

# Hyper-ITS: A Web-Based Architecture for Evolving and Configurable Learning Environment

KINSHUK<sup>1</sup>, ASHOK PATEL<sup>2</sup> and DAVID RUSSELL<sup>2</sup>

<sup>1</sup>*Massey University, New Zealand*

<sup>2</sup>*De Montfort University, United Kingdom*

**Abstract:** The World Wide Web facilitates co-operative teaching and learning, offering an exciting potential for sharing both the learning resources as well as the tasks of creating new learning resources through international collaboration. The mainstay of the web so far, the Hyper-text and Hyper-media offer a link based flexible architecture where addition, amendment and removal of material is relatively easy. This allows incremental construction and upgrading of web based systems to facilitate division of large tasks, easier prototyping and the ability to selectively update various learning resources to keep them current as well as to take advantage of improving technologies.

This paper proposes a web based architecture for intelligent tutoring systems that extends similar benefits to designing interactive learning, particularly with reference to Cognitive Apprenticeship Based Learning Environments (CABLE). Since the acquisition of cognitive skills is critical for learning any applied discipline, there will always be a need for interactive learning environments where the learner can practice to acquire new skills and to improve the grasp over any theoretical learning. Since the current web based learning systems cannot adequately provide for interactive learning that is supported by dynamic feedback, it is important to extend the benefits of the traditional ITS to the web based systems through a modular architecture supported by authoring tools and an indexing mechanism for the repository of the various tutoring components.

**Keywords:** Cognitive Apprenticeship, Interactive Learning Environments, Knowledge Representation Framework, Web-based Education.

## Introduction

The Internet is now becoming integral to all types of education and training. The global learner community includes users of all age and backgrounds with different interests and motivations for using computer networks. Internet has generally been successful in keeping up with these motivations and interests by providing useful information and a greater sense of enjoyment through new ways of cognition. In the current educational environment with an increasing demand on educational opportunities accompanied by constrained resources available to the traditional education sectors, alternative approaches such as distance education are becoming more attractive for academic institutions. Also, as Fischer (1996) pointed out, the earlier notions of a lifetime divided into neat phases of education followed by work are no longer tenable and learning can no longer be dichotomised either spatially or temporally into a place and time to acquire knowledge (school/college) and a place and time to apply knowledge (workplace). Lifelong learning integrates and mutually enriches both the cultures of education and work. The 'new' frameworks for education, therefore, include the integration of working and learning, learning on demand, authentic problems, self-directed learning and collaborative learning among others.

Leigh (1996) observed that the Internet can provide an instantaneous dissemination of information to a wider audience and education at a distance can meet the financial needs of both the institution and the learner often proving a very effective alternative to traditional instruction or training. However, Simonson (1996) observed that the students do not prefer to learn at a distance as they value the presence of an informal interaction that provides *active learning*. In this connection, the Internet may prove more beneficial than the traditional distance learning methods as it has the potential for accommodating far more interactions - involving information in textual, graphical, audio and video forms, in a cost effective manner. The Internet already provides mechanism for asynchronous interactions through discussion forums, newsgroups and email. Synchronous interactions through Electronic White Boards, Web Phone and Web Meeting are also becoming common. In some ways, this medium of instruction surpasses even the traditional classroom as it facilitates communications between academic communities located in different parts of the world.

The Byzantium project, a consortium of six Universities, recognised the growing importance of the Internet as an educational platform but decided that the efficacy of any interactive teaching and learning tools needed to be established on stand alone computers and Local Area Networks before they could be extended to the Internet. This view resulted in the development of four *basic level* Intelligent Tutoring Tools (ITTs) for introductory numeric topics in Management Accounting, under the Teaching and Learning Technology Programme of the Higher Education Funding Councils of the United Kingdom (Patel & Kinshuk, 1996a; Patel & Kinshuk, 1997a). The ITTs, as they became available were introduced into the mainstream teaching across a number of institutions and were well received (Kinshuk, 1996). They have now been used by more than 7,000 students at multiple institutions.

### **Intelligent Tutoring under the Cognitive Apprenticeship Approach**

Anderson et. al. (1996) observed, "While some context will often be required to assess a component, there are always bounds on how complex such a context need be. It is a well-documented fact of human cognition that large tasks decompose into nearly independent subtasks, so that only the context of the appropriate subtask is needed to study its components." They claimed that it is better to train independent parts of a task separately because fewer cognitive resources will then be required for performance, thereby reserving adequate capacity for learning. An intelligent tutoring system can be a useful tool to provide efficient learning by representing the domain content at suitable granularity and providing interactive guidance to the learner at the level of independent parts of the domain tasks.

The possibility of decomposing a large task and the need only for the context of the sub task for learning, makes it possible to learn from much simpler interactions. Intelligent tutoring systems can greatly help in this purpose by presenting various domain tasks through a granular interface. Here, the learning tasks are decomposed into smaller components at varying levels of granularity with the perspective shift enabled through the user interface. There is no need for the system to engage in complex inferencing about user knowledge as the system can provide a simple correct/incorrect feedback at a coarser grain size and advise the student to use a fine grained interface for more detailed interaction. More details and examples are available in Patel & Kinshuk (1997b).

To address the need for intelligent tutoring with a focus on cognitive skills acquisition, the Byzantium cognitive apprenticeship based learning environment applies the Collins, Brown & Newman (1989) framework in the design of its ITTs (see Fig. 1). The framework provides that the following functionality should be present:

- (a) The learners can study task solving patterns of experts to develop their own cognitive model of the domain (modelling). The ITTs provide a Basic Concepts mode presenting textual/graphical explanations and solved examples. The same material is also available through the Help button in the interactive learning mode.
- (b) The learners can solve tasks on their own by consulting a tutorial component (coaching). The ITTs offer qualitatively better coaching through interactive guidance and dynamic feedback while a student is attempting to solve a problem.
- (c) The tutoring activity of the system is gradually reduced with the learner's improving performances and problem solving (fading). The ITTs provide help 'by exception' and the tutoring activity is triggered by an illegal or incorrect attempt. Improved performance will automatically see less tutoring intervention.

