Formalization and Verification of a Live Multimedia Presentation Model

Rustam Asnawi
Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia and Electrical Engineering, Yogyakarta State University, Indonesia

Wan Fatimah Wan Ahmad
Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia

Dayang Rohaya Awang Rambli
Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia

ABSTRACT
As the complement of the stored multimedia presentation, live multimedia presentation (LMP) is similar to the stored multimedia presentation except for the process of selecting, authoring, constructing and playing out the multimedia elements are done during presentation time. Unfortunately, there is no established model of LMP as well as in stored multimedia presentation with its SMIL. This paper proposes some basic concepts for developing any LMP application, and then from such concept we design a model of LMP using formal definition. Finally, to verify the model we used scenario-based and developed an application as a tool to accomplish the verification process.

Keywords
modeling, formalization, live multimedia presentation, multimedia application.

1. INTRODUCTION
Many researches in multimedia presentation area are recently concerned with stored multimedia presentation with some types of constraints [1-8]. These constraints are used to synchronize each multimedia data in order to attain the goal of its author. Mostly multimedia presentation researches are web-based, network-oriented and distributed environment [9-11]. Spatial and temporal constraints are the most interesting ones as the topic of the researches in this area [4, 12-13]. The modeling of the multimedia presentations is commonly approached by graph theory and formal definition to analyze the model [1, 3-4, 7]. Moreover, in stored multimedia presentation, between the author and the end-user are separated, and may never see each other directly before and after presentation. The author constructs and organizes the multimedia presentation into a presentation organization document at once, then publish it in web. This presentation document will be accessed by any end-users any time in distributed areas through the web.

Meanwhile, to present those multimedia data types in many different domain areas does not always use the stored multimedia presentation style. Definitely in some domain areas, users need to present those multimedia data in live. Users in other domain areas require the processes of authoring, constructing and managing the presentation of each multimedia data types performed *on the fly*. For instance, in class room or educational presentation, training, surveillance security, and entertainment, it needs a real time or *live* multimedia presentation (LMP) organization. The multimedia users in those domain areas often need to present multimedia data types in live without prior to construct a few lines of "program". In the LMP there are no additional efforts to build a stored and documented "script" with much declarative syntax before the process of presentation is begun. By this way, exactly the presenter is not work loaded by developing such multimedia documents.

Unfortunately there is no standardized model of LMP as well-established as like in stored multimedia presentation with SMIL. However, the rapid growth of multimedia applications for presenting multimedia in live presentation urges to develop a framework and standard model of the LMP.

This paper proposes a model of the LMP. The main objective of this paper is to provide a global description on the design and developing a framework of LMP applications. Based on such framework, a model of the LMP is designed and proposed with formal definition. To prove the concept, an application has been developed based on the proposed framework and model. Then verification was performed by utilizing the application to accomplish a scenario in the live multimedia presentation.
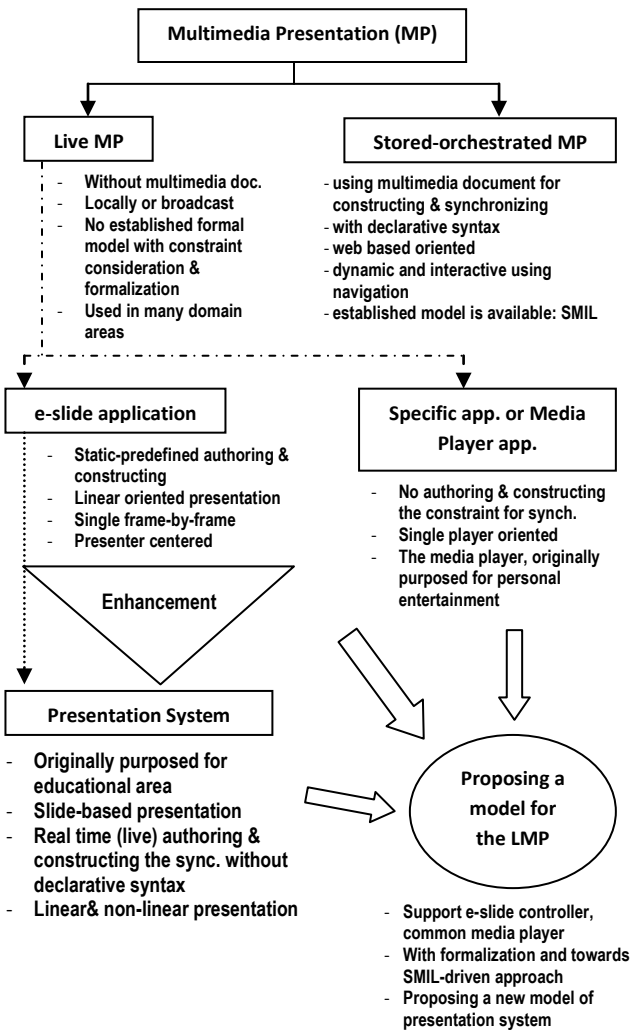
2. THE LIVE MULTIMEDIA PRESENTATION
Simple examples of the LMP are the conventional slideware application and common *media player* applications. Today, the most popular slideware application is Microsoft PowerPoint. The slideware is as a *simple* example of LMP since in the PowerPoint has no consideration about the temporal and spatial constraints for performing the synchronization between the presentations of each multimedia data in live. Temporal and spatial constraints in PowerPoint are pre-defined construction, which is done during authoring processes that prior to the real presentation. Therefore the slideware applications have behaviors and characteristics as follows [14-17]:

- *Frame by frame*, originally those presentation applications are designed based on slide oriented. Presentation is performed by displaying slide by slide or discrete presentation manner.

- *Static organization*. The organization of presentation document is difficult to be changed in flexible way during presentation time. Temporal, spatial and access constraints are already fixed inside presentation organization document. Whereas, in the middle of presentation, the presenter (author) sometimes needs to change or modify such presentation constraints (and contents) according to the real time condition. There is no chance with user interactivity to control both constraints.

- *Linear presentation.* As the consequence of slide-based orientation, the flow of the presentation is basically linear. Although it is possible to skip and jump forward or backward but it still requires more steps.

Those behaviors seem to be a drawback of such kind of LMP that caused some problems during the act of presentation. In common slideware application the pre-defined authoring behavior emerges some impacts. All multimedia data types (any video, audio, image and animation) that included or embedded inside its slides are viewed as an object of slide. It means, for an example if a slide consist of text, a video and an audio file, so by such application those multimedia data will be treated as a single frame (single slide). Both video and audio data will be played back in one slide without complete controller.

