A T-learning Courses Development and Presentation Framework

F. Bellotti, S. Vrochidis, E. Parissi, P. Lhoas, D. Mathevon, M. Pellegrino, G. Bo and I. Kompatsiaris

Abstract—**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.**

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 a 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

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 Associate within the Informatics and Telematics Institute, Thessaloniki, Greece, (e-mail: stefanos@iti.gr).

Eirini Parissi is working as a Research Assistant within the Informatics and Telematics Institute, Thessaloniki, Greece, (e-mail: parissi@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).

Damien Mathevon is with the Center of IT innovation as engineer specialized in multimedia and iDTV within the Public Research Center Henri Tudor, Luxembourg (damien.mathevon@tudor.lu).

Matteo Pellegrino is a PhD student in 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).

Publisher Identification Number 1558-7908-062008-04

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. 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.
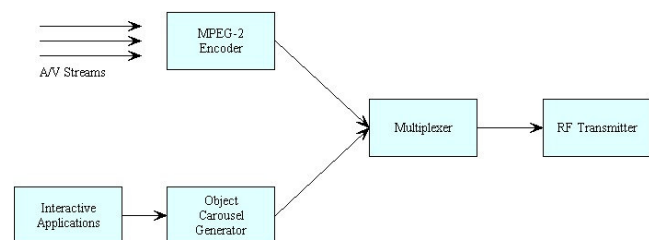

Fig. 1: overall broadcasting system for digital TV.

Subsequently, the transport stream 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.

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 which are essential in the digital TV world such as low-level access to the transport stream, 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.
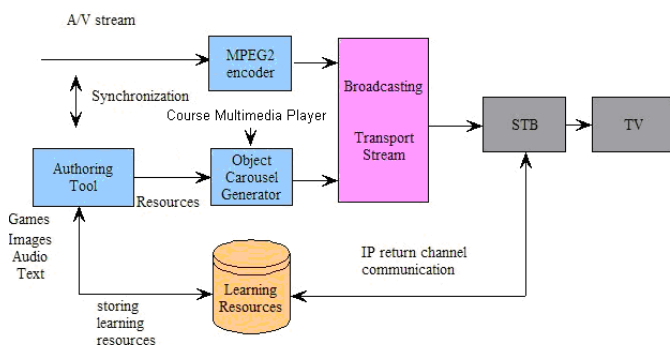

Fig. 2: General Framework for the development and delivery of a t-learning course.

The purpose of this article 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 article is organized as follows: section II presents the proposed framework, the t-learning course structure and the main components involved. Subsequently, section III describes the development of t-learning courses, by analyzing the authoring tool, while section IV deals with the client side, where the Course Multimedia Player is presented. Eventually, section V concludes the paper.

II. GENERAL OVERVIEW

A. General Framework

The proposed architecture, which is illustrated in Figure 2 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 which 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 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 transport stream, which is broadcast.

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 transport stream. 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.

B. T-learning Course Structure

The course creation 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 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 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.

This XML script describes the t-learning course and is interpreted at runtime by the Course Multimedia Player on the STB. 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 and can contain 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. As multimedia page is considered a normal page that could appear in the TV screen while games include quizzes, puzzles and so on.

In order to support reusability, the above items (Mpage and games) are implemented as specific templates, which are fully configurable in terms of contents and appearence. These templates can thus be instantiated one or more time in a course. Configuration files also support personalization. That is, the properties of a single template instance can further be parametrized 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.

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 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 specificy 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 4 iDTV colour buttons (red, green, yellow, blue). Sample functionalities 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 4 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.

C. Multimedia Page

A multimedia page is the most common item used in a card to create a course. It allows the content creator to freely locate static assets as titles, pictures, text zones and also interactive assets as buttons.

Figure 3 shows a multimedia page displayed by transparency on a full screen TV-program.

The Mpage is template-based in order to allow reusability. Such page can be shown at runtime by using the specific MHP-java classes which read the respective property file. The property file supports the customization and the personalization of the page including text fonts, text boxes size and location, etc.

