A Technological Framework for the Authoring and Presentation of T-learning Courses

F. Bellotti[1], S. Vrochidis[2], S. Ramel[3], I. Tsampoulatidis[2], P. Lhoas[3], M. Pellegrino[1], G. Bo[4], I. Kompatsiaris[2]

[1] University of Genoa, Genoa, Italy, [2] Centre for Research and Technology Hellas, Thessaloniki, Greece
[3] Public Research Center Henri Tudor, Luxembourg[4], Giunti Labs S.r.I, Sesti Levante, Italy

Abstract—**Broadcasting interactive learning applications through the digital TV promises to open new pedagogical perspectives, also in a life-long learning perspective, given the wide penetration of the medium. This article proposes an open flexible and composable framework for the development, the delivery and the presentation of t-learning courses in interactive digital TV (iDTV). The framework is divided into two main parts: the production side, where the course is prepared and the client side, where it is presented on iDTV, and where the user can perform the educational interaction. The course production is supported by an ad-hoc designed authoring tool, while the runtime user interaction on iDTV is managed by a multimedia course player providing personalization services and a library of educational and entertainment elements and services. Seven experimental t-learning courses were created by pedagogical experts in several knowledge domains and served as an important test and evaluation bench for the framework, in view of the upcoming extensive end-user testing.**

Index Terms—**authoring tool, iDTV, MHP, t-learning**

I. INTRODUCTION

Television has had a long history of performing an educational function for the mass audience, typically by broadcasting culturally relevant movies, documentaries and news as well as educational programs. The idea of Distance Learning through TV blossomed extensively in particular as a complementary educational option besides PC-based e-learning and traditional analogue TV educational programs. In particular, TV-based interactive education promises a huge potential due to its ability to support interactivity, while compensating for the low penetration of Internet-enabled computers in comparison with the penetration of a TV in a household.

"T-learning" was the new term, which prevailed for the definition of TV-based interactive learning [1]. The first forms of learning with interactive digital TV (iDTV) have been little more than modified or enhanced videoconferencing. Today, iDTV platforms for learning provide a large amount of audiovisual and educational content to the viewer through interactive and content personalization. iDTV is considered as the convergence of television and computer technologies by encompassing three important features typical of computer-based technologies [2], such as interactivity, personalization and digitization.

Digital television mostly relies on the Digital Video Broadcasting (DVB) standard, characterized as DVB-T for terrestrial, DVB-S for Satellite and DVB-C for Cable transmissions. DVB has been defined by a consortium of public and private organizations in the iDTV sector [3]. In the DVB schema, the digital TV signal is transmitted as a stream of MPEG-2 data known as a Transport Stream (TS). This stream consists of a set of sub-streams (elementary streams), where each sub-stream can contain MPEG-2 encoded audio, MPEG-2 encoded video or data encapsulated in MPEG-2 stream. Subsequently, the TS is passed to the multiplexer and then to a Radio Frequency (RF) transmitter in order to be broadcasted. The overall broadcasting system for digital TV is illustrated in Figure 1.

The received signal is demodulated and afterwards it has to be decoded appropriately. The common TV sets are manufactured to deal with analogue signals. Hence, a device called Set Top Box (STB) is used to transform the digital signal. Moreover, it also provides a middleware, based on an embedded Operating System (OS), which is an execution environment for running the interactive applications that are broadcast in a channel together with the main audiovisual stream. Execution environments are standard and the most common are: the European Multimedia Home Platform (MHP), the American Open Cable Application Platform (OCAP) and DTV Application Software Environment (DASE), the Japanese STD-B23/STD-B24. Since MHP is the standard in Europe and a subset of it, the Globally Executable MHP (GEM), is becoming the common reference world-wide, in this article we focus on MHP.

MHP is the middleware system for interactive TV development designed by the DVB Project [4]. The first draft of MHP was released in August 1999 and the first version of MHP 1.0 was approved by DVB in February 2000. MHP offers a standard platform for application developers. Applications are written in Java and HTML, so they don't depend on any single hardware platform or operating system. Due to the iDTV's special context, MHP-Java applications are slightly different from normal Java applications. However due to the similarities with Java applets, MHP-Java applications are called Xlets.
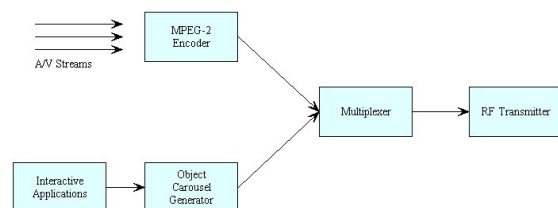

Figure 1.   Overall broadcasting system for digital TV

On the one hand, MHP Java limitations are mainly related to the constraints given by the STB's hardware and OS in terms of computational power, memory size, storage, communication facilities, screen resolution, font and colour availability and their size is severely constrained by the limited bandwidth available. On the other hand, MHP provides support for those special features that are essential in the digital TV world such as low-level access to the TS, service information access, and support for the specialized graphics model of the digital TV. MHP can be extensively exploited by t-learning as it offers the proper middleware for learning interactive applications.

The purpose of this paper is to propose an open flexible platform for the development of t-learning courses by exploiting the available iDTV technologies in a pedagogical way.

The paper is organized as follows: section II presents the the t-learning course structure and the main components involved, while the technological framework is introduced at section III. Subsequently, section IV describes the development of t-learning courses, by analyzing the authoring tool, while section V deals with the client side, where the Course Multimedia Player is presented. Results and evaluation are discussed in section VI and related work is presented in section VII. Eventually, section VIII concludes the paper.