While using the interface shown in Figure 1, if a student pressed the Help button, the system will bring up the descriptive explanations dealing with the concept of discount factors. The objectives of the learning environment reflect Collins' (1990) recommendations for constructing robust domain competence. Accordingly, the system aims to facilitate the learners in:

- i) acquiring the basic domain knowledge which can be used subsequently as a base to integrate all the bits and pieces of knowledge gained from specific situations;
- ii) applying the basic domain knowledge in abstract and contextual scenarios to generalise the knowledge and skills to be able to apply them in real world situations.

It was observed in the implementation of early prototypes that the sole use of abstract problems generated by the system resulted in a static interface allowing some students with better graphic memory to map the interface objects by position rather than the underlying concept. Their self-explanation degenerated into, 'the object here is obtained by dividing this object by that object', based on syntactic learning rather than conceptual learning. In the paper based tests such students found it necessary to draw the screen layout before attempting the solution. This observation led to the concern for ensuring better generalisation and far transfer of knowledge resulting into some additional functionalities in the ITTs as well as the recommended framework for their implementation as given in Fig. 2.

The recommended learning path consists of transition through observation, interactive learning, simple testing, learning and testing involving multiple contexts and/or interpretation of rich narrative and finally extending the learning process to integrate the essential tools of trade, such as spreadsheet, database, statistics packages, graphic and design packages. The ITT architecture provides customisation facility for creating appropriate templates. A teacher can specify various parameters to create a replica of simple real world scenarios. The templates can then be used for the purpose of both the structured and non-structured problems. This facility can be used both for providing a highly situated learning through a single context and for helping in generalising the knowledge through multiple contexts.

Based on the experience gained in the development of ITTs, the design and development of Web based Intelligent Tutoring Applications (ITA) has commenced (earlier referred to as Intelligent Tutoring Applets, see Patel & Kinshuk, 1996b; Patel & Kinshuk, 1997c). The ITAs extend the scope of ITTs by providing the same amount of adaptivity and intelligence available in ITTs for individual instruction along with added benefits of wider accessibility through the Internet and distributed architecture to take advantage of frequent and economic updates of domain content, availability of expertise of teachers from any part of the world and collaboration among teachers and students to facilitate the reuse of expertise (Kinshuk & Patel, 1998).

**Net Present Value**      Rate %

	Inflows	Outflows	Net Cashflows	Discount Factors	Present Value
Investment		29000	( 29000)	1.0000	
Year One	12000	1000	11000		
Year Two	11000	1000	10000		
Year Three	10000	2000	8000		
Year Four	8000	3000	5000		
Year Five	7000	3000	4000		
Residual Val	3000		3000		
<b>Project Lifetime Surplus</b>			<b>12000</b>	<b>Net P. V.</b>	

**Calculator**  
 Left Click on a Box to pick the value. Right Click to drop result of calculation.  
 ✕ Per Cent    ^ Raise to  
 / Divide      ✕ Multiply  
 Q : Quit

**Interactive Messages**  
 Sorry ! That isn't correct...  
 Use the Formula feature for better interaction if you want to alter an existing value, please change the Rate % !

**Formula**  
 Explain     $D = \frac{1}{(1+I)^n}$   
 Calculate  
  
 +  ^

**Formula**     $D = \frac{1}{(1+I)^n}$   
 FUTURE VALUE of £ 100 in 3 year's time @ 10% is  
 Year 1 end: £ 100 + (100 \* 0.10) = £ 110  
 Year 2 end: £ 110 + (110 \* 0.10) = £ 121  
 Year 3 end: £ 121 + (121 \* 0.10) = £ 133.10  
 which is the same as £ 100 \* (1 + I)<sup>n</sup>  
 where I is the rate and n is the number of periods  
 Check: 100 \* (1+0.1)<sup>3</sup> = 100 \* (1.1)<sup>3</sup> = 100 \* 1.331 = 133.1  
 PRESENT VALUE of £ 133.1 to be received in 3 year's time  
 is 133.1 \* 1/(1+I)<sup>n</sup> = 133.1 \* 1/(1.1)<sup>3</sup> = 133.1 \* 1/1.331  
 = 133.1 \* Discount factor = 133.1 \* 0.7513 = £ 100  
 Basic concepts on NPV explains this in more detail.

Fig. 1: Design framework of intelligent tutoring tool

### Intelligent Tutoring on the Internet

Duchastel (1992) has provided a good overview of various possibilities of how hypermedia and intelligent tutoring could shake hands and the recent developments in distance education research include the facilitation of intelligent tutoring on the Internet. Nawarecki & Dobrowolski (1993) proposed an intelligent distributed and decentralised (multi-agent) system which allows various students and teachers to interact with each other, while supporting learning intelligently through the use of autonomous agents. Angelides & Gibson (1993) described PEDRO - a hypertext-based intelligent tutoring system for foreign language learning. PEDRO - The Spanish Tutor is designed to assist intermediate level students with their learning of Spanish grammar, by testing their knowledge of regular and irregular verbs. The system adapts to the student's performance in selecting and presenting the incrementally difficult hypercards.