A sample Mpage as it presented to the viewer through the iDTV screen is illustrated in 0.
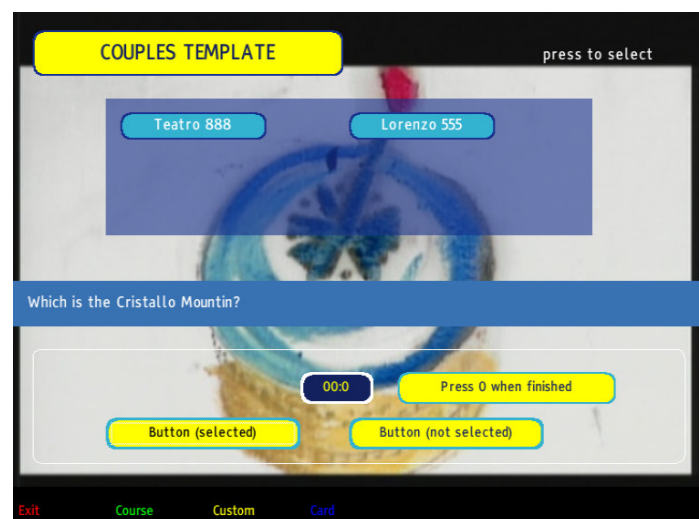

Figure 3: Instance of a Multimedia Page.

D. 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, multiple-choice 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. The games are considered as an integral part of a t-learning course as they could support the learning procedure involving a wide audience through challenges and engaging activities that are anyway able to meet the typical user need for relax and sympathy.

The games that can be included in a course are based on specific configurable templates and form a Java class package, which is structured as follows. Game Template is a Java class that abstracts general game functionalities. Subclasses of Game Template are the game typologies offered by the standard game

library. Sample game typologies include: Memory, Puzzles, Visual Quiz, Couples, etc. The Actual games played by the end-users will be instances of such subclasses.

From a software point of view, Game Templates is a hierarchy of Java classes. Game Template is the abstract superclass. The game typologies are the concrete subclasses.

The actual games running on the STB will be instances of such game typologies. Every game instance will come boundled with its own resources (i.e. images, animations, fonts, graphics, text, etc.).

The actual games running on the STB will be instances of such game typologies. Every game instance will come boundled with its own resources (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) and inserting contents (e.g. images, graphics, questions, animations, etc.). This can be done .manually, or, much more efficiently, thorugh an ad-hoc developed authoring tool, that we will describe in the next section.

Figure 4: Instances of Games.

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 by the script. A game will be able itself to recursively manage other games (i.e. building trees of games, according to the user preferences).

In Figure 4, sample instances of Quiz, Couples, Memory and Puzzle games are illustrated in the way they appear in the TV screen.

E. Personalization in t-learning courses

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.

III. 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, as described in the previous section, the property files of the templates and the resources (images, audio files, etc) involved, in a structure package. The whole production chain is illustrated in Figure 5.
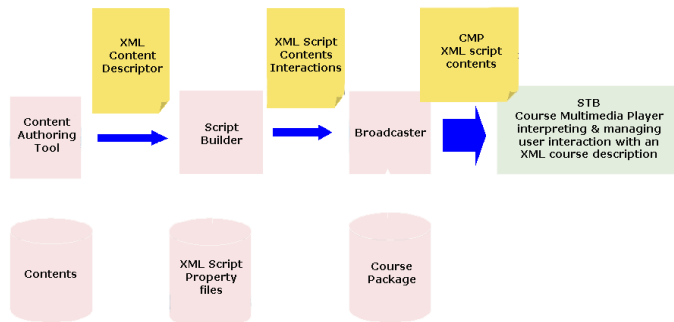
Fig. 5: Production Chain.

A. Authoring Tool

The proposed authoring tool is an efficient environment where the author is capable of creating educative courses for iDTV. It consists mainly of the two logical aforementioned parts: the course sequence and events definition part, where the structure of the course is defined based on event and the templates configuration part, which is responsible for the customization of existing templates like multimedia pages and games.

The tool provides a clear graphical interface (Figure 6) which supports features as drag and drop, image previews and object designing. The concept of the tool is based on the creation of Learning Objects in a form of XML script and on the configuration of existing templates.

A typical t-learning course is based on a learning object and can contain the following main items:

• Globals: used for global tags definition
• Variables: for the definition of variables used within the object. It is possible also to declare here all the variables that has to be used inside events
• Cards: used for creating and setting up cards which correspond to a set of templates as Multimedia pages, presentations and games, as described in section II
• Categories: used for creating and setting up categories, which correspond to different course paths, and those related events that define both the user interaction and how the course will be presented to the learner. As categories can be considered levels of difficulty (i.e. hard, easy) or other domain categories (i.e. History, Architecture, etc)

The part of the authoring tool dedicated to the creation of categories provides authors with an intuitive way for defining events lists (temporal or logical) linked to each category. 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.