II. T-LEARNING COURSE

A t-learning course is considered to consist of two main parts: the broadcast video (i.e the audiovisual stream) and the interactive application. These parts can be fully or semi synchronized in order to form a t-learning course. In this section the structure of the interactive part of the course will be discusssed and the synchronization with the video will be defined. The proposed t-learning course structure is illustrated in Figure 2 and will be described in detail in the next subsection.
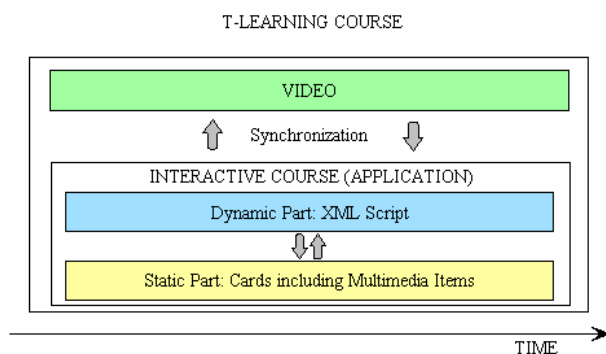

Figure 2.   T-learning Course structure

A. Course Structure

The definition of the interactive course is based on the structure of a Learning Object (LO) [5]. A Learning Object can be defined as both the basic unit of a learning experience and as a small, atomic chunk of learning that can be reused in different context. In other words, the LO structure is actually an aggregation of items, which grants a customizable and flexible reuse.

Following this definition and by adopting the LO model specifically developed for t-learning interactive courses, it is possible to integrate several multimedia components and thus create educational material and content suitable for distribution through an iDTV infrastructure. Thanks to this model, the interactive course can be divided into a static and a dynamic part so the content author can manage separately the various components, as well as the parameters that define their behavior, and then generate in a simple way the final LO (*course*) in the form of an XML script.

The dynamic part of the interactive course is described in this XML script, which can be interpreted at runtime. It is Object Oriented, which is close to human reasoning, easy to agree and specify. Moreover, Object Orientation is well supported through powerful development tools (e.g. UML). The script is event-oriented as well, which is particularly suited for the TV environment, where an application may be synchronized with the underlying A/V stream.

The main objects nested in the script are called cards. Cards are units of contents displayed on the screen and can contain a multimedia item. As multimedia item can be considered either a multimedia page (MPage), which may consist of texts, images, audios and buttons accompanied by the TV stream in various formats (e.g. inside a quarter frame), or an interactive edutainment unit, such as a game. The cards form the static part of the interactive course. As multimedia page is considered a normal page that could appear in the TV screen while games include quizzes, puzzles and so on.

The XML script specifies the cards that constitute a course, their content (in terms of the above mentioned templates), and their time scheduling, since several of them may be synchronized with the A/V stream. Synchronization is expressed in terms of time triggers that are inserted in the A/V stream and are able to trigger relevant cards, as it is specified in the script. The order in which cards are displayed when a corresponding time trigger event is received is defined in tree-like structured sequences that can contain personalization tests. The time triggers are also divided in categories that correspond to alternative paths of the course.

The script also contains information about global variables, that can be defined by the author and serve to store, for the whole length of a course, course-specific information that are typically used to schedule the triggering of the cards according to the preferences, needs and/or other parameters of the current user.

Finally, through the script the course author can also specify the aspect and the functionalities of the navigation bar, which is displayed as a stripe at the bottom (or top) of the screen and is controlled through the four iDTV colour buttons (red, green, yellow, blue). Sample functionalities that can be offered by the navigation bar include: personalization settings, choice of course categories, exit/hiding/restoring of cards, helps, further info, other advanced services. Functionalities are grouped in four clusters, each one of which is mapped to (and accessed through, by end-users) one the four coloured buttons. A course may have several different navigation bar configurations that change dynamically during the course itself (e.g. in different cards, different categories, when no card is on display) in order to provide the most suited, context-aware support. These different configurations are to be specified by the author in the course script.

B. Multimeidia Items

In order to support reusability, the multimedia items (Mpage and games) that can be included in the cards, are implemented as specific templates, which are fully configurable in terms of contents and appearance. These templates can thus be instantiated one or more time in a course. Each template includes a large number of configurable properties (e.g. backgrounds, fonts, area sizes and positions, etc.) which are stored in configuration files. These files also support personalization. That is, the properties of a single template instance can further be parameterized according to the dynamic (i.e. evolving even during the course) profile of the end-user. This is achieved through the introduction of parametrical tags that can be parameterized by the course content author on the parameters defined by the Personalizer's module, which is described in the next section.

1) Multimedia Page

A multimedia page is the most common multimedia item used in courses: it presents information to the user, with basic interactivity. As presentation potential capabilities for content creators should not be limited, multimedia pages do not restrict the position of graphical elements, but allow free configuration of these components on the page. Multimedia pages are template-based in order to allow reusability. They can be shown at runtime by using the specific MHP-java classes, which read the respective configuration file. This file supports the definition of components as well as personalized expressions.
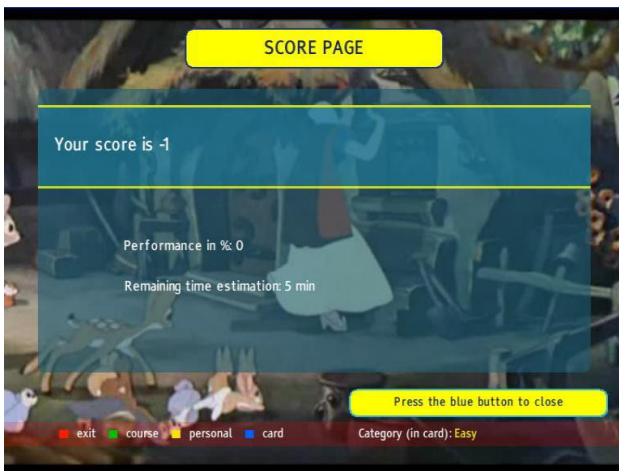

Figure 3. Instance of a Multimedia Page

The components' types define what kind of graphical element they correspond to. More specifically, they can be text components, color rectangles, images, small animations, buttons, image buttons or timers. Buttons allow some minimal interactivity by providing pre-defined possible actions, like closing the card, sending an event, opening another card, changing the current category or setting a personalization value. Each component type contains specific properties that must be defined to configure it, in addition to the common properties corresponding to size and position. In addition, the page can also contain a background, and defines the position of the video stream (either full screen, full screen leaving a stripe at the bottom, or in a rectangle whose bounds can be defined). A sample multimedia page as it presented to the viewer through the iDTV screen is illustrated in Figure 3.