Fig 1: Global view of the multimedia presentation classification

From this, many researches already studied about the characteristic and the impacts of the slideware application particularly in educational area. As mentioned above, the static, linear, single frame-by-frame, and presenter-centered presentation are the characteristics of the popular slideware applications that have been seen by some researchers as the dominant drawbacks of such applications. Therefore many researches and studies have been done to overcome those drawbacks by improving and enhancing the slideware application that finally lead to emerge a *presentation system* [15-16, 18-21]. Presentation system does not only consider how some slides are made and presented to the audience but it involves how to deliver the detail content of the information to the audience [16].

In its progress, the presentation system becomes an important area of research topics in multimedia system. The research topics in presentation system are around overcoming those weaknesses of slideware applications. Then by designing the presentation system that support flexible, interactive, dynamic, audience-centered, and non linear presentation in order to conduct a professional class room presentation. However most of the current proposed models of the presentation system are slide-based oriented without considering the detail concepts of either the temporal or spatial constraints to perform the synchronization. It is reasonable since originally presentation system is designed to support the teaching learning processes in class room environment. Consequently, most of the proposed models of the presentation systems are only suitable to be applied in educational area.

Meanwhile, to present various multimedia data type in other domain areas (except education) people usually use *media player applications* or a specific multimedia application that special designed for such domain area. Since each domain area has specific characteristic and behavior, obviously such multimedia application would not compatible if want to be applied in other different domain areas. Lastly, Figure 1 illustrates the classification of today multimedia presentation in global view to simplify the understanding. The solid arrow line represents the classification of the multimedia presentation, the dotted-dash line points an example of application in real word and the dotted line represent an enhancement.

3. THE DEVELOPMENT OF THE LMP MODEL

According to the description in previous sections and also taking into account to the illustration in Figure 1, therefore, an idea to propose and develop a model for the LMP is urgently realized soon because of some reasons as follows.

- The rapid increment of the utilization of multimedia in any domain area that must be presented in live manner. Various multimedia applications to present multimedia data in live at many different domain areas are developed without standard model (framework).

- Already exist and standardized model in stored multimedia presentation (i.e. SMIL) that from this as the basis, it is highly possibility to derive a model for the LMP.

- By exploiting such proposed model of the LMP, it rises to develop a new model of presentation system that has different features and behavior than the others; particularly on constraints consideration.

- The development of the live multimedia presentation application in different domain area can be done more effective since there is a basic concept to design such application.

4. RELATED WORKS

As we know, some drawbacks and weaknesses of the conventional e-slide presentation tools available today encourage researchers to explore and improve such e-slide tools. Chiu *et al*.[18] developed ModSlideShow, a presentation system that is based on a discrete modular model to manage the slides to show onto multi displays. The slides can also be manipulated and annotated in a simple and flexible way. Liu et al.[21] created an EPIC. It views and treats all type of multimedia elements as a (hyper) slide. Epic considers the quality of views to the audience members through a model. Turban et al.[19-20] have proposed and developed an educational presentation system. This system works based on slide model with scenarios. It has more consideration with presentation in educational area. Some special features such as screen capturing and live video capturing are provided. Then, a flexible presentation tool for diverse multi monitors has been presented by Kurihara et al.[22] This presentation system will separate the contents from the views. For instance, if there are three monitors, the first for displaying previous slide, the second is for the current slide and the last monitor to show the next slide. Lastly, Lanir et al. [15-16] have introduced MultiPresenter. MultiPresenter is a presentation system with a slide-oriented approach. It considers presenting slides in a large and multi display environment. Various presentation styles and the way to deliver each component of slides are stressed and highlighted by MultiPresenter. It views that a slide which consists of other multimedia elements can be shown as another slide independently.

Most of the recent slide-based presentation systems have similarity, *i.e*. use a separate view mechanism between presenter and audience. Presentation system assumes the presenter as the author that has full authority to control, and maintain the flow of information while delivering it to the audience, while the interest of the audience is what and how the entire information-content can be received completely and easily. This difference implies that a presentation system should have a different view between presenter and audiences to bridge the gap.

On the other hand, some researches about synchronization in multimedia presentation with temporal and spatial constraints have been accomplished. Schnepf et al.*[23]* have introduced a Flexible Interactive Presentation Synchronization (Flip). Flip works by event-based model that supports inclusion of various media displayer and user interaction. Then, Bailey et al.[24] have proposed a multimedia synchronization toolkit, named the Nsync. According to Hakkoymaz and Ozsoyoglu [1] multimedia presentation is "a synchronized and, possibly, interactive delivery of multimedia data to users". Hakkoymaz [7] describes a multimedia presentation in reference to the presentation of multimedia segments with particular arrangement using a number of output devices. Sapino [25] has seen multimedia presentation as a collections of multimedia data provided with synchronization primitives, and possibly interactive feature with user. Tailor to Hakkoymaz [7] states that Multimedia presentation also considers the real time presentation of various data stream. Moreover, the multimedia presentation task for end-user presentation requires ability to exercise control via user's environment parameters.

Meanwhile, in the mid of 1998 the W3C first released the SMIL (Synchronized Multimedia Integration Language) as the standard on multimedia presentation over web [26-27]. Currently SMIL becomes a popular markup language in web-based multimedia presentation. As a family of XML, SMIL has some abilities to integrate and synchronize various types of media data to be presented in the client browser by considering the temporal and spatial constraint [28-29]. From this, regards to the LMP characteristics explained in the previous sections, a framework and model of the LMP is developed and proposed. Formal definition is used to simplify the analysis and verification processes.

5. ISSUES IDENTIFICATION

Before continuing into further discussion on the design of LMP model, this session will describe some issues related to the development of the concept of LMP. According to the previous description, several issues can be identified as follows:

5.1 Perspective differences between presenter and audience

The needs and interests of the presenter and the target user are exactly different and cannot be unified. Presenter requires delivering the information-content of any multimedia data to the target user through presentation applications with clear and true processes. Meanwhile, the target user has needs and interests to get and to understand such information contents of the multimedia data without any disturbances and noises [19-20]. Those disturbances of the target users' view may come from the confusion between the view of control panel and the video output layout since both are placed in same display or same region window (See Fig.2).
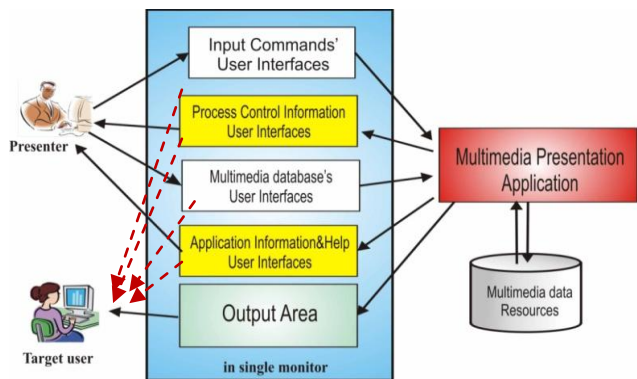

Fig 2: Conventional LMP application

From the presenter's view, a presenter definitely has some private information that actually is not worthy if known by target user. To solve the problem we proposed a separation mechanism between control panel user interface and the video output layout in the end-user presentation application (Fig.3).

