Constructive Learning Environments for E-learning

M Sasikumar, CDAC Mumbai
sasi@cdacmumbai.in

1 Instruction design for E-learning

From the early days of hype, e-learning seems to be settling down in select areas. There is still a lot of expectations from e-learning particularly in India, given our lack of good quality faculty at all levels of education. At many levels, including government, NGOs, etc, there is substantial effort in using computers and internet in enhancing learning at different levels.

It has also been generally recognised that content is a major concern for effective e-learning. And, not surprisingly, a lot of the effort is focused on content development. Internationally too, content is a major focus area in e-learning and its variants. The topic of content design - from the perspective of designing *effective* content - is also one of serious research interest. RC Clark's work in architectures of instruction [2] and David Merril's work in identifying first principles of instruction [1] as well as instructional transaction theory are some significant works in this area.

Before, we move on to content design issues, do note that the overall framework being used for e-learning plays as important a role in effective e-learning as content itself. A well-organised learning management system is necessary to interlink the various aspects of learning, such as content, assessment, discussions and feedback; without this, the various fragments do not support or reinforce one another. Similarly, a well thought out assessment component also plays a significant role, in maintaining learner control, motivation and providing feedback.

In the next subsection, we will outline some of the major concerns in instruction design to provide an overall perspective of the field. Instruction design today is a rich field covering the traditional models of instruction, learning theories, instruction design process, standardisation, and so on. Therefore, in this paper, we will just review some general concerns and point out some major aspects, and then focus on a couple of specific aspects of instruction design. In subsequent sections, we will elaborate on these aspects focussed on constructive learning environments. We will, then, briefly look at some of the work in progress at CDAC Mumbai, in this respect, to provide concrete examples of the ideas we are discussing here.

1.1 Aspects of Instruction Design

Instruction design for e-learning has a number of aspects, including the following:

1. Visual design of elements: The color selection, the placement of items on the screen, the amount of detail that is provided on the screen, etc. Apart

from aesthetics, there are concerns such as disconnect among the various elements, placing related elements together, etc which are relevant here. For example, [3] recommends that annotations should be placed next to what is being annotated, instead of listing them below; the book denotes this as the 'contiguity principle'.

2. Effective navigation: The famous indicators of a good UI, namely "where am I", "how did I get here", and "how do I get out of here" are important here also. Advance organisers, showing the overall plan, the objectives and how the various parts of the content relate to them, are all useful devices in minimising navigation-related cognitive load on the user and in allowing him to focus on the learning per se.
3. Type of content: In an e-learning framework, there are many options for presenting the content, ranging from plain formatted text (often in the form of HTML, PDF documents) to live/recorded lecture sessions by human faculty. The other options are audio, animations, images, programmatic simulations, games, etc. A variety of considerations apply in choice of a suitable medium for a particular lesson. These include users' bandwidth constraints (often preventing live video as an option), guidelines from learning theory, and relevance of the media for the type of lesson.
4. Organisation of content: Various strategies for introducing a topic have been studied. Theories such as elaboration theory [6] address this issue. Strategies such as chronological order, increasing complexity order, increasing level of detail order, etc are examples.
5. Pedagogy: This deals with the various pedagogical techniques deployed through the course. This aspect is concerned with the right approach to explain a topic for the intended audience and the expected level of performance. This is, often, the most difficult, and often the most important part in determining effectiveness of the learning framework.

In [1], Merril proposes a set of first principles for instruction, and gives the following prescriptions for an effective e-learning system:

1. learners should be engaged in solving real-world problems
2. existing knowledge should be activated as a foundation for new knowledge
3. new knowledge should be demonstrated to the learner
4. new knowledge should be applied by the learner
5. new knowledge should be integrated into the learner's world

Similar ideas can also be found in [3], which examines a number of general concerns in instruction design and compiles available studies to support/disprove them. The concerns include aspects such as use of graphics, audio overlay, worked examples, collaboration and so on. The authors recommend significant use of worked examples and practice problems relating to real-world scenarios.

