A new Development Model for Educational Software

Paula Escudeiro1, José Bidarra2

6th September 2006

1Departamento de Engenharia Informática,
Instituto Superior de Engenharia do Porto, Portugal
pmo@isep.ipp.pt; paula@dei.isep.ipp.pt
2Departamento de Ciências Exactas e Tecnologia,
Universidade Aberta, Lisboa Portugal
bidarra@univ-ab

Abstract

In this paper we propose a new development model for Educational Software called X-TEC. It is based on the paradigms of software engineering applied to the construction of educational software. This model allows educational software developers to reduce the gap between instructional design and technical development. Our approach presents two overlapping extensions: the instructional model and the learning environment. The instructional model will be related to the instructor/educational software and the learning environment will be associated with the student/educational software. The X-TEC model promotes the interaction between these two extensions, allowing the deployment of a common development platform.

1 Introduction

In general, educational software is based on development methodologies or methodological approaches, concerned fundamentally with processes or data. Its lifecycle is supported on two different and independent stages: instructional design and technical development.

The gap between the typical skills and terminologies of these two stages usually leads to a problem: the final product is far away from the initial requirements proposed by the author. Consequently, these approaches usually imply the high risk of obtaining low quality products.

In this work, we analyze some of the existing structured methodologies, such as [Yourdon, 1998]) and object oriented methodologies [Booch, 1991] [Coad and Yourdon, 1991], [Jacobson, 1992], [Rumbaugh, 1994].

This study led us to conclude that although object oriented approaches seem to be more adequate than structured approaches, they still fall short of solving the above mentioned gap between instructional design and technical development. We argue that there is a need for a new model focused on results.

This orientation is very important due to the high quality demand placed upon educational systems.

We found different methodological approaches proposed by different authors but none of them seems to fill the required quality patterns. The X-TEC model tries to solve this problem.

The process of creating this model is supported by the software engineering paradigm proposed by Pressman.

2 Conceptual Model

2.1 The overlapping extensions on X-TEC model

The X-TEC model presents two overlapping extensions: instructional model and learning environment. This model promotes an interaction between these two extensions, allowing the deployment of a common development platform, represented in fig. [1].
This platform has quality factors settled on a multifaceted conception described by a set of internal and external factors.

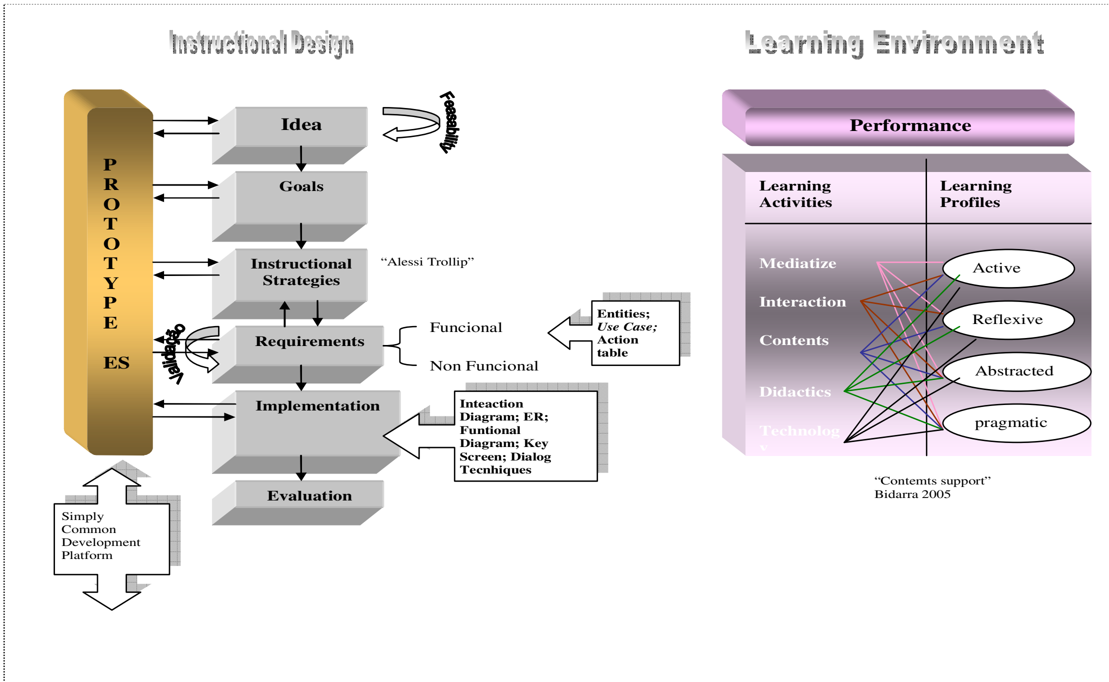

Fig 1: The X-TEC conceptual model

The X-TEC lifecycle is mainly supported on three major activities:

- Cognitive (Knowledge) - mental skills where the brain must be used to perform intellectual tasks.
- Affective (Attitude) - best described as making a commitment - just because we know something, it does not mean we will act upon it.
- Psychomotor (Skills) - physical skills where the body must coordinate muscular activities (some are minor, such as turning a dial with your fingers).