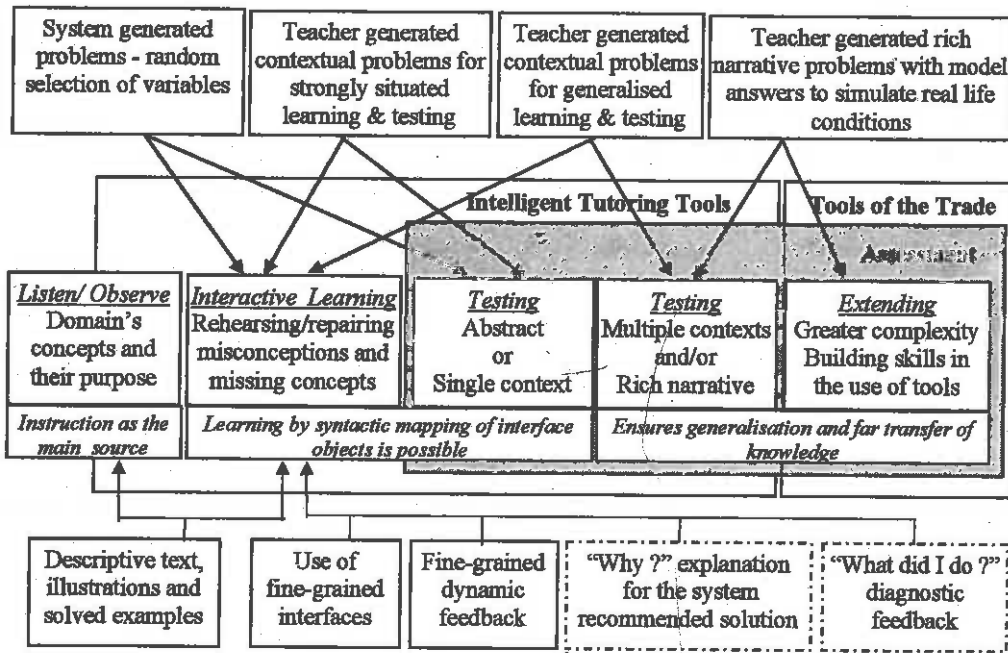


Fig. 2: A cognitive apprenticeship based learning environment implementing ITTs

In general, intelligent/adaptive hypermedia systems adapt the content of hypermedia page and the links from a page to related pages. They keep a learner model between different sessions and use it to provide adaptive teaching sequence and material presentation (Brusilovsky et. al., 1996). There are many systems available for teaching and learning which adapt to the users' requirements and provide customisation of domain content and tutoring strategies for better knowledge acquisition. Most of these systems use concept hierarchy based knowledge representation where domain content is represented through interconnected pages (or hypercards). The Adaptive Electronic Textbooks (Brusilovsky et. al., 1996) facilitate transfer of normal textbooks existing in electronic form into adaptive electronic textbooks. The system contains a domain model and a learner model. The domain model can either be a set of domain concepts or a network with nodes corresponding to domain concepts and with links reflecting several kinds of relationships between concepts. The learner model is of overlay type storing estimation values for each concept. The concepts are visually annotated representing learner's state of interaction with them and system's recommendation at a point of time.

Vassileva (1997) described a Dynamic Course Generator (DCG) which generates individual courses according to the learner's goals and previous knowledge and dynamically adapts the course according to the learner's success in acquiring knowledge. The teaching material is kept in html files and a graphical editor allows the creation of concepts, connecting them among each other with various types of semantic relations. Once the teaching material is exhausted, or on

learner's explicit request, the system provides test material on the current concept and updates the student model. In case learner is unsuccessful in the tests, the system tries to present additional teaching material, and if cannot, it requests the 'planner' a new course plan based on student model.

Wang (1996) presented an intelligent tutoring system LearnMedia, with its architecture consisting of three layers, the educational agent layer, the knowledge server layer and the repository layer. By deploying repository technology, the architecture provides the means for multiple educational agents to conduct co-operative interactions between students and teachers and among students. Schoch et. al. (1998) described a personal teacher agent, called ADI integrated within an adaptive statistics tutor (AST). ADI supports the learner at the time of browsing through the domain content of the system. It is adaptable to the learner's individual preferences and is adaptive to the learner's current state of knowledge.

The research literature records the efforts to facilitate co-operation among the teacher and student communities in the use of an ITS. However, there is not much evidence to indicate significant attempts to allow the teachers to contribute actively to the development of an ITS without starting the design process from scratch. Almost all traditional ITS are, as Yum & Crawford (1996) observed, "proprietary" programs which only their developers know and can modify. They are not modifiable to suit individual person or curriculum needs and have difficulty to cope with the changing educational, technological and social environment. A rapid ITS development methodology is needed to provide ways in which the subject experts can create small tutoring systems that can be linked into different configurations of larger tutoring systems, customised according to the curriculum needs and thankfully some systems are emerging with an explicit goal of facilitating collaboration in the creation and implementation of courses.

Web Recourse, the Web Retargetable Course Generation System (Lemone, 1996) facilitates creation and sharing of courses on the Internet and easy update and reworking of the course over time. The course uses pre-existing elements such as HTML files, graphics files and header files besides allowing creation of new ones and facilitates their reuse and re-targeting. System uses pre-defined tags for targeting the user, level of competence and other attributes, and provides pre-defined chunks of domain content. Each chunk of knowledge (as contained in various HTML pages) can have pre-defined pre and post topics which can serve as background and more advanced knowledge respectively. Yum & Crawford (1996) proposed a generic interoperable Intelligent Tutoring System (ITS) architecture based on Open DataBase Connectivity (ODBC). They accepted that though ODBC supports interoperable data transfer, the specifications of the ITS domain data and multimedia information need to be agreed upon to enable implementations by external parties.

CALAT from NTT (Nakabayashi et.al., 1996; Nakabayashi et.al., 1997) extends the scope further. The system consists of a WWW server integrated with a tutoring system, and a WWW client equipped with a multimedia scene viewer. The system makes use of text, sound, animated graphics and other media, as well as simulation techniques for hands-on experience. Three types of pages are supported: explanation pages, which are html pages containing plain text, audio and images; exercise pages with possibility of true/false, selection and description types of questions; and simulation pages with interactive simulation environment. The system is based on client-server architecture. The tutoring functions of CALAT are managed by CAIRNEY (Fukuhara & Kiyama, 1993) CAI system which is a multimedia, multi-window ICAI system. CAIRNEY provides adaptive navigation and courseware presentation based on the overlay learner model it maintains. The system is ideal for skills related to visual processes which can be supported directly by multimedia representation.