By separating the video output area into particular display (monitor hardware) we can assure that the target user only watches the application output without being disturbed by several displays of control panel elements. Also the presenter can watch the control panel with full focus and attention without being disturbed by the output layout.
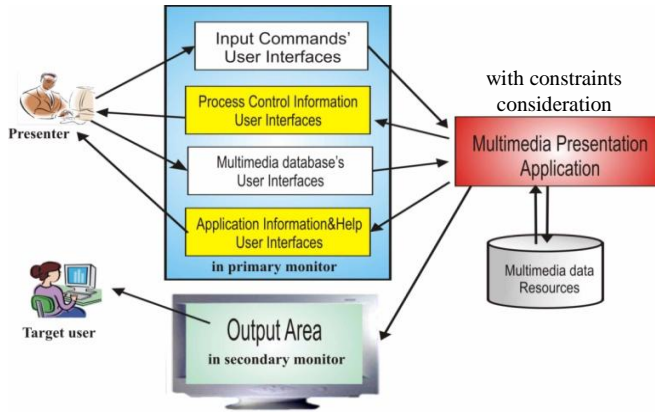
Fig 3: the different user's view in LMP

5.2 Multi domain areas of the user

The presenter may come from various domain areas and of course they have different characteristics and purposes [30]. Some users, such as a lecturer or teacher, need to present more than one media type simultaneously. For example, in a class room tutorial the presenter must present a complex content of presentation as comprise of video, audio, text/e-slides, and animation in one session of presentation.

Usually this task can be accomplished with e-slide application tools and a media player that support video, audio and animation. Since the presenter uses more than one application in one live presentation to play out those multimedia data, it is possible raising some problems. First, the presenter requires extra expertise to utilize the slideware and also media player applications. Second, while running some type of applications often distract the flow of such presentation. Third, because all applications are not integrated, the synchronization (that commonly based on spatial-temporal constraint) of the each element of presentation are neglected.

Furthermore, the common conventional media player is single player oriented. Therefore, it should be upgraded and optimized into a multiple-media player oriented or into multimedia player.

5.3 Problem on complex presentation

Mostly the slide-based presentation application contains various multimedia data (video, audio, and animation). The presenter often needs to control each multimedia data independently [31]. It is better and reasonable because embedded multimedia data inside a slide may create problems for the presenter e.g. file size becomes bigger, linked file does not exist, mismatch destination linked or the media file cannot be played back [32]. Moreover, the presenter may have to use several types of media player when he or she needs to present varying video file formats. It raises a new problem for inexperienced presenters.

This paper proposes an idea to address those issues. We have developed a concept and model for the LMP.

6. A FRAMEWORK OF DEVELOPING LMP APPLICATION

Exactly, today various types of the live multimedia presentation applications have been developed to fulfill the user's demand. Unfortunately, there is no established framework and model to

develop such applications as well as in stored multimedia presentation. Therefore, based on the explanation in previous sections, a framework to develop an LMP application is developed and proposed here.

Apparently multimedia presentation is not only how to present any multimedia data in front of the target users. Multimedia presentation is more than playing back any combination of text, picture, video or animation. It involves all processes starting from multimedia data resources until attending the data in the user's hand. Handling all constraints through the entire process for synchronization is the most important characteristic in multimedia presentation.

Thus based on the previous description, in order to meet requirements on realizing the LMP applications, there are four basic criteria that should be completed. The four basic criteria also reflect the main concept of LMP.

a. *Live-integrated controller*, the control of the play out of each multimedia data is performed directly and instantaneously. All types of multimedia data i.e. video, audio, image, text, animation and e-slide should have particular (independent) controller in an integrated manner.

b. *Two groups of users*, mainly all users that interact with the LMP can be identified into two groups. The first user's group is the author or presenter, a user that has importance and needs to present one or more multimedia data to the second group, the target user. The presenter has abilities such as authoring, constructing, synchronizing and playing out their presentation to the target user. The second user's group is the target user, i.e. one or more people as the objective of delivering the multimedia data that under the control of the presenter. Both presenter and target user can be a single or multiple persons. In very simplest case, each of the presenter and target users can be occupied by a single individual.

c. *View separation*, since the authoring, constructing and controlling of the presentation are done directly and instantaneously, thus it needs to separate between the author/presenter screen and the target user screen.

d. *Constraint consideration*. Temporal and spatial constraints are the two main constraints that must be applied in any type of multimedia presentation. To attain this proposed model, the formal definition is used to analyze those constraints.

Additionally, the above four main concepts of LMP also should support broadcast live presentation through local or internet network. By this way, it can support remote live multimedia presentation with integrated controller. Next, after establishing the concept of LMP, we can begin to design a model of the LMP.

7. THE PROPOSED MODEL OF LMP

To embody the concepts of LMP in real world, we have to design a model of the underlying architecture of such LMP. Liu et.al. [21] developed a presentation system called EPIC. They view and treat all types of multimedia data as a (hyper) slide. The presentation system is based on a model of visual fidelity which also considers the quality of views to the audience through the model. The proposed idea in this paper was initially inspired by EPIC but in reverse way. We assume that an e-slide (PowerPoint or Presentation) file can be considered as a

"metamedia" since it consists of some slides where each slide comprises of any multimedia data (media) file types (see Fig.4).

Therefore, an e-slide file can be thought as an independent multimedia data similar with other multimedia data (*i.e.* video, audio, animation, image and text). Consequently the e-slide should have a particular controller. Since each multimedia data type has special controller, then we integrate the individual multimedia data players into integrated multi-mode player.
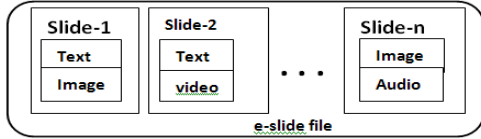

Fig 4: An e-slide file that can be considered as a metamedia

Axiom 1: By separating each multimedia data type with its particular controller, each type of the multimedia data can be played back independently [30-31]. Such conceptual model leads to develop a multiple media player or a multimedia player rather than merely a media player.