A related and extended concept is the notion of Multimedia Presentation. A multimedia presentation is a special kind of multimedia page that is spread over more than one screen, from which the user can navigate by using "previous" and "next" buttons. Each screen is called a multimedia presentation shot, and can be defined using the same components as for multimedia pages. The additional button types "previous" and "next" allow content creators to define where and how they want to appear on each shot. Multimedia Presentation can be considered as a set of interconnected Multimedia Pages.

2) Interactive Edutainment units

Games in iDTV could play an important role in t-learning although today are used mostly for entertainment. Nowadays, a considerable number of games exist for iDTV, covering various categories, such as arcade, adventure, puzzle and educational games. Quizzes, Puzzle and Memory games could increase the interest of the viewer-learner supporting the concept of relaxed-learning that seems suited to TV. T-learning has the ambition of creating educational games for a wide range of users, in particular those with limited attitude to computers. Games are considered as an integral part of a t-learning course as they could support the learning procedure involving a wide audience through challenging and engaging activities that are able to meet the typical user need for relax and sympathy.

Figure 4. Instances of Games

The games that can be included in a course are based on specific configurable templates that constitute a Java class package, which is structured as follows. GameTemplate is a Java class that abstracts general game functionalities. The concrete subclasses of GameTemplate are the actual game typologies offered by the library. Sample game typologies include: Memory, Puzzles, VisualQuiz and Couples. The games played by the end-users will be instances of such subclasses. The actual games running on the STB are instances of such game typologies. Every game instance comes bundled with its own resources specified by the pedagogical content author (i.e. images, animations, fonts, graphics, text, etc.).

On the production side (i.e. off-line), an author – even not skilled in computer science – will be able to create instances of the various game typologies (i.e. actual games that will be broadcast and played) by editing parameters (e.g. number of available lives, score and timing rules) and inserting contents (e.g. images, graphics, questions, animations, etc.). This can be done manually, or, much more efficiently, through an ad-hoc developed authoring tool that we will describe in the next section.

The runtime system running on the STB manages the life-cycle of every card and its contained elements, games included (i.e. loading, starting, managing its exit and possibly forcing a pause and/or stop), according to what specified in the script. A game will be able itself to recursively manage other games (i.e. building trees of games, according to the author's script and the user preferences).

In Figure 4, sample instances of Quiz, Couples, Memory and Puzzle games are taken from the TV screen.

C. Personalization in t-learning course

In general, the final goal of personalized learning is to provide a learning path that is matched to the learner's needs and abilities, resulting in a more efficient and high quality learning process. In order to obtain this matching of learner's profile and objectives, current learning context and available pedagogical resources, a well-defined description of each component involved in the process is needed, with specific focus on the user model. An additional interesting aspect of the personalization process is that, once the user model has been identified, the accuracy of the personalization can be iteratively improved with time, as more dynamic data are collected and stored regarding the ongoing interactions of the user with the system and the continuous monitoring and re-assessment of the user's satisfaction. This also allows for a classification and "clustering" of learners [6].

Personalization in terms of t-learning implies that a potential iDTV learner can easily be offered on his/her TV equipment a selection of available pedagogical contents and services according to his/her interests, skills and preferences. The module, which is responsible for allowing personalization in the proposed t-learning framework, is called Personalizer. This module is capable of user tracking, taking into account long term features as age, sex, etc., and session features, like score in a game, difficulty course level and other course related preferences.

The Personalizer keeps track of the dynamic user profile (e.g. current values of the user interaction, such as score, what pages have already been visited, what quizzes, etc.) and of the persistent user profile (e.g. preferences, etc.). Before finishing, every card is responsible for updating the Personalizer. For instance, the results of a questionnaire may change so the global score of the course has to be updated. The values stored can then impact the navigation of the course (by running alternative paths depending on personalization values), or the content of cards (by taking into account personalization variables in cards configuration files).

III. TECHNOLOGICAL FRAMEWORK

The proposed architecture, which is illustrated in Figure 5, is split in two parts: the production side where the content is prepared and the receiver side where the course is presented to the viewer through the appropriate terminal.

The production side is the area where the course content is created. As the course development is based on the requirement of reusability, the content is structured in the appropriate format so it is playable by a Course Multimedia Player that is also transmitted with the course package.

The educational A/V stream for the t-learning course is built by a TV producer, while the contents are developed in the authoring tool by the educators. The content including games, images and text is used for the development of personalized courses and can be retrieved from a server where learning resources are stored. Eventual A/V-application synchronization is achieved with the aid of the authoring tool as well, where the content created is matched on specific time stamps inserted in the A/V stream. Subsequently, the A/V stream is fed into the MPEG2 encoder while the content produced by the authoring tool and the Course Player Xlet are inserted into the object carousel. In this way the sub-streams are constructed and then multiplexed to form the final TS, which is broadcast.
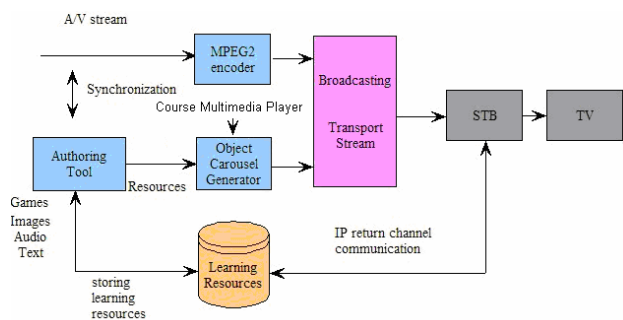

Figure 5.   General Framework for the development and delivery of a t-learning course

The signal is received at the receiver side and processed by the STB where the A/V stream and the applications are restored from the TS. The Xlet that contains the t-learning course runs on the STB MHP middleware presenting the content of the course. The existence of an Internet IP return channel on the STB allows the use of on-demand features. Through this return channel it is possible to send requests regarding the retrieval of additional learning resources as well as information about the viewer in order to support more advanced personalization features.