Most of the currently existing web based educational systems are aimed at domain concepts and the development of the learner's cognitive skill through hands on practice has not received adequate attention, with the possible exception of CALAT where the simulation pages could be used for this purpose. The Hyper-ITS, described in this paper is predominantly based on a cognitive apprenticeship approach. The major differences as compared to some of the existing systems are listed below.

#### ***No pre-defined navigation structure***

In Hyper-ITS, the system's advice is completely dependent on the competence level achieved by the user and is tailored to the actions already carried out by the user. The concepts are linked in an interrelationship network and the sequence of navigation and guidance is not pre-defined. To be valid, any user action must be consistent with the earlier actions. In case of inconsistency, the system will offer graded feedback first pointing out that the current action is not valid and then identifying the possible actions which conflict with the current action. At this point the user gets an opportunity to review the current state of knowledge, identifying any misconceptions or missing conceptions, and can go back and correct earlier actions or implement a different current action. Where the problem is given in the form of the teacher having already set out a scene through some actions, naturally the student cannot change the teacher's actions.

#### ***Major focus on cognitive skill development, not on conceptual knowledge***

Hyper-ITS is employed within a cognitive apprenticeship approach to facilitate acquisition of cognitive skills through the process of observation, imitation, dynamic feedback while learning, interpretation of data, testing, static feedback for testing as well as ensuring far transfer of knowledge and skills acquired through attempting more authentic problems using the 'tools of the trade' - the tools for carrying out the work as opposed to the tools used for learning. The conceptual knowledge is presented in an easily accessible form and is available for reference as and when required while attempting to resolve a problem. The sequence of learning is therefore more likely to be a broad understanding of concepts followed by acquisition of skills while refining the understanding of concepts. However, depending upon individual learning style, a user may wish to spend more time initially to repeatedly go through the exposition presenting conceptual knowledge and obtain a deeper understanding before attempting problems. The system will not prevent this.

#### ***Granular representation***

The domain concepts within Hyper-ITS are acquired in the context of their inter-related concepts enabling a constructive approach to learning. Just as a Hypertext/Hypermedia link brings up a particular text page or audio/video/animation clip, the Hyper-ITS control panel object brings up another interface whether it relates to (i) another perspective on the current data set (ii) a fine grained interface that deals with the details of a coarser grained concept on the current interface or (iii) a fine grained basic application for a student who needs to revise basic steps while learning at a more advanced level. Thus Hyper-ITS has a granular architecture and while the system allows interaction at a coarser grain size, it recommends transition to a finer grain size to be able to guide a user more precisely.

#### ***Process modelling***

Besides employing the overlay model, the Hyper-ITS can also implement process modelling to enable the system, where appropriate, to obtain a greater understanding of the user's mental processes. While this facility may have a limited value at the very basic level of learning, it is highly desirable at an advanced level where the solution may require branching into a number of paths before converging again.

The Hyper-ITS is based on the well proven ITT design briefly described in the Introduction. The Internet permits the smaller knowledge entities (i.e. ITAs) to be held and accessed in a structured manner. With the appropriate structuring parameters, the ITAs created by different teachers build up to a large inventory of accessible knowledge. For designing the ITS for more complex topics covering the knowledge areas of multiple ITAs, an interface is required with defined inputs, outputs and controls linking any combination of existing ITAs. The Hyper-ITS may thus be seen as a system consisting of configurable building blocks within configurable systems that can be utilised by the teachers in various configurations to suit basic or more advanced topics.

### Developing a Web Based Intelligent Tutoring Application

The Hyper-ITS architecture is based on data driven technology which enables the applications to be relatively domain independent and allow easy incorporation of additional domain content and modification of existing material. Creation of new ITAs involves the use of core functionality provided by the Byzantium team (general purpose interface manager and tutoring modules) while the teachers provide domain contents, pedagogy and optional problem bank as shown in the ITA workbench in Fig. 3.

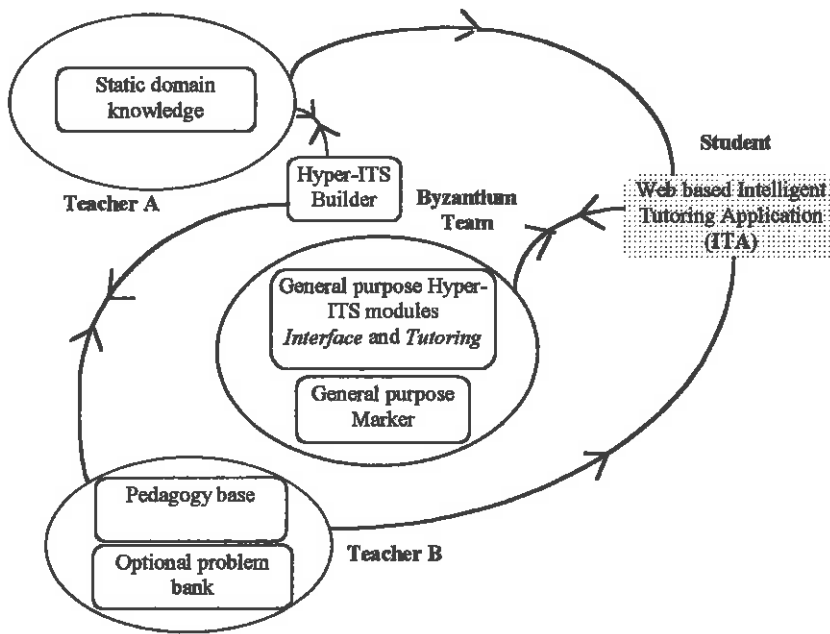


Fig. 3 : Core functionality of a web based ITA



The ITAs enable co-operation in teaching. At the *tutoring tool level*, students in different parts of the world can use an ITA designed by a teacher from any part of the world. Teachers can also incorporate ITAs designed by other teachers into the curriculum and share the *narrative form* questions and *model answers* created by different teachers (see Patel & Kinshuk, 1997a). They may even link the individual ITAs to develop more complex Tutoring Systems.