According to axiom 1, a model of the LMP can be designed and proposed. Figure 5 depicts the layout of the proposed model in conceptual level.
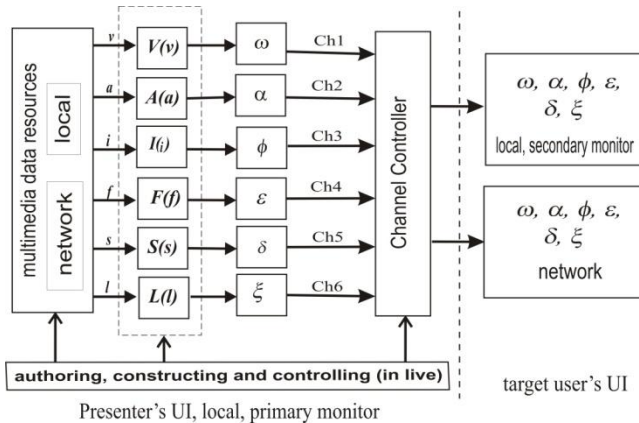

Fig 5: The proposed model in the conceptual level

In succession *V, A, I, F, S* and *L* are video, audio, image, animation, e-slide and live data streaming controller. Certainly, each player or controller has particular input and output data. The input data is the related file format with its player function. It can come from local (local storage: hard disk or CD) or network of multimedia data resources. The output is the data stream of related player function that needs hardware devices to actualize it. From this, each player can be viewed as a particular *function*. So using formalization, we can write serially $V(v)$, $A(a)$, $I(i)$, $F(f)$, $S(s)$, and $L(l)$ to represent the function of players (or controllers) for such related multimedia data. The output of each player is the related multimedia data streaming, consecutively for each player is ω: Video streaming, α: audio streaming, ϕ: Image streaming, ε: animation data streaming, δ: Slide data streaming and ξ for live video data streaming.

Each multimedia data stream has a specific *channel*. The term *channel* in this context is like the conceptual path for particular multimedia data stream. In this design model, channel-1 (Ch_1) is assigned to serve the video stream. The other channels (Ch_2, Ch_3, Ch_4, Ch_5, and Ch_6) are assigned to serve the other multimedia data streams, i.e. audio, image, animation, e-slide, and live video stream respectively. Except for Ch_2, all those channels will utilize display monitor to show its output as the end-point presentation to the target user.

Definitely, it requires a mechanism to control and manage those channel outputs in order to be displayed in the target user's monitor in the right way. Although those outputs can be displayed in the same monitor with the control panel UI, but according to the objectives of this research, all outputs will be displayed into the secondary monitor using the extended desktop or multi display feature.

7.1 Temporal Constraint Consideration
As described in the previous section, this concept of LMP utilizing some constraints while presenting each multimedia data element directly and instantaneously. In this paper we only focus on temporal constraint, since it has the most important role during presentation. Allen [33] introduced 13 types of temporal relationship (also cited by Agius [6]).

Table 1. Allen's [33] temporal relationships

Temporal relation	Notation	Inverse notation	Pictorial example
X before Y	<	>	XXX YYY
X equal Y	=	=	XXX YYY
X meet Y	m	mi	XXXYYY
X overlap Y	o	oi	XXX YYY
X during Y	d	di	XXX YYYYY
X starts Y	s	si	XXX YYYYYY
X finished Y	f	fi	XXX YYYYYY

In this model of LMP we refer to the above 13 temporal relationship to be the temporal constraints. Hence, each $V(v)$, $L(l)$, $I(i)$, $F(f)$, $S(s)$, and $A(a)$ has an individual and independent controller so that all types of the temporal constraint can be applied in such model of LMP.

However, after performing some experiments, a few types of such temporal relationship seem overlapping. The *meet (m)* and *meet inverse (mi)* are similar with *before* (<) and *after* (>) relationship. However we emphasize the difference between *before* (<) and *meet (m)* is, in *before* relation there is a bit space time between X and Y presentation, but in *meet* there is no space time (Y directly presented after X without delay).

8. FORMALIZATION OF THE SYSTEM CHARACTERISTICS
Assume that the each player and controller can be defined as a function that has main goal to control those multimedia data. As a function it has specific input and output variable. The input variable is the related multimedia data and the output is the

multimedia data stream that can be seen in the secondary monitor or a speaker (for Ch$_2$). Thus, the characterization of the proposed model can be analyzed from those functions as describe below.

8.1 Basic functions of the LMP

Let, $V(v)$ denotes video player or controller function that is defined for $v \in \{$ any video file types $\}$. So we have:

$$V(v) = \omega \tag{1}$$

with ω is the video stream output. Then we have the other basic functions as follows:

$$A(a) = \alpha \tag{2}$$
$$I(i) = \phi \tag{3}$$
$$F(f) = \varepsilon \tag{4}$$
$$S(s) = \delta \tag{5}$$
$$L(l) = \xi \tag{6}$$

Where:

$A(a)$: audio player function is defined for $a \in \{$ any audio file types $\}$ with output is audio streaming α. $I(i)$: image slide show or viewer function is defined for $i \in \{$ any image file types $\}$ with output is an image data stream ϕ. Animation player function: $F(f)$ is defined for $f =$ animation file such as flash file with ε is the animation data output. e-slide controller function: $S(s)$ is defined for input s is a PowerPoint file with δ is the slide show output. The live data streaming function: $L(l)$ is defined for $l \in \{$live video data streaming, live audio data streaming$\}$ with ξ is the live data streaming.

As we know, really any video file comprises of two parts i.e. video part and audio part. The video part is composed from many images (called video frames). So, it can be assumed:

$$V(v) = V(\nabla) + A(a) \quad \text{or}$$
$$V(v) = I(i) + A(a) \tag{7}$$

Where ∇ is video data without audio and $I(i)$ is continuous images (video frames) viewer and $A(a)$ is the audio part of such video file. We use "+" operator to refer a concurrent function operation between both operands (functions). Both operands and functions are interdependent, synchronized, and cannot be timely separated. Then, analogue for the animation file, it comprises of many animation frames (images). So,

$$F(f) = I(i) + A(a) \tag{8}$$

where $I(i)$ is continuous image viewer (animation frames).

Here, $A(a)$, $V(v)$, $F(f)$, and $L(l)$ are continuous functions, and $I(i)$, and $S(s)$ can be continuous or discrete/ discontinuous functions.

8.2 Multimode Player Function (Augmented-Union Operator)

In multimode player a presenter needs to activate more than one players or controllers simultaneously to present more than one types of multimedia elements to the target user.

For example, a lecturer (as a presenter) needs to present a video tutorial about PC interfacing using LPT1 parallel port to control a stepper motor through data parallel transmission. Before starting to play back the video, the presenter needs to display the

diagram of the concept of such interfacing system in an image file format. In the middle of the video presentation, the presenter needs to pause the video for a while and presenting an animation about the skew effect (the weakness in the data communication using parallel transmission) then back again to continue playing back the video tutorial.