2 Theories of Learning and E-learning

In the previous section, we briefly outlined the various aspects of instruction design for e-learning, and remarked that pedagogical issues are perhaps the most critical in this. Much of the concerns in the pedagogical aspects come from the theory of learning, a field with rich contents and even richer history. The early models of learning were of behaviourist nature, considering the learner as a partially filled slate with the task being to insert the missing pieces. The teacher would impart the missing pieces to the learner, almost surgically attaching them to the learner. This model was adequate for certain specific types of tasks where the learner's final behaviour was the primary concern.

With increased understanding of how humans learn, this model gave way to what came to be known as cognitivist learning theory. Ideas such as learner model and strategies of instruction were introduced at this stage. The most popular model today is a refinement of cognitivism, named as constructivism. This model introduces a number of powerful notions of how people learn, and also provides powerful directions to exploit the opportunities provided by computer and communication technologies. Not surprisingly, constructivism has caught the attention of many researchers and practitioners in the field of e-learning. We will also focus on the same in this paper.

The basic premise of constructivism is that learning is individual, in that learning takes place when a learner revises his current mental model of the domain. Neither the existing model nor the nature of revision is directly under the control of the teacher. Therefore, in general, there is no way to ensure that identical learning takes place among a set of learners, against a given instruction. Each learner absorbs the instruction depending on his existing mental model and other characteristics including his learning and behavioural style, and 'constructs' a revised mental model. The task of the teacher becomes, primarily to facilitate this process as well as possible.

Given the focus of constructivism on individual variations in learning, the model encourages use of exploratory and collaborative learning scenarios, unlike the traditional 'lecture' mode delivery of instruction used in behaviourist models. And the strength of e-learning is precisely in supporting these areas of exploratory and collaborative learning.

A number of pedagogical models have been devised, derived partly from the traditional education literature and partly from the work in e-learning. These include repetition, apprenticeship, and so on. The pedagogical model to be used for a specific component of instruction depends on the objective of that part as well as the nature of that part. Merrill has proposed an early matrix [7] for this, a depiction of which is given below.

	Fact	***Procedure***	***Principle***
Recall			
Apply			
Find			

The columns indicate the nature of the part, from the simplest to the most difficult. The rows indicate the learning objective of the part, again from the simplest of being able to recall to apply to novel situations. The grid at the bottom right is the hardest, where the expectation is that the learner will be able to identify where a principle that he has learned can be applied and then apply it suitably. Something like seeing a mirage on the road, and then recognising it as an instance of total internal reflection of light. One can visualise various pedagogical devices in each of the grid cells. For example, for recall kind of performance, one can use iterative practice. For principle-apply performance, one needs problem solving as part of the pedagogy.

Constructive learning environments and learning through games are two major approaches built on constructivist model of learning. We discuss the former in some detail in the rest of this paper. Learning through games creates a game-like environment where the learner plays an interactive game, and learning is integrated into the process. While there are debates on the desirability and effectiveness of using non-serious devices such as games for learning, in the face of lack of adequate evidence, the field has strong supporters too.

3 Constructive Learning Environments

Constructive Learning Environments (CLEs) are learning environments built on a constructivist learning model. These systems provide effective playing grounds for learners to try out what they learn and get constructive feedback. Often the playing ground drives the learning as well, since the focus is on learning whatever is required to handle the assigned task well. The playing ground can take a variety of forms from the simple descriptive problem solving to simulated building of a device. Examples of such CLEs (apart from the two case studies described in the section below) are the chemistry lab simulation available on the web (http://www.chemcollective.org/vlab/vlab.php), and the practice environments provided in the www.w3schools.com for learning languages like HTML, DHTML, Javascript, etc.

Along with most lessons, w3schools provide the user two different windows, one where he can modify a code fragment that is pre-loaded, and the other which shows how that modified page would look on a normal browser. This kind of setups enable students to explore the usage of various constructs, in a simulated reality, giving them a feel for how these actually work. It also enables them to experiment with complexities, such as interaction of different constructs, which are difficult to do in a text book or classroom.

The chemistry lab framework, similarly, allows a student to load various chemical constituents used in a lab (acids, bases, salts, indicators, etc) into typical lab equipments like beakers, pippette, etc and mix them as they like. Indicators can be added to check status and transitions of ph values. The system shows elaborate details for each product selected by the learner, and also tracks behaviour of the products loaded in the containers as the products are mixed. These are powerful ways for students to internalise concepts of chemical reaction.