### Knowledge Representation within the System

The knowledge representation within an ITA is in the form of a consistent network of conceptual atoms. Patel & Kinshuk (1997a) describe the conceptual atoms as containers connected by inter-relationships. Fig. 4 demonstrates such inter-relationships for 7 out of 14 conceptual atoms involved in an enterprise's profitability model based on production volumes and period costs:

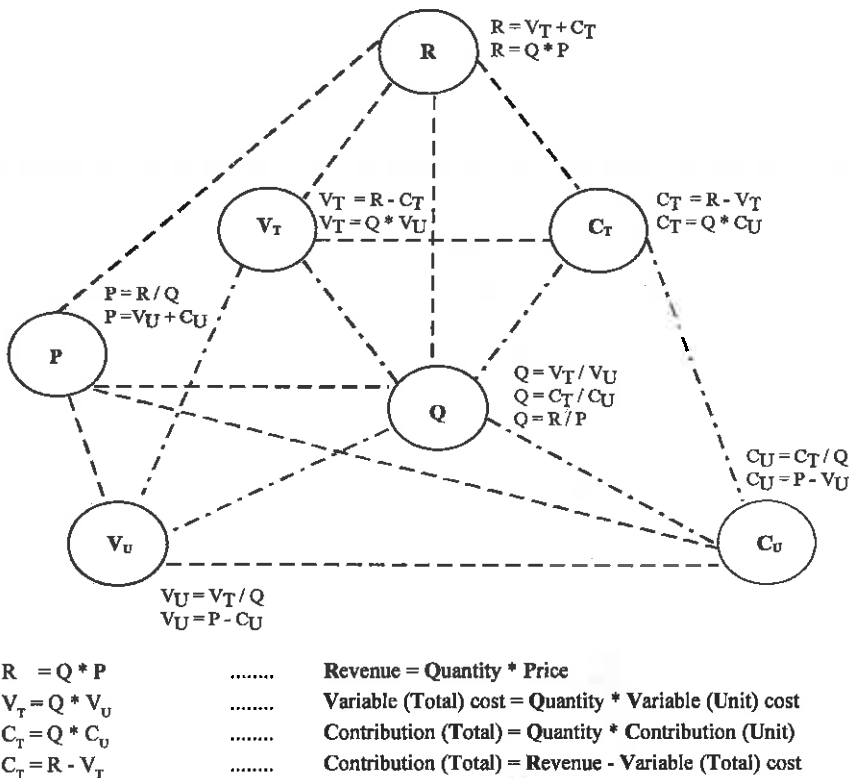


Fig. 4 : P profitability model

An instance of a conceptual atom is legal if it does not conflict with an already existing instance of any other conceptual atom within the inter-relationship network. The network can be seeded by any combination of the conceptual atoms and there is no fixed sequence for the instance creation. Legal instances are permissible even if intermediate conceptual atoms are not instantiated. However, an illegal instance in this case is rejected as the ITA infers a 'missing' conception and the student is advised to first attempt to instantiate the suggested intermediate conceptual atoms. Thus, an ITA is a fine grained tutoring tool that guides a student to an

appropriate level of detail instead of attempting to build up complex feedback messages. The aggregation granularity within an ITA is negotiated through (i) the provision of intermediate conceptual atoms in the interface, (ii) a multi-view interface where a view carries forward the aggregation of the details in the previous view and (iii) a functional interface that zooms-in to reveal the constituents of a complex value presented as a single concept on the main interface e.g. discount factor in the Capital Investment Appraisal ITA.

### **The Framework of Knowledge Representation**

The knowledge representation framework used in the ITAs is similar to the conceptual model for expert systems reported by Wielinga & Breuker (1986). The Wielinga & Breuker model uses four layers to represent an expert's knowledge. The first layer contains the static knowledge of the domain. The second layer is the inference layer. The third layer is the task layer and the fourth layer is the strategic layer. The ITA framework contains the first three layers of Wielinga & Breuker model. Since the ITAs are essentially educational systems and not expert systems, the layers in ITAs are referred by the educational purpose they serve. The first layer contains the static domain material and hence is called domain layer. The second layer contains the pedagogical information which a teacher uses to facilitate effective learning. Since this layer represents the personal choices, biases and personality attributes of the teacher it is called 'teacher model layer'. The third layer relates to information regarding the current problem space, immediate goals and other contextual information. This layer is called 'contextual layer'.

Since the ITAs rely on a granular approach towards concept acquisition and error detection, the system does not explicitly represent any strategic planning. The learning process is broken down into smaller steps through suitable interfaces at various granularity levels. The system attempts to analyse the missing and mis-conceptions in student's competence at fine grain level, eliminating the need of strategic layer present in the Wielinga & Breuker model.

Following sub-sections describe the components of each layer.

#### ***Domain layer***

The domain layer consists of static domain content provided by the designing teacher. The constituents of this layer are as follows:

- Concepts, which are the smallest learning units in the system.
- Concept sets or relationships among concepts in the form of production rules.
- Priorities associated with the relationships to decide which relationship is optimal in case more than one relationship is applicable in a particular context.
- Custom operator definitions to define complex domain dependent operations including logical, mathematical and sequencing operators.
- Constraints on backward chaining to restrict the expert solution and interactive feedback system to employ forward chaining only, if desired.

#### ***Teacher model layer***

This layer consists of the pedagogy base reflecting various tutoring strategies and scaffolding provided by the implementing teacher to help a novice student to limit the problem space and grasp the concepts easily and without the possibility of cognitive overload. This layer also contains optional problem bank created by the implementing teacher to situate the concepts in a particular context, for example, teaching the accounting techniques in relation to, say, leisure management. The teacher can also provide additional diverse contexts for the students to better appreciate the underlying abstract conception of a particular accounting technique.