Each category is graphically represented as a tree, this way giving to the author a visual and easy perception of the sequence and articulation of the events. Other supported features - as drag and drop cards, a temporal line where the events can be sequenced and the possibility to play a video content (.avi) – further allow for an easy creation of categories (Figure 6). As a result, the tool provides a "What You See Is What You Get"

(WYSIWYG) environment where authors will not have to be concerned by the internal structure and the constraints imposed by the XML script.
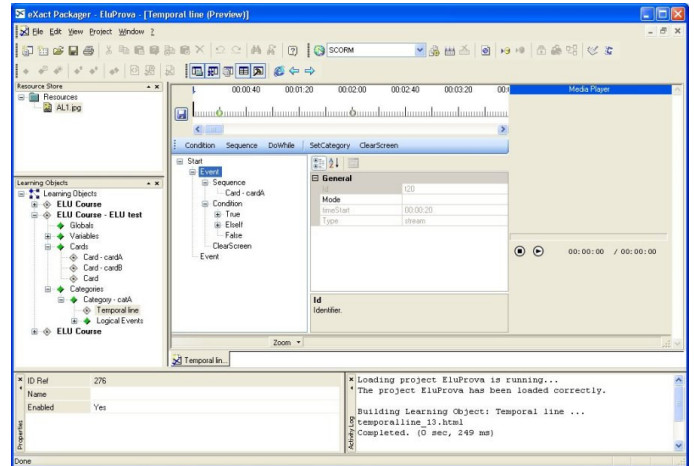

Fig. 6: Authoring Tool interface with events and timeline.

The part of the tool 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 and games, which can be configured appropriately in terms of appearance, fonts, screen mode and content (Fig. 7).

The idea of templates allows full reusage of existing code and material which is very important in the case of iDTV where the bandwidth for data transmission and the STB memory are limited. Hence, this authoring tool is a proper solution for t-learning as it has been developed to tackle the special needs and constraints of the iDTV and to offer educational oriented content creation.

The proposed templates have been designed in a way that satisfies the needs of the learners while the full configuration capability, which can also be based on conditions, is an added pedagogical value as it supports the creation of personalized courses.
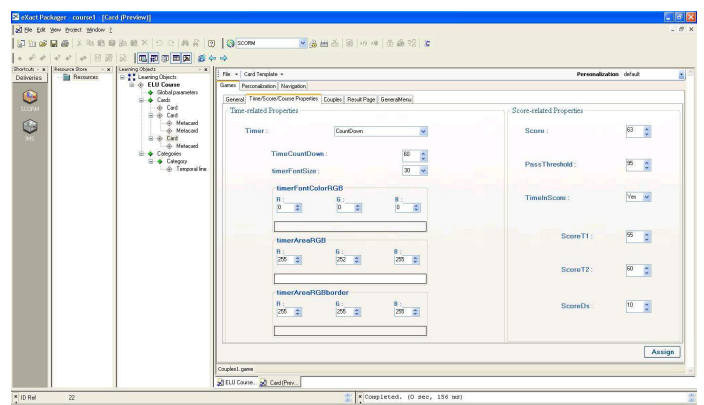

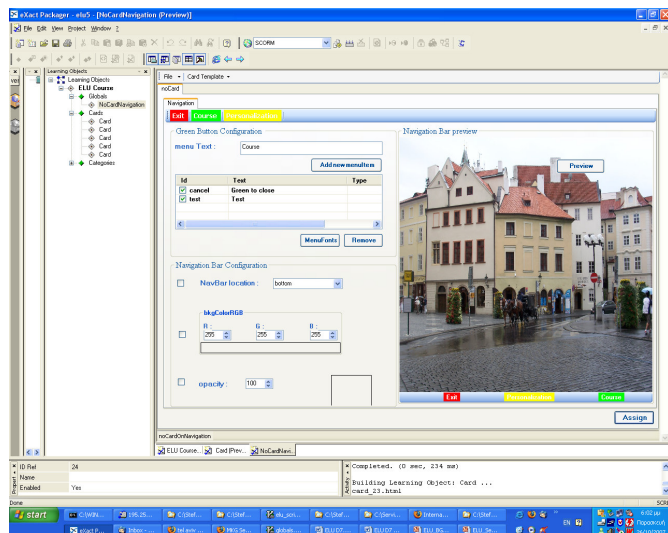Fig. 7: Authoring Tool interface with templates configuration.