This schema is realizable through existing TVs, equipped with STBs, thus reaching a wide share of population. An Internet connection (even not broadband) is not required, however additional services could be supported in order to enhance the user interaction and experience. On the other hand, non-linear A/V stream contents would require (at least with present technologies) broadband connectivity, which is not supported by the majority of current TVs.

From a pedagogical point of view, this schema involves an important role of the author, who defines the learning space and provides strong guidance (the A/V stream, the cards (scheduling and contents)), and of the user, who has a certain freedom in exploring this knowledge space (e.g. deciding whether to perform an interactive test/quiz/game or not, follow a branch or another of a set of interactive

cards). For any course's implementation, the author can choose the desired level of freedom/customization (e.g. by preparing contents for various categories, setting user-performance-based triggers for more information cards, quizzes, letting a number of parameters variable for personalization, etc.).

IV. PRODUCTION SIDE

The main component involved in the production side is the authoring tool. This tool is composed of two logical parts: the content insertion and configuration tool and the script builder. The former is responsible for the template configuration while the latter deals with the definition of the sequence of the course based on time and logical events. The output of this tool is composed of the sequence script in XML format (i.e. the XML script that defines the dynamic part of the interactive course), the configuration files of the templates and the resources (images, audio files, etc) involved, in a structure package.
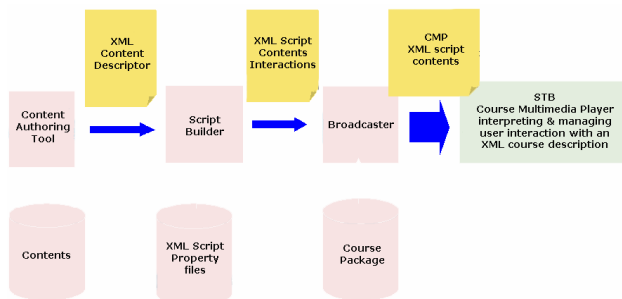
Figure 6. Production Chain

The whole production chain is illustrated in Figure 6.

A. Authoring Tool

The AT is a visual development environment where the author is supported in the creation educational courses for iDTV. It consists mainly of two logical parts which are integrated in a common user interface: a) The script builder, which is the part for writing the course script and actually defines the dynamic part of the interactive course (scheduling of the cards, definition of alternative course paths) and b) The card configurator part for the creation and the configuration of the instances of the service templates (e.g. Edutainment Templates) which are included in the static part of the course. The architecture of the AT is illustrated in Figure 7.
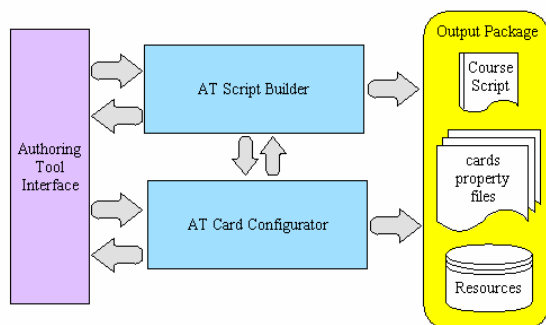
Figure 7. Authoring Tool architecture

The concept of the AT is based on the creation of Learning Objects in a form of XML script and on the

configuration of existing templates. The tool provides a clear graphical interface (Figures 8, 9, 10) which supports visual composition features as drag and drop, content previews and object designing. The tool provides a "WYSIWYG" environment where authors can focus on the contents and the logic of their course without being concerned about the internal structure and the constraints imposed by the XML script.
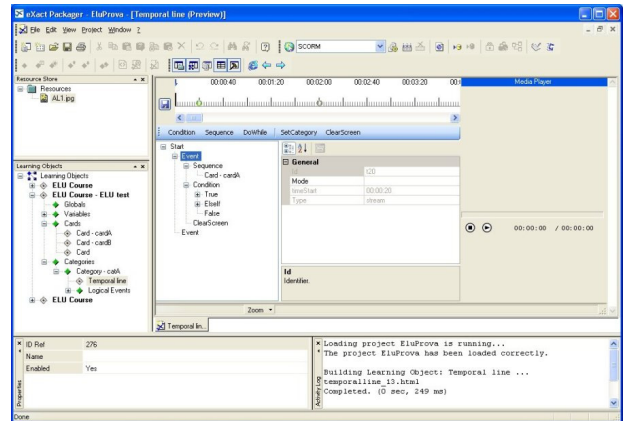
Figure 8. AT Script Builder

The AT script builder (Figure 8), which is dedicated to the definition of the user categories and the scheduling of the cards, provides authors with an intuitive way for defining events lists (temporal or logical) and consequent actions, with the support of personalization as well. The timeline bar proved an easy medium for content authors to use since it is more comprehensible comparing to event triggered-based flow charts. The tool handles the definition of all the supported operations/instructions by using an XSD schema in a "secure" mode, thus avoiding the risk for the author to generate invalid XML files.

The AT card configurator (Figure 9), which is responsible for the configuration of the cards provides a friendly environment where existing templates can be customized. These templates include Multimedia pages with images, buttons and text, multimedia presentations, delivery broker and edutainment templates, which can be configured appropriately in terms of appearance, fonts, score computation modalities, content etc., as described in a previous sub-section.
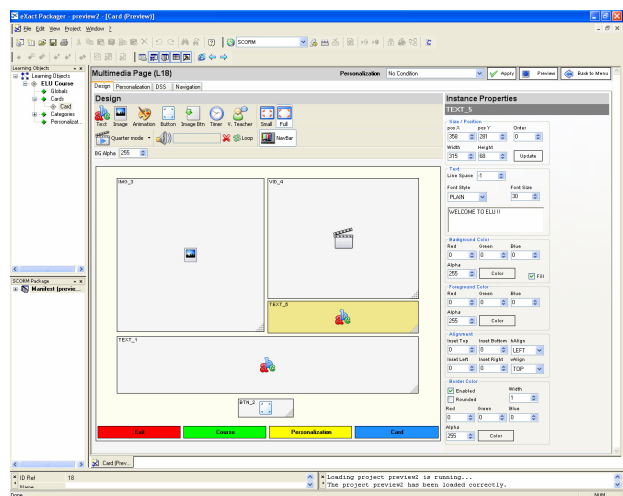
Figure 9. AT Card configurator

Navigation bar configuration is possible at card level where the user can select from the existing menu items and also create new ones as it is shown in Figure 10. The new menu items and navigation bars, designed by the user, are stored automatically in the XML script.