X-TEC model should leverage the educator's creativity and make possible a better vision of Information Technology on Educational Systems, guaranteeing a new related perspective which is represented in the following structure:

Idea;
Objectives;
    target id;
    Support materials;
    Instruction;

Evaluation
Didactics strategies id (based on Allessi Trolip strategies);
Planning (activities chronogram);
Cost/benefit estimate;
Requirement analysis (functional and non functional);
Implementation feasibility;
Development model;
Metaphor id;
Actor’s id;
Action table;
Educational software architecture;
Educational software
System analysis and framing;
Implementation;
Data base structure id;
Relational data model;
Interaction diagram;
Screen architecture;
Key screen;
Dialogue id;
Educational software prototype;
Educational software evaluation (creation of a method to evaluate the educational software quality)

3 Intervening

3.1 Development team

The development team proposed on X-TEC model is composed by three elements: Educational System Annalist (ESA)/Subject Specialist (SS) and Informatics Specialist (IS).
During the development phase a new element may come by: Designer/Audio visual Consulter (AVC)
The role of the development team is organized as follows:

ESA/CS - Prototype developer
IS to Implement explicit messages from educational software scenario.
AVC to Implement implicit messages from scenario.

Implicit messages: trigger student’s reactions; for instance, color, object screen positioning, scenario presentation, images, sound, moving arrow around screen.

Explicit messages: sent by the program and explicitly activated by the student; for instance, menu options.
The consistent use of implicit messages as a complement of explicit messages will enable a better interaction between the student and the educator and between these and the educational software.

4 Actors

4.1 X-TEC main actors

The main actors on X-TEC model are the student, the program and the educator as shown in fig. **Error! Reference source not found.**.

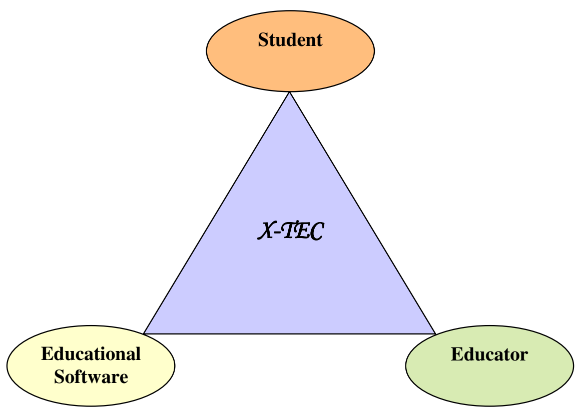

Fig 2: The X-TEC main actors

- Student: uses the educational software for learning the subjects. The students choose the learning environment adapted to their personal characteristics. They should be in control of the system.
- Educator: is the educational system supervisor.
- Educational Software: is the knowledge detainer and adviser.

The X-TEC model will be responsible for giving guidelines, controlling the process phases to allow for a better contribution of these intervenients in an evolutive perspective of the process of creating a quality learning environment.

5 The Learning Environment

5.1 Activity learning net

The activity learning net proposed by Bidarra "Contempt's Construction Support", (Bidarra 2005), is shown through the construction matrix where we can identify four learning profiles (based on Kolb): active; reflexive; abstracted; pragmatic. These learning profiles allow us to fit in two main methods for presenting instructional content "Simulations and the Future of Learning: An Innovative (and perhaps Revolutionary) Approach to e-Learning, Clark Aldrich": deductive and inductive.

- Deductive: allows the learners to work from general information to examples.
- Inductive: supplies examples and requires learners to generalize common patterns.

In addition, there are two main approaches for helping learners to learn: inquisitory and expository.

- Inquisitory: allows the learners to find their examples or general information.
- Expository: supplies examples and general information.

The learning environment on X-TEC model is based on blended learning fig. [3]. The term blended learning is used to describe a solution that combines several different delivery methods, such as web-based courses and knowledge management practices. It is used to describe learning that mixes various event-based activities, including face-to-face classrooms, live e-learning, and self-placed learning. "Learning circuits- American Society for Training & Development"

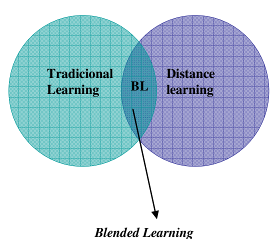

Fig 3: The X-TEC learning environment

Learning theories of Keller, Gagné, Bloom, Merrill, Clark and Gery identify five important elements emerging from a blended process [fig 4].