Fig. 8: Authoring Tool interface with navigation bar configuration.

Furthermore the tool supports a standard way of designing cards and templates based on a predefined style which will set a default for many configuration values and let the author specify only the main ones.

This feature increases the usability of the tool as it makes it even more user-friendly and usable also by end-users who do not intent to waste a lot of time for creating the course and designing new multimedia pages.

In addition, the authoring tool provides a functionality that allows the author to customize the navigation bar by offering a proper interface with menus preview and add-remove menu items capability. Figure 8 illustrates the above functionality.

Pedagogists and teaching professional will be interested in such a tool as it provides a user-friendly and course oriented environment for the creation of iDTV supported pedagogical content without requiring programming skills. In addition, this tool could be a very convenient and innovative solution for professional content providers who are interested in creating learning content, as at this point there is no authoring software in the market that supports the development of education oriented content for iDTV.

IV. CLIENT SIDE

The Course Multimedia Player (CMP) is the t-learning client software that reads the data generated by the authoring tools to present the final course. It runs on the user MHP STB to allow interactivity between the learner and the application.

Two main steps in the playing process can be identified as shown in Figure 9:
- Building the interactive application.
- Executing the application with the video.

A. Building the interactive application

As explained in previous section, the xml script contains all required data 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 where

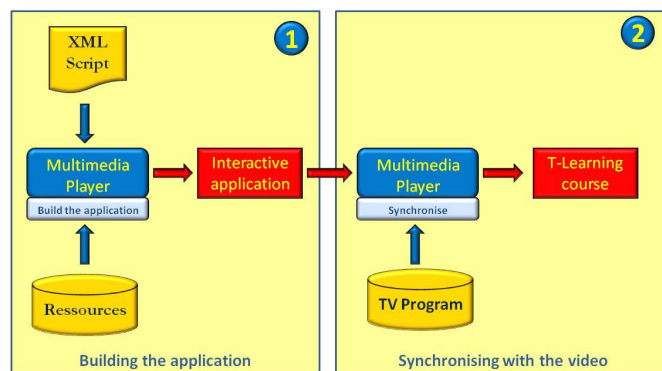navigation, personalized path based on the user profile, and synchronization with the TV-program are involved.

Fig. 9: Building the application in the Client Side.

B. Executing the Application with the video

When a T-Learning application is launched by a TV viewer, the Course Multimedia Player parses the corresponding scripts to instantiate the T-learning course. Subsequently, the following components are involved:
- Graphical Rendering Manager
- Navigation Manager
- Synchronization Manager
- Personalisation Manager
- Interactivity Manager

1) Graphical Rendering Manager

First, the Multimedia Player is responsible for the graphical interface. It displays all the course components and manages the video stream rendering. For instance, following the course creator choice or according to the Personalizer's values, each course screen can be displayed in 3 different modes: full Screen (FS), full Screen with TV picture (FSTV) and Stripe (S) (Fig. 10).

Fig. 10: Card Display Modalities: Stripe, Full Screen with TV Picture, and Full Screen Instance of a Multimedia Page.

2) Navigation Manager (personalised)

Interpreted at runtime, the course path can change following several rules from the personalization manager (0). This path can also be influenced by the synchronization manager or the user interaction. It allows a dynamic adaptation of the learning path. At a higher level, different paths may be specified by the author for different user categories, as shown in Fig. 11.