The tool supports two modalities of configuring a newly instantiated service: full configuration, which gives the author full access to all the configurable parameters of the service, and the Style configuration, where the graphics are pre-defined and the author has to insert the contents (e.g. questions and answers, images to be manipulated, graphs, etc.). The first approach is suitable for high customizable solutions, but may be time-consuming and addresses mostly to experienced authors, while the second approach is more efficient since the author can focus more on content than appearance while authoring process becomes more rapid and homogeneity insured.
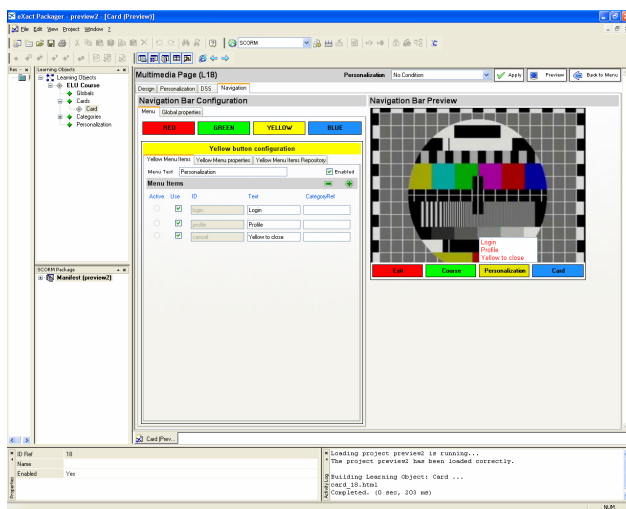

Figure 10.   Navigation Bar Configuration

V.   CLIENT SIDE

The Course Multimedia Player (CMP) is the t-learning client software that reads the data generated by the authoring tools to present and manage the iDTV course. It runs on the user's MHP STB, that supports interactivity between the learner and the application.

Two main steps in the presentation process can be identified as shown in Figure 11:

- Building the interactive application specified in the XML script.
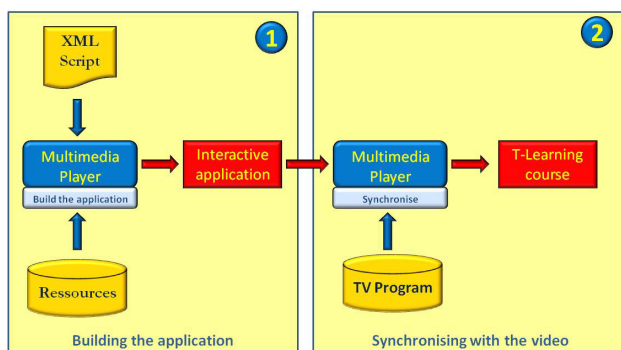- Executing the application with the video.

Figure 11.   Running the application in the Client Side

A.   Building the interactive application

When the t-Learning application is launched by a TV user, the CMP parses the scripts to instantiate the T-learning course. As explained in previous section, the XML script contains all the data required to instantiate a course. By reading this XML description file, the multimedia player creates the static part of the course (each page/card) and the dynamic part, involving inter-card navigation, user personalized paths, and synchronization with the TV-program.

B.   Executing the Application with the video

After the instantiation, the CMP starts playing the course, which involves the following components:

- Graphical Rendering Manager
- Navigation Manager
- Synchronization Manager
- Personalization Manager
- Interactivity Manager

1)   Graphical Rendering Manager

The Graphical Rendering Manager is responsible for the graphical interface. It displays all the course components, including the video stream, also managing overlapping, transparencies and "picture in picture" effects. As reference templates, the content author can select among three main models: full Screen (FS), full Screen with TV picture (FSTV) and Stripe (S) (Figure 12).
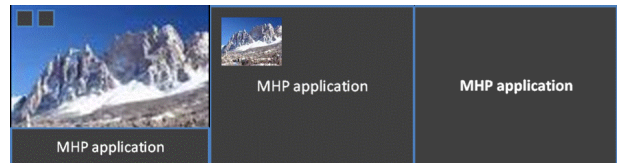

Figure 12.   Card Display Modalities: Stripe, Full Screen with TV Picture, and Full Screen Instance of a Multimedia Page

2)   Navigation Manager (personalised)

The course path defines which card to show when, mainly based on time events. However, some constructs allow the content creators to define more complex paths that can change dynamically:

- Sequences can be defined to specify the next card to show once a card has been closed
- Timer information is used to specify maximum duration of cards (after which a card is closed), as well as latest start time (global time after which the card is not started – this prevents cards to be shown for a too short time)
- Expressions based on personalization conditions are evaluated at run-time and allow to choose between alternatives (in the form of "if-then-else" or "do-while" statements). Such expressions use either pre-defined variables (e.g. age, sex, etc.) or variables that have been defined by the user during the creation of the course.
- Finally, categories allow the definition of complete alternative paths for the course. They can be defined freely, but usually the are used to

specify user levels or areas of interest (e.g. hard, easy level).
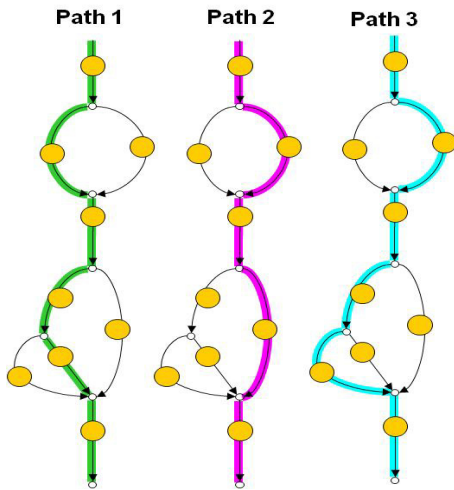
The Figure 13 illustrates these possibilities.