- Live Events: Synchronous, instructor-led learning events in which all learners participate at the same time, such as in a live "virtual classroom."
- Self-Paced Learning: Learning experiences that the learner completes individually, at his own space/rythm and in his own time, such as interactive, Internet-based or CD-ROM training.
- Collaboration: Environments in which learners communicate with others, for example, e-mail, threaded discussions or online chat.
- Assessment: A measure of learners' knowledge. Pre-assessments can come before live or self-paced events to determine prior knowledge, and post-assessments can occur after live or self-paced learning events to measure learning transfer.
- Performance Support Materials: On-the-job reference materials that enhance learning retention and transfer, including PDA downloads and printable references, summaries and job aids.

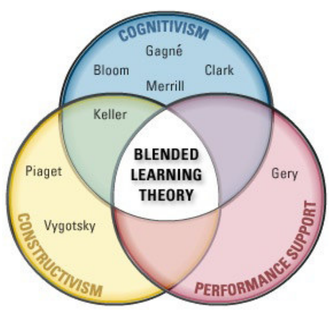

Fig 4: The X-TEC blended process

6 Educational System Architecture

6.1 Three tiered architecture

The X-TEC model is supported by a three tiered architecture [Eckerson 95]: User Interface, Rules and Information Repository, according to fig.5.
The three tier architecture is used to provide increased performance, flexibility, maintainability, reusability and scalability while hiding the complexity of distributed processing from the end user.

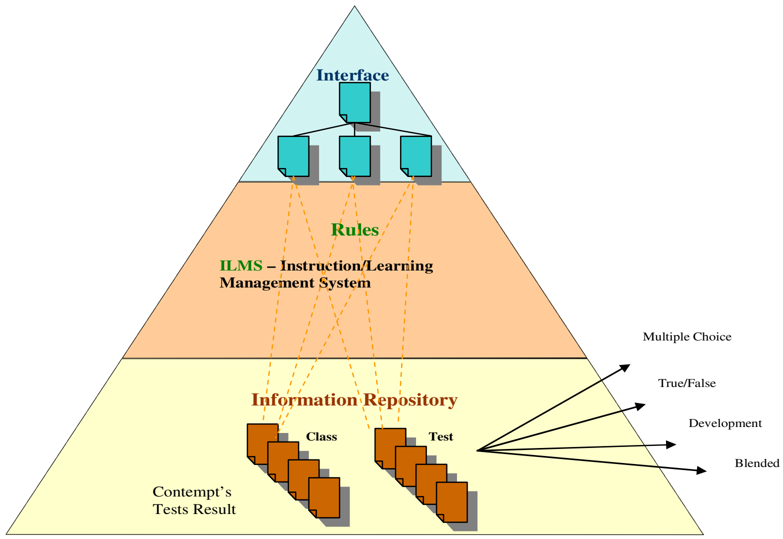

Fig 5: The X-TEC architecture

1st Tier: Interface
It is related with the scenario identification, synchronous and asynchronous communication technologies and implicit and explicit messages.
This following elements act on this tier: Educational Software; Content Specialist's and Designers.

2nd Tier: Rules
It is related with the virtual abstracted organization of the content.
The intervenient on the Rules tier is: ILMS – Instruction/Learning Management System.

3rd Tier: Information Repository
It will allow for all the contents, rules and interface specifications being stored on a warehousing platform.
The intervenient is: Data Base Management Functionality.

In the initial development of X-TEC model we have adopted the instructional strategies defined by Alessi and Trolip, "Computer Based Instruction: Methods and Development" Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA. 1985. The instructional strategies are tutorials, drills, tests, simulation, and educational games.

The ESA has to fit the educational software in one of these didactical strategies. This identification will enable the classification of the educational software into two classes: consultation (class 1) and evaluation (class 2).

The definition of these two main groups obeying specific orientation guides conducts the ESA to choose a learning strategy (learning alone or learning by retroaction) for the educational software.
The ESA plays the role of moderator between the student and the educational software in this learning process.

The ESA should easily be able to evaluate the student learning process, appealing when needed to the information stored in the educational software database.

Three tier architectures facilitate educational software development because each tier can be built and executed on a separated platform, thus making it easier to organize the implementation.

