

# Web-based Personalization and Management of Interactive Video

Rune Hjelsvold  
Siemens Corporate Research, Inc.  
755 College Road East  
Princeton, NJ 08540, USA  
+1 609 734 6518  
runehj@scr.siemens.com

Subu Vdaygiri  
Siemens Corporate Research, Inc.  
755 College Road East  
Princeton, NJ 08540, USA  
+1 609 734 3687  
subuv.scr.siemens.com

Yves Léauté  
Siemens AG  
Hofmannstr. 51,  
D-81359 Munich, Germany  
+49 89 722 45488  
yves.leaute@icn.siemens.de

## ABSTRACT

This paper describes an application framework for providing and managing personalized, interactive video on the web. The application framework enables content providers and aggregators to stream personalized content to end-users. The server stores video and multimedia content that is structured and profiled. The content profile is matched against an end-user's interest profile to determine which parts of the video to include and what hotspots and hyperlinks to provide to the user. The server generates SMIL files that represent the personalized content and returns their unique URLs to the client when responding to end-user requests.

This paper also describes how the application framework is implemented by the HotStreams system. HotStreams is implemented using the Java 2 Enterprise Edition (J2EE). HotStreams is eCommerce and micropayment enabled and provides the means for content providers to specify the value of the content down to the level of individual video clips. This allows the content provider to charge for the parts of the video that the end-user actually viewed or received.

The HotStreams application includes web-based management tools that enable content managers to manage multimedia objects, to compose, structure, and profile the content, to create and profile hotspots and hyperlinks, and to manage pricing information and advertisements. The management tools are implemented in the form of Java applets that run inside a web browser.

## Categories and Subject Descriptors

H.5.4 [Hypertext/Hypermedia]: Architectures, Navigation

## General Terms

Management, Design, Human Factors

## Keywords

Interactive video; video personalization; SMIL; dynamic content generation; media asset management; micro-payment; J2EE

## 1. Introduction

Arguably, television has been the most important medium for delivering information and entertainment to consumers for many years. The World-Wide Web is gradually replacing the TV as the dominant vehicle for information and entertainment distribution. Video and media streaming products for wireline connections (such as [1], [2], [3], [4]) and for wireless connections (such as [5]) enable pervasive access to television and video content. The World-Wide Web does not only have the potential of replacing the television. World-Wide Web technologies can also be integrated with traditional television to provide interactive TV. Interactive TV is a fast growing area where also Internet bodies such as W3C [6] and IETF [7] are active.

The World-Wide Web offers service quality features that traditional television cannot match:

1. **Availability of information object.** Television has a predefined schedule. If you cannot watch the six o'clock news, you may have to wait until the 10 o'clock news to see the latest news. A well-managed web server, on the other hand, is available 24 hours a day and should be able to return any object being requested at any point in time.
2. **Interactivity.** Television does not provide much of interactivity – other than switching between different channels. A well-composed web site, on the other hand, will provide the user with a variety of hyperlinks and other tools for interaction.
3. **Personalization.** Television is based on broadcasting where the same content is delivered to all viewers at the same time. More and more, web sites offer users ways to define their own profile for personalization.
4. **Easy to become a “publisher”.** Television and video production is currently mostly for the professionals because it requires expensive software that is mostly designed for professionals. Most Internet Service Provider offers web hosting services and there are many easy-to-use and inexpensive web authoring tools available.

The main strength of the television compared to the World-Wide Web is that it allows the viewers to receive information *without* requiring continuous user interaction. A user who wants to get the latest news will have to go through a number of web pages and select several video streaming objects to view “television” news on the web.

In this paper we discuss a system that combines the main strengths of television with the World-Wide Web quality features we

mentioned above, i.e., availability of information objects, interactivity, and personalization. We decided to refer to the technology as *interactive video* rather than *interactive television* to emphasize that the end-user may be using a computer or a PDA to access the content and that the content is streamed over the Internet. Interactive video is a subset of interactive multimedia and hypermedia technology where the video content defines the timeline of the presentation and is thereby the driving force. We will discuss the rationale behind this constraint in the next two subsections.

## 1.1 Interactive Video Application Areas

In this section we will discuss a few application areas for interactive video, but it is by no means exhaustive. We have picked a few areas that will help us identify some of the features that an interactive video system needs to provide.