Developing a CLE is a challenging task in general. Unlike simple e-book or video lectures, CLEs require the system to possess a sophisticated domain model. The system need to model the various components of the task (e.g. the various compounds in the chemistry lab scenario), the way these interact with each other so that it can predict the result of such interactions, and the various misconceptions that a learner can have in the domain. Each of these are, in general, open ended problems. In order to realise practical systems, developers device various engineering approximations and solutions, to make the problem tractable. The Acharya case study below illustrates this point.

Use of pedagogical agents [3] is an often used technique in CLEs. These are animated characters, which watches over the learner activities, and provides guidelines/instructions, either voluntarily or on being prompted. Compared to annotations embedded in the text, this is reported to be more effective in communicating such information to the learner. Such an agent can provide hints, ask probing questions to help the learner identify mistakes he has done, and also advise on the course of action to be followed.

4 Case Studies

CLEs is an active area of research work in CDAC Mumbai. An early work was done in creating an intelligent tutoring system for teaching SQL (Structured Query Language, used for querying relational databases). The system, named

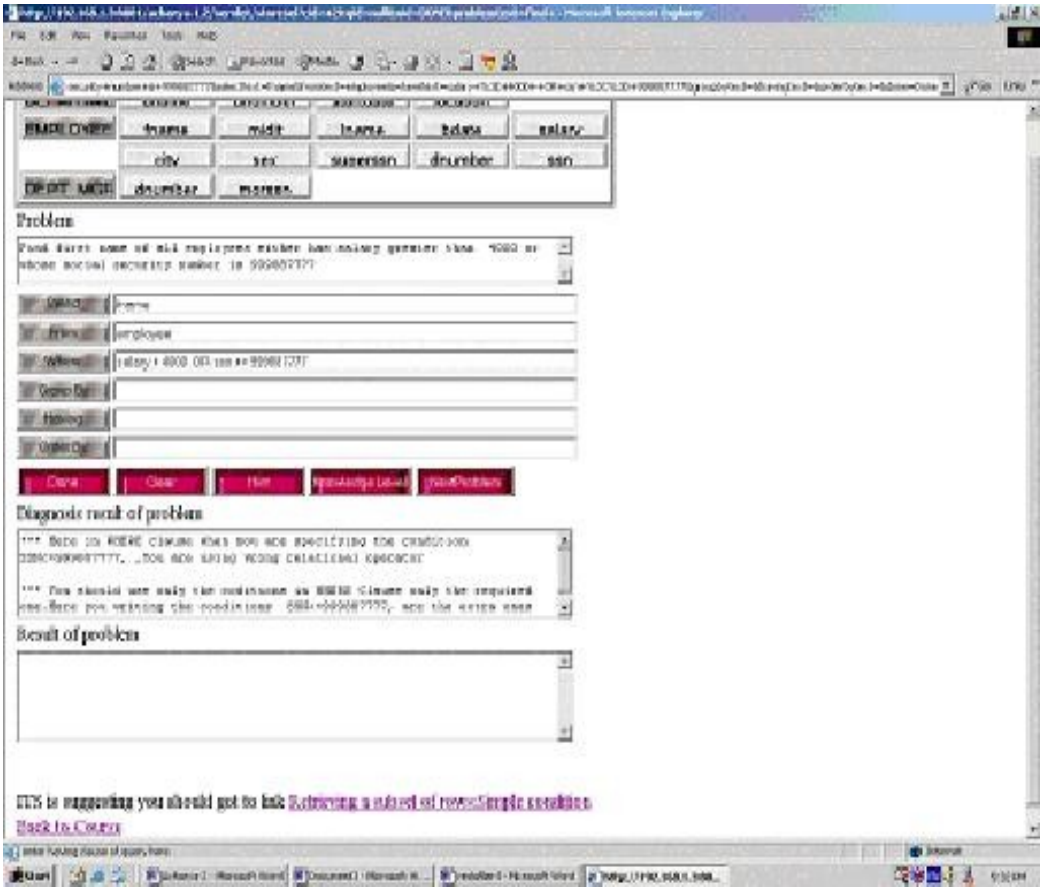

Illustration 1: Acharya - Screenshot

Acharya, [4] has a traditional tutoring component, consisting of textual short tutorials on the various concepts of SQL, starting from the notion of table to complex querying constructs. While the system allowed the user to move around in the lesson structure, at each stage, based on the lessons already completed and his performance so far, the system would recommend the most appropriate lesson for him to pursue.