3) Synchronisation Manager

Synchronization is realized between the TV-program and the application. Indeed, regarding the timeline, some time triggers (they may correspond to events such as beginning of the first

part, end of the second part) are listened by the synchronization manager that can thus take adequate actions. For instance, start an application or change the section in a course.
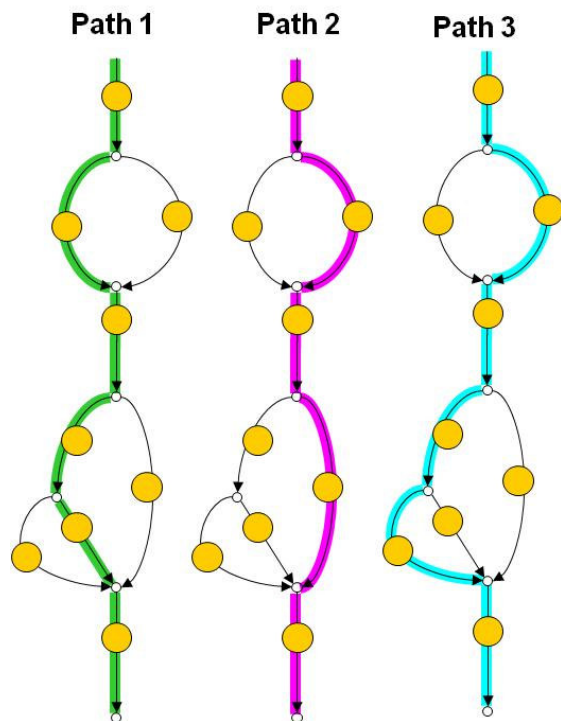

Fig. 11: Different Course paths paths – due to a user's personal values - for 3 different user categories. Circles correspond to cards.

If we consider a TV-program, which is related to tourism, with a first part dedicated to Paris and a second part dedicated to London, the synchronization mechanism will send an event to the course to change the relative data from Paris to London. The T-learning course may be thus context-aware regarding the video stream.

4) Personalisation Manager

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, age, etc.). Not only will this module be responsible for updating the Personalizer's data, but it will also orient the user by taking decision at runtime as explained in the subsection II-E.

5) Interactivity Manager

Finally, the Course Multimedia Player manages Interactivity. Interactivity can be seen in two levels:

• Local interactivity: it is provided by programmable components that enable the action of the user on visible parts of the screen and change the behavior/settings/appearence of the objects that are displayed, without the need of a return channel. 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 [c.f. General Framework Part] and the behavior of the displayed objects is modified through the reception of "commands" from the servers, application servers or broadcast server. 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.

V. RELATED WORK

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

An interesting approach was the technological framework for TV supported collaborative learning proposed in [7]. 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 [8] 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.

Considering the market approach, nowadays there are considerable tools available, both open source and commercial, like Composer [9] and Grins Pro Editor [10]. More specifically, the former is capable of creating content in NCL [11] language for Ginga middleware while the latter is a SMIL [12] 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.

VI. CONCLUSION

In this article a flexible framework for the creation and display of t-learning content on iDTV was presented. The proposed platform exploits the available iDTV technologies in order to provide support for the development and presentation of t-learning courses satisfying the pedagogical requirements.

During the course creation, a clear structure for the content is defined, while the template logic allows reusability, which is a big advantage at 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 the insertion of personalization, which is considered as an added pedagogical value.

This template-based approach allows the creation of flexible, standard courses that are interpreted by a multimedia course player running on a STB, The player interprets the course configuration script and accordingly presents the structure content to the viewer and manages her/his interaction through the remote control.

Another advantage of the proposed framework is the extensibility as it is possible to support services on demand, also

by exploiting the return channel of the STB, and including other kinds of contents, such as virtual avatars and more complex games.

REFERENCES

[1] Aarreniemi-Jokipelto Paivi, "T-learning Model for Learning via Digital TV", 16th EAEEIE Annual Conference on Innovation in Education for Electrical and Information Engineering (EIE), Lappeenranta, Finland, 2005W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[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.B. Smith, "An approach to graphs of linear forms (Unpublished work style)," unpublished.

[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-Lopez, Ana Fernandez-Vilas, Rebeca P. Diaz-Redondo, Jose J. Pazos-Arias, Jesus Bermejo-Munoz, "Adaptive Learning Objects for T-learning", in the 5th International Conference on Web-based Learning (ICWL), 2006.

[6] Blanco-Fernandez, Y., Pazos-Arias, J. J., Gil-Solla, A., Ramos-Cabrer, M., Barragans-Martinez, B. & Lopez-Nores, M., "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] 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.

[8] 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.

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

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

[11] 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.

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

[13] Enhanced Learning Unlimited. Retrieved October 18 2007, from http://www.elu-project.com.