We have already mentioned how interactivity will add value to television news services. The world is becoming increasingly more complex and global. As a result of this, users' interests are getting more diverse than before. A European stockbroker working for an international company on Wall Street in New York is usually not satisfied with the television news services provided for the "average" New Yorker. Personalized television news would enable the stockbroker to include more financial and international news stories into the video stream. The stockbroker may find certain news stories important and may want to pause the video to access the World-Wide Web to find more details about the given event.

The rate of change today makes it difficult for everyone to keep up to date on the new technologies. This creates a need for people to constantly acquire new skills and expertise. Distant education using video and multimedia material is a good way to provide training *when* people need it. Personalized, interactive video will allow the student to customize the training material to fit their needs and knowledge. Assume that a professor is giving a lecture on image processing and recognition. A student graduating in computer science may need to understand the details of common image segmentation while a user of image processing software may only need to know what the limitations of the various techniques might be.

Video streaming technology is becoming a new tool for companies to distribute business TV. A personalized, interactive video system will increase the effectiveness of these videos by allowing employees to select the information that may have an impact on their work and responsibilities.

## 1.2 Related Work

### 1.2.1 Hypervideo and Interactive Video

Several digital video systems [1, 2, 3, 8, 9, 10] and document models [11, 12] have been developed for interactive video and hypervideo. The term *hypervideo* was first proposed by the creators of HyperCafe [8]. Hypercafe was developed as a new way to structure and dynamically present a combination of video and text-based data. Other systems, such as the Networked Hyper QuickTime prototype [10], takes a *video-driven* approach where the video playback provides a timeline with which other media (such as slide shows) are synchronized.

The Synchronized Multimedia Integration Language (SMIL) [13] developed by the World-Wide Web Consortium provides a

standard for *scripting* multimedia presentations that can be used to develop interactive video and hypervideo on the web. The streaming server developed by Real Networks, Inc. [1], for instance, implements SMIL on top of the Real-Time Streaming Protocol and provides a standard way to deliver interactive video on the web.

### 1.2.2 Personalized Content

Personalization and adaption of information to the meet the users needs and interest is a very active research area. Most of this work is related to text and hypertext personalization. A few projects have also been investigating personalization of multimedia content: Not and Zancanaro [14] discuss content adaption for audio-based hypertext; Papadimitriou et al. [15] describe implementation of personalized entertainment services on broadband network; and Merialdo et al. [16] describe a system for personalizing television news to optimize the content value for a specific user.

### 1.2.3 Hypermedia Authoring and Presentation Management

There are many hypermedia authoring systems and techniques discussed in the literature and being commercially available. This includes the GRiNS [17] editor for SMIL, the Microsoft Media Producer [18], MediaLoom [19], the HyperProp system [20], and the authoring tool implemented by Auffret et al. [21]. These systems give the author a high degree of freedom – and responsibility – in designing and organizing the hypermedia document.

## 1.3 Interactive Video Application Features

Any information system needs to provide *delivery* features to be used by the viewer to consume content as well as *management* features that will be used by the service manager to manage the service and the content. Based on results from the most important delivery features can be summarized as follows:

- ❑ **Direct access to segments.** Interactive video systems need to provide users with tools to directly access segments within the video. A news video, for instance, should provide the user direct access to each of the news stories.
- ❑ **Multimedia object timing.** Synchronization and timing of media object is a basic and challenging part of authoring interactive presentations that are supported by authoring tools. An interactive video system needs to support the creation of synchronization points in the video stream that initiates loading or presentation of images, web pages, etc. A training video, for instance, may be accompanied by a slide presentation and the slide transitions need to be synchronized with the video showing the professor talking.
- ❑ **Hotspots/temporally defined hyperlinks.** An interactive video system also needs to handle hotspots and hyperlinks that are temporally defined. Given such features, a financial news story, for instance, may have a hyperlink to a web page holding the latest financial indices and headlines.
- ❑ **Personalization.** An interactive video system further needs to provide personalization of video content as well as other multimedia objects and hyperlinks. A corporate video, for instance, may announce a new product. Employees of technical departments may benefit from learning the technical specification of the product while employees of the

marketing and sales department may need to know how the product is to be priced.

- ❑ **Micropayment support.** The content provider may have invested money and efforts in producing interactive content and may want to ensure a proper return on this investment. A pay-per-video model will not always be enough. Micropayment [22] will allow the content provider to charge for individual parts of a video being delivered to the user. In a training video, for instance, a student may be looking at an interactive video explaining MPEG encoding. The user may have indicated to the system that his mathematics skills are not deep, and a clip from another professor’s presentation of the Fourier transform may be inserted. Obviously, the second professor would want to be assured that the appropriate fee is collected.