For the CLE component, with each lesson, a practice environment was provided, where the learner would be given a problem based on concepts covered. See illustration 1**.** At the top is the context of the problem, describing the database tables involved and their relevant attributes. Next is the problem, requiring the learner to write a query to retrieve a specified set of records from one or more of the tables given. In order to reduce issues relating to parsing of the user input, the user is provided a template structure for SQL queries as a set of boxes. He is supposed to fill in the relevant parts in the respective boxes. When done, the system evaluates the result, and generates constructive feedback.

Rather than run the user response against a database and report results, the system evaluates the response comparing the response against the teachers' response to the same query. This allows more constructive inputs to the learner, as to what aspect of the query is wrong, and the nature of the mistake made. Thus, the system can suggest that he has not understood the usage of a specific construct, and suggest that he revise the corresponding part of the course curriculum.

Note that, in general, comparing if two queries are equivalent is a computationally challenging problem. However, considering the contest of a learner of SQL, one can narrow down the dimensions of difference. Thus, we adopted to take an expert answer as the base line for comparison, and addressed common problems in the comparison process itself. For example, one can write the various conditions in the 'where' clause in any order, and produce a semantically equivalent query. Though, there are subtle differences between such queries, they can often be ignored in the case of a learner of SQL. Such types of approximations are critical for effective performance of CLEs.

Marathi Tutor (MT) [5] is another application of constructive learning in the domain of language learning. The target is to teach someone the colloquial use of Marathi language. We decided not to follow the traditional grammar based language teaching, introducing the various grammatical components and slowly leading to sentences. Most users are not interested in that level or kind of exposure to the language. At the same time, it is important for them to learn how to construct phrases and sentences in Marathi, and not be restricted to a set of canned sentences.

Keeping this in mind, the basic approach adopted in MT is to let the user form his own model of sentence formation based on a carefully presented variety of examples. In the main part of the system, relating to sentence formation, the system presents a sentence like “I am going to Delhi tomorrow”, and shows its Marathi equivalent. Once the user is comfortable with this, the system shows variants of this sentence, varying small segments of this sentence. For example, “I am going to Mumbai tomorrow”, “I am going to Delhi next week”, “He is going