Using our model of LMP the above tasks can be done by the presenter just by activating a few controller functions without considering the related temporal constraints. For that case, the steps to perform the task can be written:

$$I (i) < V(v) \, d \, F(f) :$$

activate image viewer "before" activating video controller, and activate flash controller "during" video controller is being activated.

While presenting $F(f)$, we prefer *during* (d) as the temporal constraint operator to *finishes* (f) since d has a really different temporal interval with f. In d after finishing the second function ($F(f)$) the first function ($V(v)$) would be still active, but for *finishes* (f) both functions are stopped at the same time.

Actually, the presenter is able to activate more than one of any basic functions of multimedia data player above. However, there are some important functions that have special purposes (consider Table 2). Those multimode functions are obtained based on the equations (1) until (8).

Here, introducing an augmented-union operator $\hat{U}$. Since, $V(v)$, $L(l)$, $I(i)$, $F(f)$, $S(s)$, and $A(a)$ have individual and independent controller so all the temporal constraint types can be applied in such a model of LMP.

Axiom 2: Let $P(m)$ is end-user LMP function which m is variable input of any multimedia data type. Thus,

$$m \in \{v, a, i, f, s, l\}$$

With regards to the temporal relationships as described in Table 1 (also as temporal constraint), it can be defined in a complete presentation notation in one among these:

$$P(m) \equiv V(v) \, \hat{U} \, A(a) \, \hat{U} \, I(i) \, \hat{U} \, F(f) \, \hat{U} \, S(s) \, \hat{U} \, L(l)$$

Where, $\hat{U}$ is the augmented-union operator with temporal relationship (temporal constraint). The $\hat{U}$ operator is a set of the 11 temporal relationships:

$$\hat{U} = \{ <, >, =, \, o, oi, d, di, s, si, f, fi \}$$

Table 2. Some important multimode player functions

Some Important Functions		Purposes
$V(v) \, \hat{U} \, A(a)$	(9)	Substituting the audio part of v, with other audio data from user preferences
$I(i) \, \hat{U} \, A(a)$	(10)	To create a new movie with images slide-show and particular background music

$F(f)\,\hat{U}\,A(a)$	(11)	Substituting the audio part of *f* with other audio data from user preferences
$S(s)\,\hat{U}\,A(a)$	(12)	Adding background music (sound) into e-slide data
$I(v)\,\hat{U}\,V(v)$	(13)	User prefers to listen the audio data of *V(v)* but dislike to see the video data. So, it will substitute the video part of *V(v)* with image slide show.

8.3 Advanced Player Function

Let τ is text data then we can compose:

$$L'(l) = L(l)\,\hat{U}\,\tau \qquad (14)$$

where $L'(l)$ is the live video streaming with text overlay. This function is purposed for real-time annotating in live video streaming. Beside that, we also have:

$$V'(v) = V(v)\,\hat{U}\,\tau \qquad (15)$$

Where $V'(v)$ is video player with text overlay. This function is purposed for inserting subtitles into a video file in real-time.

As we know, audio data commonly use stereo output channel, i.e. left and right channel. So we can write in detail for $A(a)$ as :

$$A(a) = A_L(a_L) + A_R(a_R) \qquad (16)$$

if the left and right channel in stereo audio data input is assumed in a united-dependent streaming audio channel (since it comes from one audio file), so we can use "+" operator to represent $a = a_L + a_R$.

But, other presenters need to have the left and right channel as independent controllers. So we propose in our model, the equation (16) can be manipulated to perform this function:

$$A(a) = A_L(a_L)\,\hat{U}\,A_R(a_R) \qquad (17)$$

If the audio data is mono, $a = a_L$ or $a = a_R$, so

$$A(a) = A_L(a_L)\,\hat{U}\,A_R(a_L) \qquad (18)$$
$$= A_L(a_R)\,\hat{U}\,A_R(a_R) \qquad (19)$$

Then, advanced functions can be derived:

If the audio data is stereo, $a = a_L + a_R$, so it can be:

$$A(a) = A_L(a_L)\cup A_R(a_L) \qquad (20)$$
$$A(a) = A_L(a_R)\cup A_R(a_R) \qquad (21)$$

These functions are purposed for audio channel assigning, where (20) is the left audio channel that is assigned to overwrite the right audio channel and (21) is the right audio channel that is assigned to overwrite the left audio channel. This function is useful to support audio channel removing and balancing such as in karaoke.

Furthermore, we can see that a video recording process actually can be assumed as an inverse function of a video player since:

$$V(v) = V_s \quad \text{and if} \quad V^{-1}(V_s) = v \qquad (22)$$

Next, from (22) we can use the inverse function $V^{-1}(V_s)$ to represent a recording process of any video streaming that will result in a video file (*v*) as an output. The video streaming can come from some resources such as live camera, video player output, or screen monitor.

9. E-SLIDES CONTROLLER FUNCTIONS ANALYSIS

The e-slide application such as Microsoft PowerPoint, originally only enables to digest all slides in single input (ppt) file. A *s* input (ppt) file definitely contains a collection of subsequent slides. So, the e-slide function *S(s)* can be written as:

$$S(s), \; s = s_1, s_2, s_3, \dots, s_n,$$

Where *n* = number of slides in an e-slide file.

In a PowerPoint file, all of its slides can be viewed as interrelated-dependent (as a part of *s*), then we have a united slide functions in a single e-slide file. So the formalization of the slide function can use the "+" operator.

$$S(s) = \sum_{i=1}^{n} S(s_i)$$
$$= S(s_1) + S(s_2) + S(s_3) + \dots + S(s_n) \qquad (23)$$

Thus, by considering the temporal relationships from table 1, the + operator in the equation (23) can be substituted with the "<" (*before*) operator. So, we have:

$$\equiv S(s_1) < S(s_2) < S(s_3) < \dots < S(s_n) \qquad (24)$$

This is a linear presentation that constitutes a common behavior in e-slide application tools. Since the slides of an e-slide application are originally designed as serial-single (frame-by-frame) presentation and single display oriented, so it is hard to do non linear presentation.

$$S(s_1)\cup S(s_2)\cup S(s_3)\cup \dots \cup S(s_n) \neq equation\,(23) \neq equation\,(24)$$

To overcome the problem, the e-slide file is extracted into a collection of its slide's thumbnails. Then, the slide's thumbnail are loaded into a slides gridline. To show a preferred slide, the presenter just double clicks on such slide's thumbnail on the gridline. This way makes each slides seem to have a particular-independent viewer.