### **Contextual layer**

The contextual layer contains the current goals and structural information of current tasks:

- system's solution to current problem i.e. the instances of all the dependent variables possible;
- system's problem solving approach i.e. the production rules employed to obtain above values;
- immediate goals.

This information is dynamically updated along with learner's progress in problem solving.

### **Knowledge Processing within the System**

Fig. 5 depicts the representation of framework as used in ITAs. The ITAs provide problem based learning of the domain concepts using distributed architecture involving student, Byzantium server, designing teacher and implementing teacher. The major advantages of such architecture are the ease in maintaining domain independence of the system, easy updates of domain knowledge base, adequate involvement of teachers in their designer and implementer roles, and provision of global optimisation of learning process by taking account of a student's progress in view of the domain knowledge base and pedagogy base. The inference mechanism also benefits from distributed architecture. Client (student) side inferences take care of local context and the server side inferences maintain overall domain concepts network consistency and keep track of overall student progress and performance. The server side inferences, in turn, allow modifications and additions to the pedagogy base to suit larger number of students.

Though the ITAs have capability to generating random problems within the boundaries defined by an educational designer, they allow the implementing teachers to create an optional local problem bank, if desired. They also allow the teachers to hand out problems expressed in narrative form for a student to appreciate situational contexts and develop data interpretation skills. When the students use the system, they can choose a system's randomly generated problem, a teacher created problem from a local problem bank or an empty problem space for narrative form problems, requiring them to identify the given data. The system initialises the problem space using the following contextual functionality:

- *Domain representation initialiser* initialises the system according to the current learning goal for all types of problems.
- *Random problem generator* randomly selects concepts that will be treated as independently given and creates instances of these concepts by randomly generating values within specified boundaries.
- *Prediction boundary initialiser* initialises the boundaries for the overlay model (comparison of a student's attempt to the expert solution) to infer concept acquisition by a student. These boundaries are used by the discrepancy evaluator (explained later) to evaluate a student's action.

The problem space of an ITA uses "fill in the blanks" metaphor by providing an instance container for each of the domain concepts. Students fill values into these containers and for a given problem, the set of instances must be mutually consistent. An instance of a concept that isn't constrained by existing instances of any other concepts is regarded as an independent variable within the problem space. As the students are extensively exposed to the "fill in the blanks" metaphor during their schooling, the empty containers are perceived as a challenge and the students are well-motivated to solve the problem. If the input from a student is identified as an instance of an independent variable, following tutoring functionality is used:

- *Contextual dependency finder* identifies the dependent concepts that can be derived within in the current state of the problem space.
- *Dependency activator*, at client side, activates the instances of the contextually dependent concepts as identified by the *contextual dependency finder* and invokes the *dependency calculator* at server to update their current status in the expert solution.
- *Dependency calculator*, at server side, provides values for the dependent concepts based on domain layer and pedagogy base to update the expert solution. This functionality also facilitates adaptation to a student's problem solving procedure, if it is different from the expert procedure and allows a student to adopt a different route to the solution than the one adopted by the system.

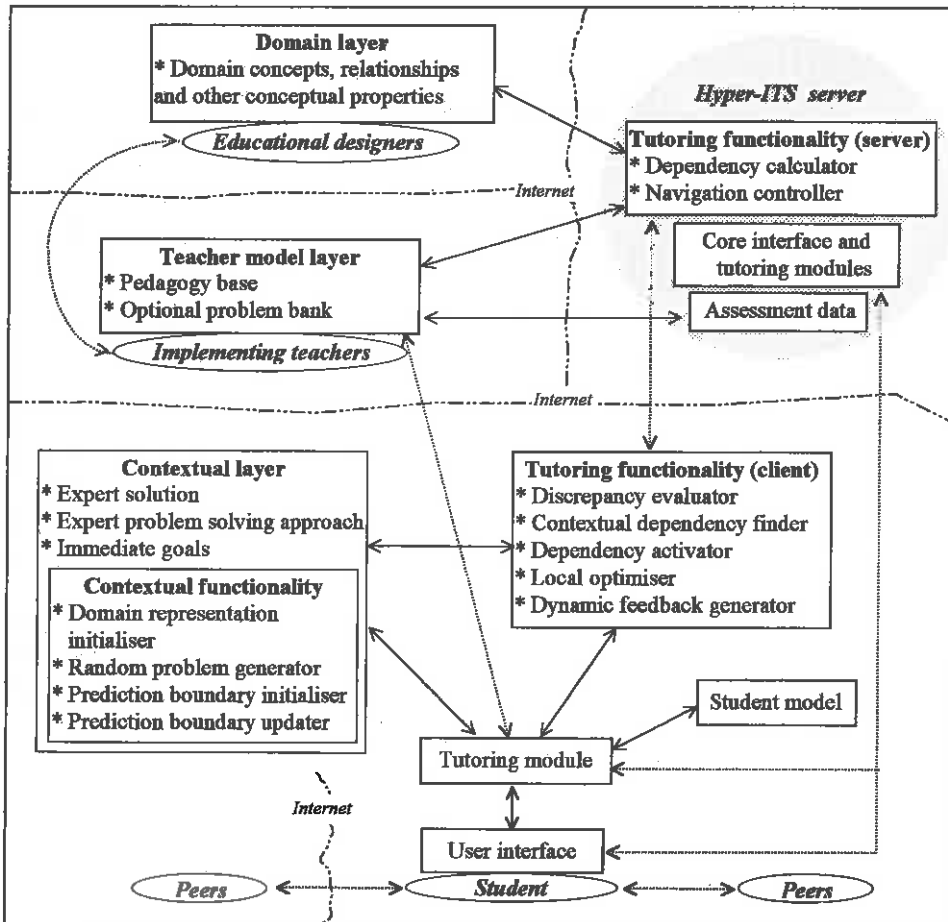


Fig. 5 : Frame work of an ITA

The following contextual functionality sets the acceptable range for a student's inputs for the dependent variables:

- *Prediction boundary updater* updates the prediction boundaries used by discrepancy evaluator in comparing a student's solution with the system's solution. The updater fine tunes the system's initial prediction boundaries to match the route to solution adopted by a student.