7 Quality Factors

7.1 Goals, principles and actions

X-TEC model is supported by software engineering goals, principles and actions [Pressman, 2001], [Bates, 2000]. In particular, the model is appraised for:

Reusability: How well the model is suited to create, as well as incorporate, reusable components into its execution.

Testability: Each stage deliverables are evaluated to specify how well they are suited for use in a testing process.

Modifiability: The degree to which the educational software product generated using the model is evaluated. In particular, it is determined the degree of object coupling allowed in the model. If the degree allowed of coupling is unconstrained, then the method provides poor modifiability.

Conceptual Integrity: Conceptual integrity is a measure of degree to which the models remain true to the concept of "objects".

Access: How accessible is a particular technology for learners? How flexible is it for a particular target group?

Cost: What is the cost structure of each technology? What is the unit cost per student?

Teaching and Learning: What kinds of learning are needed? What instructional approaches will best meet these needs? What are the best technologies supporting this teaching and learning process?

Interactivity and user-friendliness: What kind of interaction does this technology enable? How easy is it to be used?
Organization and user-friendliness: What are the organizational requirements and the barriers to be removed before this technology can be used successfully? What changes in organization need to be made?

Novelty: How new is this technology?

Speed: How quickly can courses be mounted with this technology? How quickly can materials be changed?

8 Conclusion

This study is the first step towards creating a model for the development of educational materials based on results.
It is our ultimate goal to train a workforce of talented educational designers, equipped with the fundamental skills to effectively undertake any problem in educational systems design.
We will go in depth with the process of creating a model that allows developers of educational software to reduce the gap between instructional design and technical development.
Tacit knowledge, or the knowledge held by people in their brains, is fed mainly by two streams: training and experience. We believe it could be an interesting research to track the training and experience of the present online developers of educational software in order to check if their background corresponds to the necessary profile of the knowledge holder and manager. Indeed, and besides training and experience, the profile should also include social and economics variables.

References

[1] Allesi, S. e Trollip (1985) S. Computer Based Instruction: Methods and Development, Prentice Hall, Inc. Englewood Cliffs, New Jersey, USA, 1985
[2] Bates Tony, (2000) A. W. Tony Managing Technological Change: Strategies for College

and University Leaders. San Francisco, 2000

[3] BloomBertram B. Mesia, and David R. Krathwohl, (1964) Bloom B. S. Taxonomy of Educational Objectives,: The Affective Domain & The Cognitive Domain. New York: David McKay Co Inc.

[4] Booch, (1994). G. Booch, Object Oriented Analysis and Design With Applications, Second Edition, Benjamin/Cummings, Menlo Park, Califórnia, 1994.

[5] Clark 1994, Gery, (1994) Gery, GJ Making CBT happen. Boston: Weingarten. Clark, RE Media will never influence learning. Educational Technology Research and Development

[6] Coad and Yordon, (1991) P. Coad and E. Yourdon, OOA –Object Oriented Analysis, 2nd Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1991

[7] Crossley, K. e Green (1990) Le Design des Didacticiels: Guide Pratique pour la Conception de Scénarios Pédagigiques Interactifs. ACL-Editions, Paris France 1990

[8] Eckerson (1995) Eckerson, Wayne W. "Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications." *Open Information Systems 10,* 1 (January 1995): 3(20).

[9] Gagné, (1996) Gagné, Robert M. and Medsker, Karen L. (1996). *The Conditions of Learning Training Applications.* Florida: Harcourt Brace & Company.

[10] Jacobson, (1992) Jacobson, Object Oriented Systems Engineering, Addison-Wesley, 1992.

[11] Keller/Back (2004) Keller, M.; Back, A.: Blended-Learning-Projekte im Unternehmen, Learning Center der Universität St. Gallen, St. Gallen, 2004

[12] Merrill, (1981) Merrill, M.D., Kowallis, T., & Wilson, B.G. Instructional design in transition. In F. Farley & N. Gordon (Eds.), Psychology and education: The state of union

[13] Minken, I., Stenseth, B. E Vavik L., (1998) Educational Software. ULTIMA-Gruppen A/S, Haden, Norway, 1998

[14] Pressman Roger S., (2001) Pressman, Roger S. Software Engineering a Practitioner's Approach, 5 th Edition, McGraw-Hill Companies Inc, 2001

[15] Purinima Valiathan, (2005) ASTD-Linking People, Learning & Performance. Learning circuits- American Society for Training & Development

[16] Rumbaugh et al, (1991) J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen, *Object Oriented Modeling and Design*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.

[17] Yourdon, (1998) E. Yourdon, *Managing the System Life Cycle*, 2nd Edition, Yourdon Press/prentice Hall, Englewood Cliffs, New Jersey, 1998.