Axiom 3: By converting each slides of an e-slide file into an array of its slide's thumbnails, then separating control panel UI

and the slide-show screen, make it possible to perform linear or non linear presentation. So, we have

$$S(s) \equiv S(s_1) < S(s_2) < S(s_3) < \dots < S(s_n)$$
$$\equiv S(s_1)\cup S(s_2)\cup S(s_3)\cup \dots \cup S(s_n)$$

Moreover, to perform cross-slides presentation between two or more e-slide files, we have an axiom as follows:

Axiom 4: By duplicating the implementation of Axiom 3, we will have two thumbnail-slides grids. Further, it can perform cross-slides presentation between two or more e-slide files. Let, $S_1(s)$ is e-slide controller function for the first PowerPoint file and $S_2(s)$ is e-slide controller function for the second PowerPoint file, so ideally:

$$S(s) = \sum_{i=1}^{n} S_1(s_i) + \sum_{j=1}^{m} S_2(s_j)$$

$\equiv S1(s_1) < S1(s_2) < ... < S1(s_n) \ < \ S2(s_1) < S2(s_2) < ... < S2(s_m)$

$\equiv S1(s_1) \cup S1(s_2) \cup ... \cup S1(s_n) \ \cup \ S2(s_1) \cup S2(s_2) \cup ... \cup S2(s_m)$

Where n = number of slides of the first PowerPoint file and
m = number of slides of the second PowerPoint file.

In the next session, we will describe the proof of our concept using our developed application (the IM-Player).

10. THE PROOF OF CONCEPT

As a tool to prove of our concept, a prototype of an application has been developed based on that proposed model. It is named IM-Player (Integrated Multimedia Player). IM-Player supports all of multimedia data types and can playback any of them independently. The entire multimedia data are grouped by its type. Each group have particular controller, so it has six controllers: video player, audio player, image viewer, flash player, e-slide viewer and live video streaming controller. Figure 6 depicts a screenshot of the main menu of IM-Player. It's always displayed at primary (presenter) monitor. Presenter can select one of those independent players comfortably and freely.
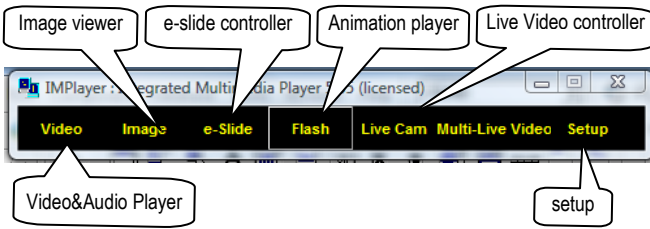

Fig 6: Screenshot of IM-Player's main menu.

Afterwards, we use a scenario to simplify understanding the system characteristic. This scenario is still related to the example in subsection 8.2 above. Let's say, before doing such example the teacher or presenter has to present two PowerPoint© files that are interrelated to each other. In the mid of presentation, while presenting slides of the first PowerPoint© file, the presenter needs to present two slides that exist in the second PowerPoint© file and then back again to the first PowerPoint© file. Normally, the presenter has to find the other files in their storage. It will somehow cause some disturbance to the audience and can interrupt their concentration as well as disturb the flow of the teaching-learning process. After that, really the presenter has complicated topic of presentation (performing the example scenario). Then the common problems will emerge: if these files are embedded in that slide, it makes the file size of PowerPoint©

bigger and then it will make the execution time become longer. Applying hyperlinks referring to these files often turns to be an ineffective solution and error-prone since if the destination file does not exist then it even raises another trouble in that presentation.

Furthermore, that scenario has some consecutive and concurrent tasks. The presenter will use the IM-Player as a tool to present all material with LMP concept. Figure 7 shows some serial screenshots while performing the tasks in that scenario. The left pictures are the displays of presenter monitor and the right pictures are the target user monitor.

First, the presenter (or the author) must prepare all the multimedia data files into the appropriate playlist. Then, the presenter starts the presentation by operating the e-slide controller $S(s)$. The presenter can select two PowerPoint files from the file list to be rendered. The two rendered PowerPoint files will be laid on the slide-gridline (Figure 7.a left). To present the slides to the audience, the presenter just double-clicks the selected slide in the slide-gridline, and then quickly it will be displayed into the target user monitor (Figure 7.a right). Using this way, the presenter can select freely each slide from the two PowerPoint files to be presented for the target user. This feature provides a non-linear way of presentation. Furthermore, the presenter can change the loaded PowerPoint file that currently exist in one of both slide-timelines with another PowerPoint file in the file list. This process is fast and invisible from the audience's view.