If an instance of a concept is constrained by already existing instances of independently given concepts, the system uses the following tutoring functionality for validation of the student input:

- *Discrepancy evaluator*, which evaluates the validity of a student's attempt by matching it with the expert solution within the prediction boundaries.
- *Dynamic feedback generator* provides context based feedback to the student. The messages are generated dynamically to improve semantics and to prevent monotony.

The ITA employs a granular approach for identifying the source of error and for providing feedback. Accordingly, the system classifies a student error into one of the following three inferences (Patel & Kinshuk, 1997b):

- i. Basic misconceptions, where the calculation of an instance for a particular concept is directly possible within a student's current problem space but the student fails to do so due misconceptions about the critical concepts. In such cases, the dynamic feedback generator provides graded scaffolding:
  - a) asks the student to try again stating that the attempt was incorrect;
  - b) suggests the relationship to be used;
  - c) provides the data needed for calculation;
  - d) shows the full calculation, updates the student's problem space with the correct value and allows the student to proceed.
- ii. Missing conceptions are inferred when it is not possible to calculate an instance of a concept directly within a student's current problem state and necessitates calculating the instances of some intermediate concepts first ( the error arising from missing knowledge about intermediate relationships. In such cases, the *dynamic feedback generator* suggests the intermediate concept that should be addressed first.
- iii. If student enters wrong value for some complex concepts, which are in themselves an output of another set of concepts, the *dynamic feedback generator* advises the student to use a finer grain interface. The finer grain interface deconstructs the complex concept into components (for Component Display Theory see Villiers, 1996). The interface captures the student's mis-conceptions in term of one or more component concepts.

If the *discrepancy evaluator* finds a student's input within acceptable range, it uses following tutoring functionality to identify any sub-optimality in student's approach:

- *Local optimiser*, identifies the possible relationships which can be used in current problem space, and determines the best relationship to use based on the priorities specified in the *domain layer*. It allows the system to advise a student based on the student's route to solution and also helps in identifying any sub-optimal approach adopted by the student. The priorities of relationships are generally based on the criteria of simplicity so that the simplest relationship within a given problem space is applied first.

The designing teachers can restrict access to advanced concepts using the navigation controller, which controls the access to different stages of learning within the system as defined by the domain layer and pedagogy base, monitoring a student's successful completion of the pre-requisite stage before allowing access to further stages.

### Further Research and Workplans

The framework, on which ITAs are being developed, has already been implemented and tested in the form of Byzantium ITTs. It has been found successful in providing adaptive and interactive learning environment to the students similar to the traditional one-to-one tutoring. The adoption of this framework will address one of the common criticisms levelled at Internet based educational resources, blaming them for not providing much interactivity except in matters of navigation.

The wide spread use of Hyper-ITS will require a systematic repository of the Hyper-ITS based resources with a facility to find the required resource among a number of variations possible and to obtain an adequate description. Adequate facilities for uploading the newly designed resources or current resources modified to obtain new variations will also have to be provided with a simultaneous updating of the resource description records. The design of a supportive infrastructure for collaborative efforts in itself is a massive task. However, it will also support sharing of other teaching resources, for example, assessment questions reflecting more authentic situations at a graded level of complexity. There is an issue about the compensation for the use of intellectual property and this will have to be resolved in line with such issues addressed by the Internet community in general.

### References

- Anderson J. R., Reder L. M. & Simon H. A. (1996) Applications and Misconceptions of Cognitive Psychology to Mathematics Education. Internal paper, Carnegie Mellon University, Pittsburgh, USA.
- Angelides M. C. & Gibson G. (1993) Pedro - The Spanish Tutor : A Hypertext-based Intelligent Tutoring System for Foreign Language Learning, *Hypermedia*, 5 (3), 205-230.
- Brusilovsky P. L.; Schwarz E. & Weber G. (1996) A tool for developing adaptive electronic textbooks on WWW, *Proceedings of WebNet '96 - World Conference of the Web Society*, October 16-19, 1996. San Francisco, CA, AACE, pp 64-69. Available <http://www.contrib.andrew.cmu.edu/~plb/WebNet96.html>
- Collins A. (1990) Generalising from situated knowledge to robust understanding, *American Educational Research Association Annual Meeting*, Boston, April.
- Collins A.; Brown J. S. & Newman S. E. (1989) Cognitive Apprenticeship : Teaching the crafts of reading, writing and mathematics, *Knowing, Learning and Instruction* (Ed. Lauren B. Resnick), Lawrence Erlbaum Associates, Hillsdale, N. J., 453-494.
- Duchastel (1992) Integrating Hypermedia into Intelligent Tutoring, *Interactive Multimedia Learning Environments* (Ed. M. Giardina), Springer, Berlin, 67-74.
- Fischer G. (1996) Making Learning a Part of Life - Beyond the 'Gift Wrapping' Approach to Technology, *WLF Position Paper based on presentation at the NSF Symposium on Learning & Intelligent Systems*, June 26, Available: <http://www.cs.colorado.edu/~13d/presentations/gf-wlf>
- Fukuhara Y. & Kiyama M. (1993) A knowledge based educational environment for real business settings. *Proceedings of the World Conference on Artificial Intelligence in Education*.
- Kinshuk (1996) Computer aided learning for entry level Accountancy students, *PhD Thesis*, De Montfort University, England, July.
- Kinshuk & Patel A. (1998) Co-operative learning in distance education: An innovation in intelligent tutoring for engineering disciplines, *Software for Engineering Education*, 12, 16-18.
- Leigh D. (1996) The Internet and Distributed Learning: Instructional Designer's Medium and Tool. Available: <http://mailer.fsu.edu/~dleigh/superflux/words/dleigh/NetDistLearn.html>.
- Lemone K. A. (1996) Retargetable Course Generation - A Methodology for Reusability in Distance Education, *ITS '96 Workshop on Architectures and Methods for Designing Cost-Effective and Reusable ITSs*, Montreal, June 10, 1996. Available: <http://advlearn.lrdc.pitt.edu/its-arch/papers/Lemone.html>
- Nakabayashi K.; Maruyama M.; Koike Y.; Fukuhara Y. & Nakamura Y. (1996) An intelligent tutoring system on the WWW - Supporting interactive simulation environment with a multimedia viewer control