Figure 13.   Different course paths– due to a user's personal values - for 3 different user categories. Circles correspond to cards.

3) Synchronization Manager

A t-learning application can be synchronized with the TV program over which it is broadcast. Synchronization is implemented through time triggers that are to be inserted in the broadcast TS, and trigger listeners that are part of the Synchronization Manager. Triggers may correspond to events such as beginning of the first part of a course, end of the second part, etc. They are listened to at runtime by the synchronization manager that can thus take adequate actions as specified by the author in the course's script. For instance, start application card could be displayed or a new section in a course activated.

Time triggers can be inserted in the TV program's timeline at various levels of granularity, allowing the t-learning course to be context-aware with regard to the video stream.

4) Personalization Manager

The Personalizer manages the dynamic user profile (e.g. global and partial scores, what cards have already been visited, what quizzes, etc.) and the persistent user profile (e.g. preferences, sex, age, etc.). Not only is this module responsible for updating the Personalizer's data, but it also orients the user by taking decision at runtime as explained in the subsection II-E.

5) Interactivity Manager

Interactivity involves two aspects:

- Local interactivity: it is provided by programmable components that enable the user to change the behavior/settings/appearence of the objects that are displayed. The user can access local interactivity simply through her/his remote control.

- Server interactivity: same as local interactivity but the actions of the user are sent to a server as "requests" via a physical return channel. For instance, the server interactivity can be used to store general data in a centralized way or also to request additional information on a given topic, even if bandwidth constraints limit the

potentialities of this option. The use of the return channel is quite rare at present. So, contents and services provided through such a channel should concern only optional features.

VI. RESULTS AND EVALUATION

In this section experimental t-learning courses which were created and delivered with the proposed framework will be presented in order to evaluate the performance and the efficiency of the system.

A. T-learning Course Development

The above presented framework has been implemented in parallel with the development of 6 t-learning applications. This has allowed a continuous interaction between technological implementation and pedagogical inspiration and evaluation.

In order to further support course creation, integration between the AT and the MHP analyzer from IRT software [7] was realized. The analyzer is capable of supporting textual debugging and profiling, as well as a PC preview of an MHP application.

Using the AT, the author could iteratively design and implement the cards, fix their scheduling in the timeline and check the preview.   The preview functionality provides two levels: single card level and course level. The first one allows immediate inspection of any card, just after its development, while the second one allows analyzing the whole course, checking also the overall scheduling, navigation and learning paths alternatives. The integration inside the AT allowed an instant preview of the course making the testing procedure easier. This intensive testing proved itself to be useful also to signal bugs of the CMP and of the AT, and highlighted the need for improvements and additional functionalities as it emerged from the concrete experience of the course developers. These iterations were important to complete a fully functional t-learning framework comprising of a runtime multimedia player and an offline authoring tool that satisfies the requirements of the pedagogues and the content creators.
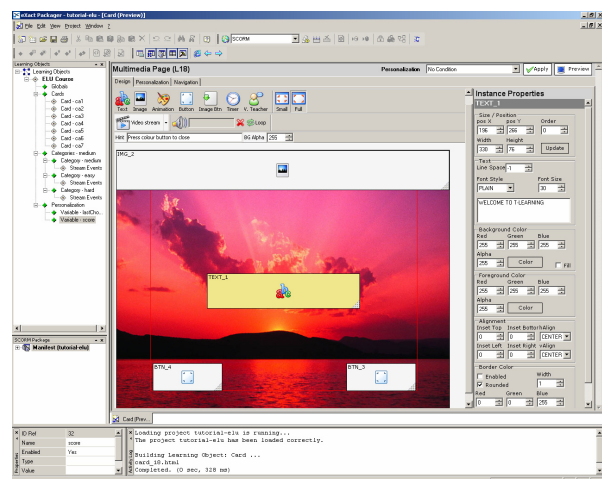

Figure 14.   Multimedia page configuration

In Figure 14 the authoring of a card composed of a simple multimedia page is presented. At this stage the author uses the card configurator in order to design the multimedia page. In this case the author has selected an

image background, a text area on the top, an image area an image area in the middle, and two buttons. At the right window of the AT interface the author is capable of configuring independently each object (i.e. button, text area, image area) of the multimedia page. The navigation bar has also been configured. All the color menus are available: exit, customization and global and card-level functionalities.

The preview of the card is always accessible by the author with the aid of MHP analyzer. The preview of the above discussed and designed multimedia page is illustrated in Figure 15.
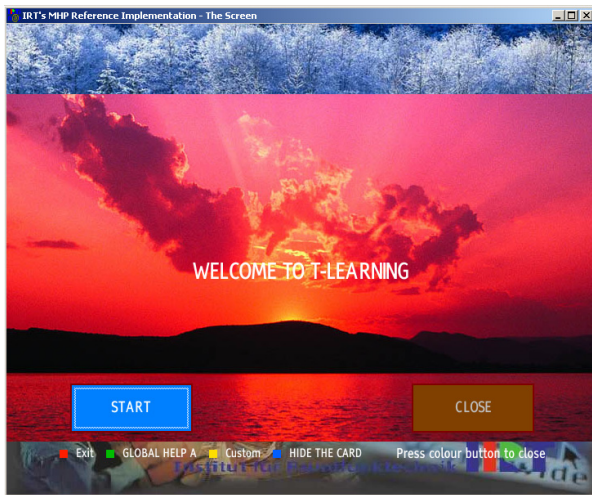

Figure 15.   card 1 preview in simulation environment.

The developed tools have supported the production of six courses, in the following topics: basic business, statistics for the Master in Business Administration, cultural heritage, mathematics for elementary schools, computers in a life-long learning perspective and driving school/road safety. The subjects of the courses were selected in order to cover a variety of different areas and address different target groups. These courses have been designed by pedagogical experts from universities and institutes from 6 European countries[1].

Besides the six aforementioned t-learning courses, an interactive enhancement of the "Snow White and the Seven dwarfs" animated film from Disney has been implemented, in order to support the study of English as a foreign language. This experimental course was created in order to further test and evaluate the framework also from a more technical perspective, considering most of the supported functionalities and realizing multiple course paths with complex sequences and conditions.