After that the presenter activates the $V(v)$ function where v is a video tutorial about interfacing method (Figure 7.c left). The output will be displayed in the target user monitor (Figure 7.c right). Then, while presenting the video, presenter paused the video player and activate the animation player $F(f)$ to display an animation about the skew effect (Figure 7.d left and right). While finishing the animation presentation, presenter can directly go back to the video player and continue playing back such video tutorial until the end of the video. Switching between each player is done quickly and so simple by single clicks in the main menu.
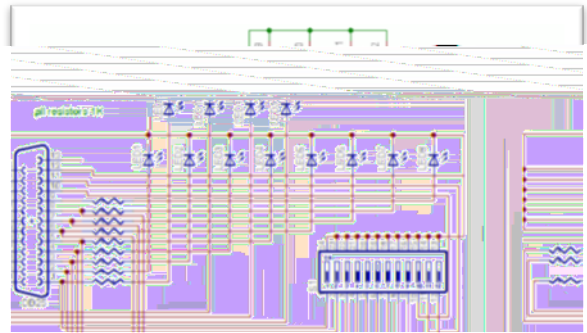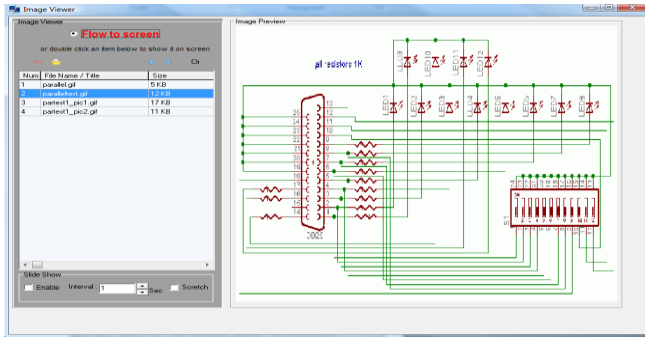
After finishing the slide presentation, the presenter chooses the image viewer $I(i)$ and double clicks the selected image in the file list (Figure 7.b left). The selected image will be shown in the target user's monitor (Figure 7.b right) and the presenter explains the picture for a while.
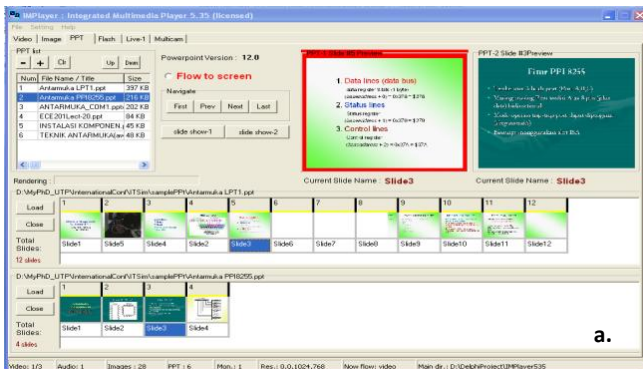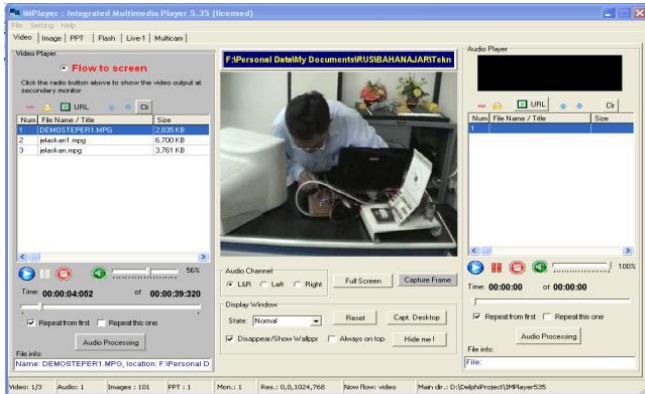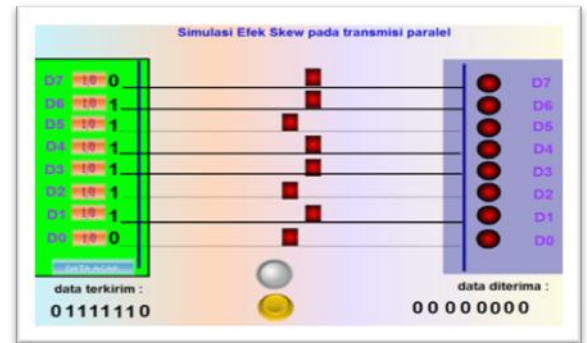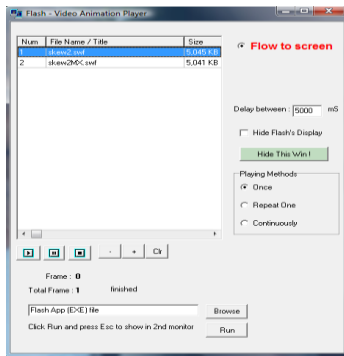
a. **Screenshot when performing** $S(s)$ **presentation with** $s = s_1, s_2$

b. **Screenshot when presenting an image in this scenario** $I(i)$.

c. **Screenshot when performing** $V(v)$ **presentation.**

d. **Screenshot when performing** $F(f)$ **presentation.**

Fig 7: Some screenshots while performing the scenario.

11. ANALYSIS AND DISCUSSION

To analyze the proof of concept, the timeline of such scenario presentation of each multimedia data is created as shown in Figure 8. The horizontal axis represents the time range from the starting point (t_0) through the end point of presentation (t_8). The vertical axis represents functions of each multimedia data type. In this scenario, there are four multimedia data types: e-slide, image, video and animation. Consequently there are four types of functions *i.e.* $S(s)$, $I(i)$, $V(v)$ and $F(f)$.
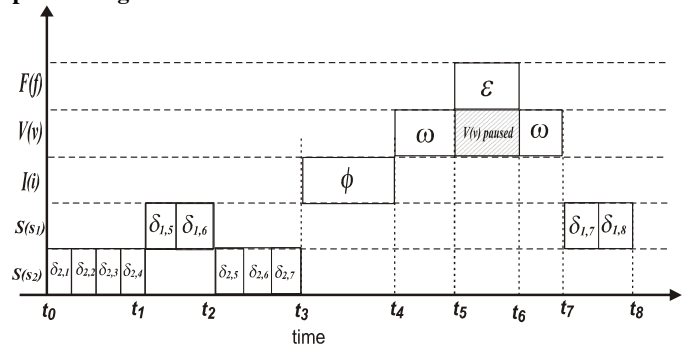

Fig 8: The timeline of such scenario presentation

The t_1, t_2, t_3,...,t_8 are the critical points where the presenter switches between the multimedia data presentation. From $t_0 - t_3$ shows e-slide presentation in linear and cross-slide presentation. In t_1, the presenter starts to show the 5th slide of s_1, and then in t_2 the presenter continues to show slides of s_2.

In advanced, really the non linear e-slide presentation can also be conducted by IM-Player since the control panel of the e-slide controller is as depicted in Figure 7.a (left). The result of some experiments on such e-slide controller has shown that the e-slide controller can be used to perform random e-slide presentation. Hence, from starting time t_0 through t_1 the presentation of slides $s_{2,1}$, $s_{2,2}$, $s_{2,3}$, $s_{2,4}$ can be done in either linear (sequential) or non-linear (random) presentation.

The image data was presented directly after 7th slide of S_2, in the interval $t_3 - t_4$. At t_4, the presenter started to play video tutorial until it paused at t_5 because the presenter need to show an animation file during a period of time t_5 - t_6. After the animation finished, the video presentation will continue again until the end of such video file (t_7). Then the presenter continues to present the s_1 slides. From this, obviously the timeline also represents an integrated and independent controller of each multimedia data.

Meanwhile, for constraint consideration analysis, at t_1, t_2, t_3, t_4, and t_7 there are an *after* or *before* temporal constraint. And from t_4-t_7 there is a *during* temporal constraint. Practically, the other temporal constraints can also be applied by IM-Player.

The IM-Player as an example of LMP application (developed based on such LMP concept), principally enables to prove the concept of LMP's model with the formal definition approach. Nevertheless, the constraint consideration should be defined in more concrete; moreover it should involve the spatial composition.

12. CONCLUSION

This paper proposed the formalization and verification of a live multimedia presentation (LMP) model. The LMP model involves the end-user of multimedia presentation where the user enable to organize, construct and synchronize the elements of presentation in a live presentation. Four basic concepts (framework) for developing any LMP application have been proposed *i.e.* live-integrated controller, two groups of users, view separation, and constraint consideration. Based on such concept, a model of LMP with formalization has been developed. To verify this model, a scenario based was used to simplify the understanding of such model's characteristics and behaviors. In order to prove the concept of LMP, an application (IM-Player) has been developed. Using IM-Player the scenario can be performed to complete the verification process.

Some possible significant contributions can be gained from this proposed framework and model as follows. First, by this proposed model of LMP, to understand the underlying architecture of LMP is easier and clearer. Second, since there is a definite framework and model on LMP, the development processes of any live multimedia presentation applications is more guided. Consequently, improving and enhancing such LMP applications become more effective and efficient.

