the point of completion of the lesson plans, when perhaps there was not a printer available). Of course, this is a difficulty in the computer system availability rather than a difficulty with the LPS itself, although the students clearly indicated in this concern that they did not perceive there to be a difference between the two systems – this was a problem that might prevent them from using the LPS as a ‘tool of convenience’ in real-world or authentic situations.

Mention was also made by two students of the system’s tight focus on a predetermined lesson plan format. Students suggested that once they knew how to create competent lesson plans in the LPS, they were interested to consider how they might use other lesson plan templates or formats to provide for different types of lessons. In this context, students are probably referring to the requirement in the LPS to plan lessons following a behavioural model, where, for example, it is a requirement for lesson objectives to be written as observable behaviours in students. This structure was suggested as being appropriate for novice and inexperienced lesson planners by the panel of experts used to advise on the components and features of the LPS. Furthermore, instructional resources in the LPS do describe to students that the LPS provides only one model for planning lessons, and that other models do exist. However, the student teachers’ concerns in this case imply a mismatch between some students’ requirements of the system and the provision in experts’ predictions of these requirements. Indeed, this mismatch might also be an example of a situation already revealed in novice-expert studies, where experts, in some cases, are seemingly unable to appreciate or predict the knowledge structures or knowledge requirements of novices, having long since been removed themselves from a similar situation (Chi et al, 1988).

From this analysis, then, it has since been decided to update the LPS so that it can provide for greater access to more diverse information about lesson planning, particularly different models of lesson planning, and to provide for greater availability of use. To implement both these provisions, it seems appropriate to provide an on-line version of the LPS. This, together
with dynamic links provided to more, and more diverse, information on
lesson planning should increase accessibility at more vantage points, for use
in and outside school classrooms.

Conclusion

This article has sought to describe the design and development of a
cognitive tool for novice teachers, based on a PSS; it has also attempted to
map the notion of a PSS onto various theories of instruction and learning. It
is further implied here that a PSS, implemented within the construct of
cognitive tools, is a valuable model for instructional systems development in
a range of domains relevant to teacher education. Of immediate interest
perhaps is the concept of a more general PSS that encompasses the
professional environment of teaching more completely.

Within the article, issues have been raised concerning the design and
development of cognitive tools such as the LPS, issues which face all of us
interested in putting the power of such tools in the hands of learners,
whether they are children or teachers. However, in this context, it is of value
to note that although design decisions for the LPS have been based on what
instructional psychologies suggest are likely to be effective conditions for
learning, the concept of the LPS, and indeed of cognitive tools in general, is
concerned more with describing how the learner might interact with the
instruction to construct new knowledge. That is, the design of the LPS is
less about creating the instructional conditions that Gagne and Glaser
prescribe as being necessary for learning (Gagne, 1977; Glaser, 1987), and
more about describing and defining how the instructor, the knowledge and
the learner should interact. Indeed, this is much more in keeping with what
Marton & Ramsden and other advocates of phenomenographic approaches
to researching teaching and learning have revealed about effective learning
(Laurillard, 1993; Marton & Ramsden, 1988).

Correspondence

Martyn Wild, Faculty of Business, Edith Cowan University, Churchlands,
Perth, Western Australia 6018, Australia (m.wild@cowan.edu.au).

References

Innovations in Education and Training International, 32, pp. 4-12.

teaching. Wentworth Falls: Social Science Press.


292


294