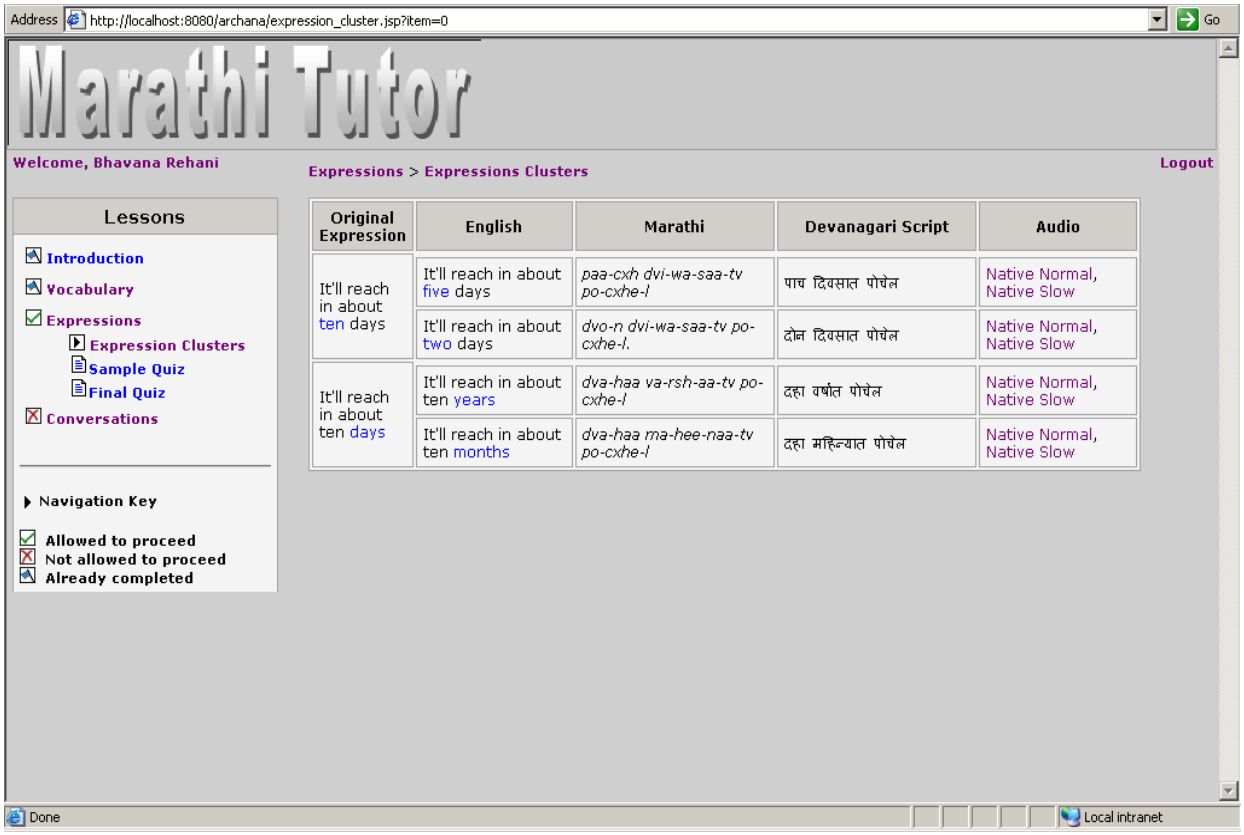

Illustration 2: Marathi tutor - screenshot

to Delhi tomorrow", etc. Thus the learner can "feel" the way in which the sentence formation varies when various components of the sentence changes. Using a pool of such sentences along with a rich set of variants, the user can understand how to form his own sentences in Marathi. See illustration 2.

The system goes through three distinct phases in the tutoring process: the vocabulary phase, the expression phase and then the sentence phase, building up the required background knowledge in the process. The system is being made available over the web.

5 Conclusion

Creating such CLE systems take significant time and resources. But these allow the users to pace themselves as per their background and level of comfort with the learning, experimenting with each concept as much as they need, before moving to another. CLEs allow us to model aspects of a personal tutor, who monitors your work and guides you along. As mentioned earlier, this is the real capability that e-learning provides us beyond what is possible in a classroom.

In this paper, we briefly looked at the basic issues in instruction design in the context of e-learning. The general trend is to move towards constructive learning environments as the primary component of instruction. While, relatively difficult and expensive to create, the returns are high, given that the users get to test out their understanding in a one-to-one interaction with the system, and gets focussed feedback on his mistakes. CLEs require a deeper model of the domain,

and a powerful mechanism to analyse user responses to track down problems. An open ended analysis of the user responses to identify problems is generally not feasible computationally, and hence often an engineering approach is adopted, striking a balance between user flexibility in interaction and difficulty in analysis. The box structure of Acharya, and the system driven model of variant generation for MT are examples of this approach.

CLEs hold substiantial promise in taking e-learning to a wider audience and increasing its effectiveness, and hence is an important field in instruction design.

6 References

1. First principles of instruction. M David Merril. Educational technology, research and development, 50(3), 2002. Available from http://cq-pan.cqu.edu.au/david-jones/Reading/papers/3/first_principles.pdf
2. Building Expertise: Cognitive Methods for Training and Performance Support. RC Clark. ISPI: Washington D.C. Available through ISPI. 1998.
3. E-learning and the science of instruction. RC Clark and RE Mayer. Pfeiffer Press, 2003.
4. Acharya: An Intelligent Tutoring environment for learning SQL. Sandhya Bhagat, Latesh Bhagat, Jojumon Kavalan, and M. Sasikumar. Proceedings of Vidyakash-2002 - International conference on Online learning, Vikas publishing house, India.
5. Marathi-tutor. Archana Rane and M Sasikumar. Technical Report, CDAC Mumbai, 2006.
6. Instructional Design Theories and Models. C. Reigeluth (ed.). Hillsdale, NJ: Erlbaum Associates, 1983.
7. Instructional Design Theory. MD Merril. Englewood Cliffs, 1994.