Figures 16-19 show some snapshots from this last application including Couples, Multiple Answers, Puzzle and VisualQuiz games respectively.

[1] Czech Technical University in Prague, http://www.cvut.cz/
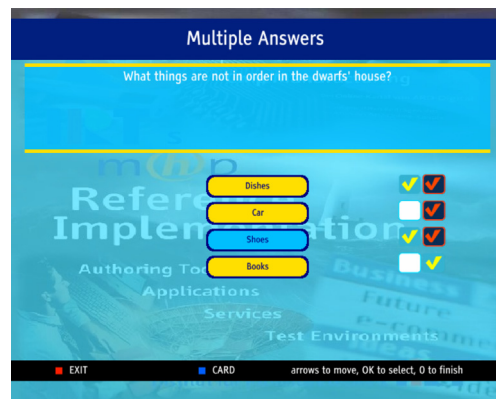University of Hradec Kralove, http:// www.uhk.cz/
Institute of the Hungarian Academy of Sciences,
http:// www.sztaki.hu/
University of Ljubljana, http://www.uni-lj.si/
Riga Technical University, http://www.rtu.lv/
State Institute of Information Technology, http://www.viti.lt/

Figure 16.   Couples Game Preview

Figure 17.   Multiple answers game preview
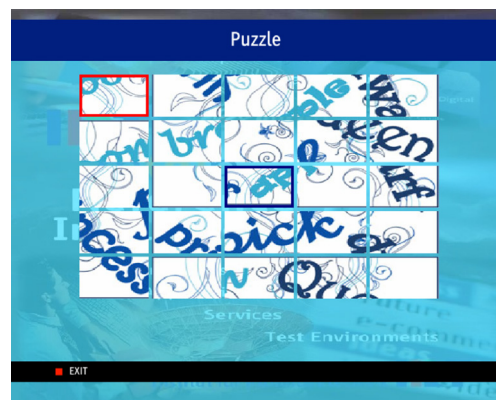
Figure 18.   Puzzle Game Preview
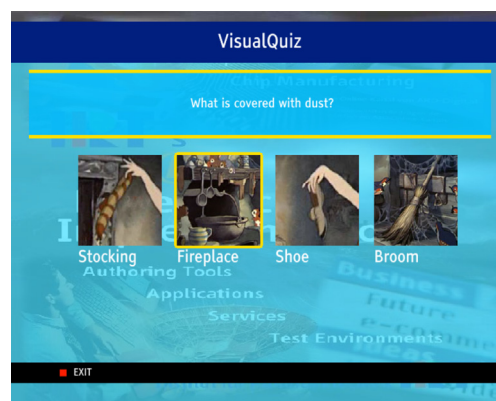
Figure 19.   Visual Quiz Preview

The implemented t-learning courses typically last 15 to 30 minutes. The size of the MHP applications and the resources is 4 to 5 MBs, of which 2MB are reserved to the CMP. The loading time, considering a data bandwidth of 200 Kb/s, is around 4 minutes, which has to be improved for commercial airing.

Functional tests have been successfully run in real context of use settings, where courses run on commercial STBs and are displayed on TV sets and users can interact through a remote control.

B. Evaluation

The proposed framework was evaluated with the development and presentation of the aforementioned t-learning courses.

The most important evaluation was performed by the pedagogical experts. The challenge was to let them free to implement their ideas using the proposed t-learning framework. During the creation of the six t-learning courses the pedagogues were able to have a clear view of the production and presentation phase. During the course creation it was possible to evaluate the performance, the efficiency and the potential of the authoring tool which appeared to be satisfactory mostly because of the friendliness of the environment, the advanced functionalities, and the capability for an instant card and course preview. This preview feature has been important also to have a preliminary evaluation of the Course Multimedia Player. However, the tests with real STBs, performed afterwards, were useful for authors to tune the courses in particular from the point of view of the graphic appearance (given the different rendering capabilities of a TV screen with respect to a PC screen), performance (given the slower processing power of the STB's CPU), and the exact synchronization with the underlying audiovisual stream.

The next steps of the work will involve the extensive testing of all the applications with samples of their intended target audience. In particular, the objective of such tests is to verify the effectiveness of the applications to meet the knowledge-acquisition potential expected by the pedagogical experts.

VII. RELATED WORK

Although the research and development in iDTV-based t-learning is rather limited up to date, some interesting approaches have been presented.

Some first ideas regarding t-learning in digital TV were expressed in [8] while in [9] a t-learning model was proposed based on MHP. This model defined different features and functions in t-learning including personalization and interactivity. Furthermore, examples of different opportunities in learning via iDTV were provided.

Moving to technical implementation, an interesting approach was the technological framework for TV-supported collaborative learning, which was proposed in [10]. In this approach t-learning contents are created with appropriate tools and delivered, however personalization was not applied and the fully automatic creation of the course was not supported. In a more recent work [11] a more general framework is presented with content creation based on XML structures and templates.

Although this work has been an interesting basis, it is still inadequate to deal with learning oriented content as it serves more general purposes. Furthermore the two aforementioned approaches haven't been tested thoroughly in iDTV or simulation environment, as no specific results are presented.

Another relevant work [12] was presented recently regarding the presentation of web contents to iDTV for entertainment. This approach is trying to convert the TV into a web browser and the return channel is required in order to allow browsing. Although such approach could be more relevant for IPTV, it would be of use in order to transform existing HTML e-learning courses to t-learning ones. Despite these advantages, this approach has limited potential for t-learning as it requires the presence of return channel and it is doubtful if it can also satisfy other learning requirements (e.g. personalization).

Considering the market approach, nowadays there are considerable tools available, both open source and commercial, like Composer [13] and Grins Pro Editor [14]. More specifically, the former is capable of creating content in NCL [15] language for Ginga middleware while the latter is a SMIL [16] editor. Such tools could have been used for the creation of pedagogical content due to their flexibility, however, as they are not education-oriented, they would fail to satisfy the needs of pedagogues.