In the future, the LMP model needs to improve its performance by taking into account the stored multimedia presentation standard i.e. SMIL (using SMIL-driven approach).

13. ACKNOWLEDGEMENTS

14. REFERENCES

[1] V. Hakkoymaz and G. Ozsoyoglu, 1996. Automating the Organization of Presentations for Playout Management in Multimedia Databases. in Proceedings of International Workshop on Multimedia Database Management Systems, Blue Mountain Lake, NY , USA pp. 28 - 36

[2] S. Wirag, 1997, Modelling of Adaptable MM Documents vol. 1309/1997: Springer.

[3] T. Lee, et al., "Querying Multimedia Presentations Based on Conctent", Knowledge and Data Engineering, IEEE Transaction, vol. 11, 1999.

[4] V. Hakkoymaz, et al., "Constraint-based Automation of Multimedia Presentation Assembly", Multimedia Systems, Springer-Verlag, vol. 7, 1999.

[5] B. Prabhakaran, "Adaptive Multimedia Presentation Strategies", Multimedia Tools and Applications, Springer, vol. 12, 2000.

[6] H. W.Agius and M. C.Angelides, "A Method for Developing Interactive Multimedia from their Semantic Content", Data & Knowledge Engineering, Elsevier, 2000.

[7] V. Hakkoymaz, "Multimedia Presentation Organization and Playout Management Using Intelligent Agents", Multimedia Tools and Applications, Springer, vol. 47, 2009.

[8] Paola Bertolotti and O. Gaggi, "A Study on Multimedia Documents Behavior: a Notion of Equivalence", Multimedia Tools and Applications, Springer, vol. vol. 33, 2007.

[9] G. O. Nevzat Hurkan Balkir, 1998. Multimedia Presentation Servers: Buffer Management and Admission Control. in Proceeding international workshop on Multimedia Database Management Systems, Dayton, OH pp. 154-161.

[10] G. O. Nevzat Hurkan Balkir, "Delivering Presentations from Multimedia Server", The VLDB Journal — The International Journal on Very Large Data Bases, Springer-Verlag New York, Inc., vol. 7, pp. 294-307, 1998.

[11] A. Y. Chang, 2005. Design of Consistent SMIL Documents for Distributed Multimedia Presentation Using Temporal Algebra. in International Conference on Parallel and Distributed Systems11th pp. 189 - 195.

[12] M. Adiba and J.-L. Zechinelli-Martini, 1999, Spatio-Temporal Multimedia Presentations as Database Objects: Springerlink.

[13] Timothy K. Shih and A. Y. Chang, June 1997. Toward a Generic Spatial/Temporal Computation Model for Multimedia Presentations. in IEEE International Conference on Multimedia Computing and Systems, Canada.

[14] J. E. Kjeldsen, "The Rhetoric of PowerPoint", Seminar.net International Journal of Media, Technology & Lifelong Learning vol. 2, 2006.

[15] J. Lanir, et al., 2008. Observing Presenter's Use of Visual Aids to Inform the Design of Classroom Presentation Software. in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems CHI08', pp. 695-704.

[16] J. Lanir, et al., 2008. MultiPresenter: A Presentation System for (Very) Large Display Spaces. in Proceeding of the 16th ACM International Conference on Multimedia, Vancouver, British Columbia, Canada.

[17] E. R. Tufte, 2003, The Cognitive Style of PowerPoint: Graphics Press, Chesire, CT, .

[18] P. Chiu, et al., 2003. Manipulating and Annotating Slides in a Multi-Display Environments. in Proceedings of INTERACT'03, pp. 583-590.

[19] G. Turban, 2007. Categorization of Educational Presentation Systems. in Proceedings of the international workshop on Educational multimedia and multimedia education, Augsburg, Bavaria, Germany, pp. 5-10.

[20] G. Turban and M. Muhlhauser, 2007. A Framework for the Development of Educational Presentation Systems and its Application. in Proceedings of the international workshop on Educational multimedia and multimedia education., Augsburg, Bavaria, Germany, pp. 115-118.

[21] Q. Liu, et al., 2005. Framework for Effective Use of Multiple Display. in Proceedings of SPIE International Symposium ITCom on Multimedia Systems and Applications VIII, , Boston, Massachusetts, USA.

[22] K. Kurihara and T. Igarashi, 2007, A Flexible Presentation Tool for Diverse Multi-display Environtments vol. 4662 Part I. Berlin: Springer Berlin.

[23] J. A. Schnepf, et al., 1996. Building A Framework for Flexible Interactive Presentations. in Proceedings of Pacific Workshop on Distributed Multimedia Systems, Hong kong.

[24] B. Bailey, et al., 1998. Nsync – A Toolkit for Building Interactive Multimedia Presentations. in Proceeding of the sixth ACM International Conference on Multimedia, Bristol, United Kingdom,, pp. 257-266

[25] M. L. Sapino, 2002, Multimedia Presentations Databases (Extended Abstract): SOFSEM 2002, LNCS Springer Berlin 2540.

[26] D. C. A. Bulterman, "SMIL 2.0 - Part 1: Overview, Concepts, and Structure", Multimedia, IEEE Journal, vol. 8, pp. 82 - 88 2001.

[27] Lloyd Rutledge, et al., "Anticipating SMIL 2.0: the developing cooperative infrastructure for multimedia on the Web", Computer Networks, Elsevier Science, vol. 31, pp. 1421–1430, 1999.

[28] L. Rutledge, "SMIL 2.0 : XML for Web Multimedia", Internet Computing, IEEE vol. 5, pp. 78 - 84, 2001.

[29] W3C. (2010, November 2010). Synchronized Multimedia Integration Language (SMIL 2.0) - [Second Edition]. Available: http://www.w3.org/TR/2005/REC-SMIL2-20050107/

[30] R. Asnawi, et al., 2010. Development of Multipurpose Integrated Multimedia Presentation System Based on the CCM Algorithm. in Proceedings of International Conference on Computer and Automation Engineering, Singapore, pp. 716-720.

[31] R. Asnawi, et al., 2010. Unifying Multimedia Player and Presentation System with Utilizing Dual-Display Feature. in Proceedings of The 4th International Symposium on Information Technology Kuala Lumpur, Malaysia, pp. 70-75.

[32] R. Asnawi, et al., 2009. Design Architecture for IM-Player as a Tool for supporting Visual Education Presentation. in Lecture Notes in Computer Science-IVIC Kuala Lumpur, Malaysia, pp. 78-89.

[33] J. F. Allen, "Maintaining Knowledge about Temporal Intervals", Communications of the ACM, vol. 26, pp. 832-843, 1983.