- mechanism, *World Conference of the Web Society*, October 16-19, 1996, San Francisco, CA. Available: [http://www.calat.com/calat\\_e/about/papers/webnet96/paper\\_camera.html](http://www.calat.com/calat_e/about/papers/webnet96/paper_camera.html)
- Nakabayashi K.; Maruyama M.; Koike Y.; Kato Y.; Touhei H. & Fukuhara Y. (1997) Architecture of an intelligent tutoring system on the WWW, *Proceedings of the workshop on intelligent educational systems on the world wide web*, 8th World Conference of the AIED Society, Kobe, Japan, 18-22 August 1997. Available [http://www.calat.com/calat\\_e/about/papers/aied97/paper\\_camera.html](http://www.calat.com/calat_e/about/papers/aied97/paper_camera.html)
- Nawarecki E. & Dobrowolski G. (1996) 'Decentralized computer learning systems based on autonomous agent approach', *Lecture Notes in Computer Science*, 1108, 105-113.
- Patel A. & Kinshuk (1996a) 'Applied Artificial Intelligence for Teaching Numeric Topics in Engineering Disciplines', *Lecture Notes in Computer Science*, 1108, 132-140.
- Patel A. & Kinshuk (1996b) Intelligent Tutoring Applets - A New Generation ITS on the Web, *Poster presented at the International Conference on Knowledge Based Computer Systems*, December 16-18, Bombay, India.
- Patel A. & Kinshuk (1997a) Intelligent Tutoring Tools in a Computer Integrated Learning Environment for introductory numeric disciplines, *Innovations in Education and Training International Journal*, 34 (3), 200-207.
- Patel A. & Kinshuk (1997b) Granular Interface Design : Decomposing Learning Tasks and Enhancing Tutoring Interaction, in Smith M. J., Salvendy G. & Koubek R. J. (Eds.), *Advances in Human Factors/ Ergonomics - 21B - Design of Computing Systems: Social and Ergonomic Considerations*, Elsevier Science B. V., Amsterdam, 161-164.
- Patel A. & Kinshuk (1997c) Intelligent Tutoring Tools on the Internet - Extending the Scope of Distance Education, *Paper presented at the 18th ICDE World Conference*, June 2-6, Pennsylvania, USA. Available: <http://www.icde.org/icdeconf18/papers/r2patel.pdf>
- Schoch V.; Specht M. & Weber G. (1998) ADI - An empirical evaluation of a tutorial agent. *Proceedings of Ed-Media/ Ed-Telecom 98*, AACE, VA, 1242-1247.
- Simonson M. (1996) Distance Education: Trends and Redefinition, *Paper presented at the 1996 Frontiers in Education Conference*, November 6-9, Salt Lake City, Utah.
- Vassileva J. (1997) 'Dynamic course generation on the WWW', *Artificial Intelligence in Education* (Eds. B. du Boulay & R. Mizoguchi), IOS Press, Amsterdam, 498-505.
- Villiers M. R. de (1996) 'A component-based interactive practice environment' *Lecture Notes in Computer Science*, 1108, 158-166.
- Wang, H. (1996) 'Learn Media: A Co-operative Intelligent Tutoring System for Learning Multimedia', *Lecture Notes in Computer Science*, 1086, 84-92.
- Wielinga B. J. & Breuker J. A. (1986) 'Models of expertise', *Proceedings of European Conference on Artificial Intelligence*, Brighton, UK, 306-318.
- Yum K. K. & Crawford J. R. (1996) On the feasibility of an interoperable tutorial machine to support the development and delivery of teaching, *Position Paper for ITS '96 Workshop on Architecture and Methods for Designing Cost-Effective and Reusable ITSs*, June 10th, Montreal, Canada.

**Kinshuk** is a Senior Lecturer in the Information Systems Department at Massey University. His areas of research interest include learning technologies, web-based learning systems and intelligent tutoring systems.

**Correspondence:** Information Systems Department, Massey University, Private Bag 11-222, Palmerston North, New Zealand. Fax : +64-6-350-5725. Email: [kinshuk@massey.ac.nz](mailto:kinshuk@massey.ac.nz).

**Ashok Patel** is the Director of CAL Research & Software Engineering Centre. He is an accountant by profession but his research involvement span educational technologies and internet based teaching systems.

**Correspondence:** CAL Research & Software Engg. Centre, Bosworks House, De Montfort University, Leicester LE1 9BH, UK. Email: [apatel@dmu.ac.uk](mailto:apatel@dmu.ac.uk). Tel/Fax: +44-116-257-7193

**David Russell** is the Professional Doctorate Programme Director in the Business School of De Montfort University. He has been involved in Byzantium project to develop computer assisted learning modules for accountancy and cost engineering.

280 *Hyper-ITS: A Web-Based Architecture*

**Correspondence:** Professional Doctorate Programme Director, Graduate School of Business, De Montfort University, Leicester LE1 9BH, UK. Fax: +44-116-251-7548. Email: [drussell@dmu.ac.uk](mailto:drussell@dmu.ac.uk)