VIII. CONCLUSIONS

Broadcasting interactive learning applications through the digital TV promises to open new pedagogical perspectives, also in a life-long learning perspective, given the wide penetration of the medium. However, this requires the availability of proper tools for authoring and deployment of applications. In this paper a flexible framework for the creation and display of t-learning content on iDTV was presented.

The course creation exploits a template logic that guarantees a clear structure for the content. This logic also allows reusability, which is a big advantage in the iDTV world, where the available bandwidth and the storage capabilities of the STBs are limited and the production costs are high and the time-to-market short. Moreover, the configurability option allows tailoring services and contents, which is considered as an added pedagogical value.

This template-based approach allows the creation of flexible, modular courses that are interpreted by a multimedia course player running on a STB. The framework is extensible, and can be upgraded with new contents/services (e.g. virtual avatars and more complex games), also on demand by exploiting the return channel.

The aim is that the description of the framework presented in this paper will contribute to informing experts in the field of the technology-enhanced learning about the design of new tools and applications based on iDTV. Possible research directions for the iDTV scientific community are various. We believe that interesting fields of future developments include: enhanced support for personalization (which is particularly important in a medium, such as the TV, which reaches a variety of users); optimization of the applications' packaging and loading; design of new templates for edutainment

services; exploitation of multi-channel synergies in order to increase the interaction among users.

REFERENCES

[1] P. Aarreniemi-Jokipelto, "T-learning Model for Learning via Digital TV", 16th EAEEIE Annual Conference on Innovation in Education for Electrical and Information Engineering (EIE), Lappeenranta, Finland, 2005

[2] M. Lytras., C. Lougos, P. Chozos & A. Pouloudi, "Interactive Television and E-learning Convergence: Examining the Potential of T-learning", *ECEL2002, The European Conference on e-learning,* Brunel University, UK, 2002.

[3] Digital Video Broadcasting, retrieved October 18, 2007, from http://www.dvb.org.

[4] Interactive TV Web. Retrieved October 18, 2007, from http://www.interactivetvweb.org.

[5] Marta Rey-López, Ana Fernández-Vilas, Rebeca P. Díaz-Redondo, José J. Pazos-Arias, Jesús Bermejo-Muñoz, "Adaptive Learning Objects for T-learning", in the 5th International Conference on Web-based Learning (ICWL), 2006.

[6] Y. Blanco-Fernández, J. J. Pazos-Arias, A. Gil-Solla, M. Ramos-Cabrer, B. Barragáns-Martínez & M. López-Nores, "A multi-agent open architecture for a TV recommender system: A case study using a Bayesian strategy",. *In Proceedings of the Sixth IEEE International Symposium on Multimedia Software Engineering,* Miami (FL), USA, 2004.

[7] MHP analyzer, IRT. http://www.irt.de.

[8] P. Aarreniemi-Jokipelto, J. Tuominen, "Experiences with an Interactive Learning Environment in Digital TV", Proceedings of the 4th IEEE International Conference on Advanced Learning Technologies, ICALT 2004, 30 August-1 September 2004, Pages: 296-300

[9] P. Aarreniemi-Jokipelto, "T-learning Model for Learning via Digital TV", 16th EAEEIE Annual Conference on Innovation in Education for Electrical and Information Engineering (EIE), Lappeenranta, Finland 6th-8th June 2005.

[10] M. Lopez-Nores, A. Elexpuru-Eguia, Y. Blanco-Fernandez, J. J. Pazos-Arias, A. Gil-Solla, J. Garcia-Duque, B. Barragans-Martinez and M. Ramos-Cabrer, "A Technological Framework for TV-supported Collaborative Learning", Proceedings of the IEEE Sixth International Symposium on Multimedia Software Engineering (ISMSE'04), 2004.

[11] V. Vrba, L. Cvrk, M. Sykora, "Framework for digital TV applications", Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies table of contents Page: 184 , ISBN:0-7695-2552-0, 2006.

[12] S. Ferretti, M. Roccetti, "MHP Meets The Web: Bringing Web Contents to Digital TV for Interactive Entertainment", Proceedings of the Eighth IEEE International Symposium on Multimedia (ISM'06), San Diego, USA, 2006.

[13] Composer, middleware Ginga – NCL, http://www.ncl.org.br/index_.html.

[14] Grins Pro Editor, http://www.oratrix.com/Products/G2E.

[15] L. Soares and G. Filho, "Interactive Television in Brazil: System Software and the Digital Divide", EURO ITV 2007, Amsterdam, the Netherlands, May 24-25, 2007.

[16] The Synchronized Multimedia Integration Language, http://www.w3.org/AudioVideo/.

[17] Enhanced Learning Unlimited (ELU) project. Retrieved March, 2008, from http://www.elu-project.com.

AUTHORS

Francesco Bellotti is with the ELIOS Lab, Department of Biophysical and Electronic Engineering (DIBE), University of Genoa (e-mail: franz@elios.unige.it).

Stefanos Vrochidis is working as a Research Engineer within the Informatics and Telematics Institute, Thessaloniki, Greece, (e-mail: stefanos@iti.gr).

Sophie Ramel is with the Center of IT innovation as software Research & Development engineer within the Public Research Center Henri Tudor, Luxembourg (sophie.ramel@tudor.lu).

Ioannis Tsampouladitis is working as a Research Associate within the Informatics and Telematics Institute, Thessaloniki, Greece, (e-mail: itsam@iti.gr).

Pascal Lhoas, Master in IT, is working as a project manager within the Center of IT Innovation of the Public Research Center Henri Tudor, Luxembourg (pascal.lhoas@tudor.lu).

Matteo Pellegrino is with the ELIOS Lab, Department of Biophysical and Electronic Engineering (DIBE), University of Genoa (e-mail: pellegrino@elios.unige.it).

Giancarlo Bo is with the Giunti Labs S.r.l., Sestri Levante, Italy (e-mail: g.bo@giuntilabs.it).

Ioannis Kompatsiaris is working as a Senior Researcher within the Informatics and Telematics Institute, Thessaloniki, Greece, (e-mail: ikom@iti.gr).