An interactive video delivery system must include a management system for composing and managing content. The management system obviously needs to provide mechanisms that can be used to manage all the delivery features properly. An interactive video management system should have the following characteristics:

1. **Easy-to-use.** Multimedia publishing is experiencing a similar breakthrough as paper-based publishing experienced with desktop publishing. You had to be a professional with expensive tools to put together high quality presentations. The video capturing, encoding, and streaming systems make it easy for end-users to generate video. It is still quite challenging, however, to put together compelling, interactive multimedia presentations. An interactive video application should provide easy-to-use tools for creating and managing interactive content.
2. **Easy-to-brand.** The content provider should be able to “brand” the content in their own way without having to be expert designers. It would be advantageous if the branding of the page is separated from the content such that content can be put together by someone without web page design skills.
3. **Remote management.** The management system should allow content managers to access the system remotely – without having to install highly specialized content management software or hardware.

## 1.4 Organization

This paper is organized as follows: Chapter 2 describes an application framework or template for interactive video applications. Chapter 3 describes the architecture of HotStreams – an interactive video system that implements the proposed framework. Chapter 4 describes in more detail the system’s interactivity and delivery features. Chapter 5 describes the system’s management tools, while Chapter 6 concludes this paper.

## 2. A Framework for Interactive Video

The interactive video application framework proposed in this paper defines a user interface for interactive video playback. The approach taken is similar to the video-based approach used in the Networked Hyper QuickTime system. The interactive video client is running inside a web browser and uses several *frames* for displaying data to the end-user: One for displaying the interactive table of contents, which provides a textual summary of each of the sections of the video; one for displaying interactive video content;

and one or more for displaying other media objects, such as images, illustrations, slides, etc.

The framework is based on a video-driven content model, similar to the one described by Auffret et al. The content model is illustrated in Figure 1. The model is comprised of the following elements:

- ❑ **Video sequence element (clip).** The video consists of a sequence of video clips called sequence elements. Sequence elements within an interactive video may come from the same or from different video streams.
- ❑ **Video section.** Contiguous sequence elements are grouped into sections. A section typically represents a meaningful unit of content – e.g., a news story.
- ❑ **Video hyperlink.** There may be hotspots within the video. Hotspots have a spatial *and* a temporal range. Two hyperlinks are shown in the example model in Figure 1. The temporal range is illustrated by the grey shading. The broken arrows indicate that the target object for the hyperlink will be loaded in one of the application’s content frames.
- ❑ **Synchronization point.** Synchronization points are temporal locations within the video that defines some transition or change of content one of the application’s content frames. Synchronization points are shown as small dots on the figure.
- ❑ **Section title.** The title of the section, appearing in the table of content.
- ❑ **Section abstract.** A textual description of the content of the section, appearing in the table of content.
- ❑ **Section URL.** A URL that identifies the beginning of the section within the video production. The arrows in Figure 1 pointing from the entries in the table of content to the interactive video section indicate the presence of section URLs.

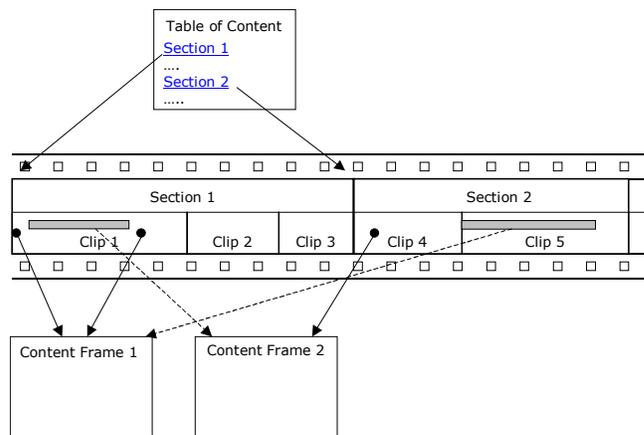


Figure 1 - The Interactive Video Content Model

### 2.1 Interactive Video Interface

The proposed application framework enables content providers to design their specific user interfaces without modifying the application logic. Figure 2 shows one example application interface where the table of contents frame is located in the lower left part of the figure. The RealNetworks G2 player is located in the frame above. There is only one additional frame for other

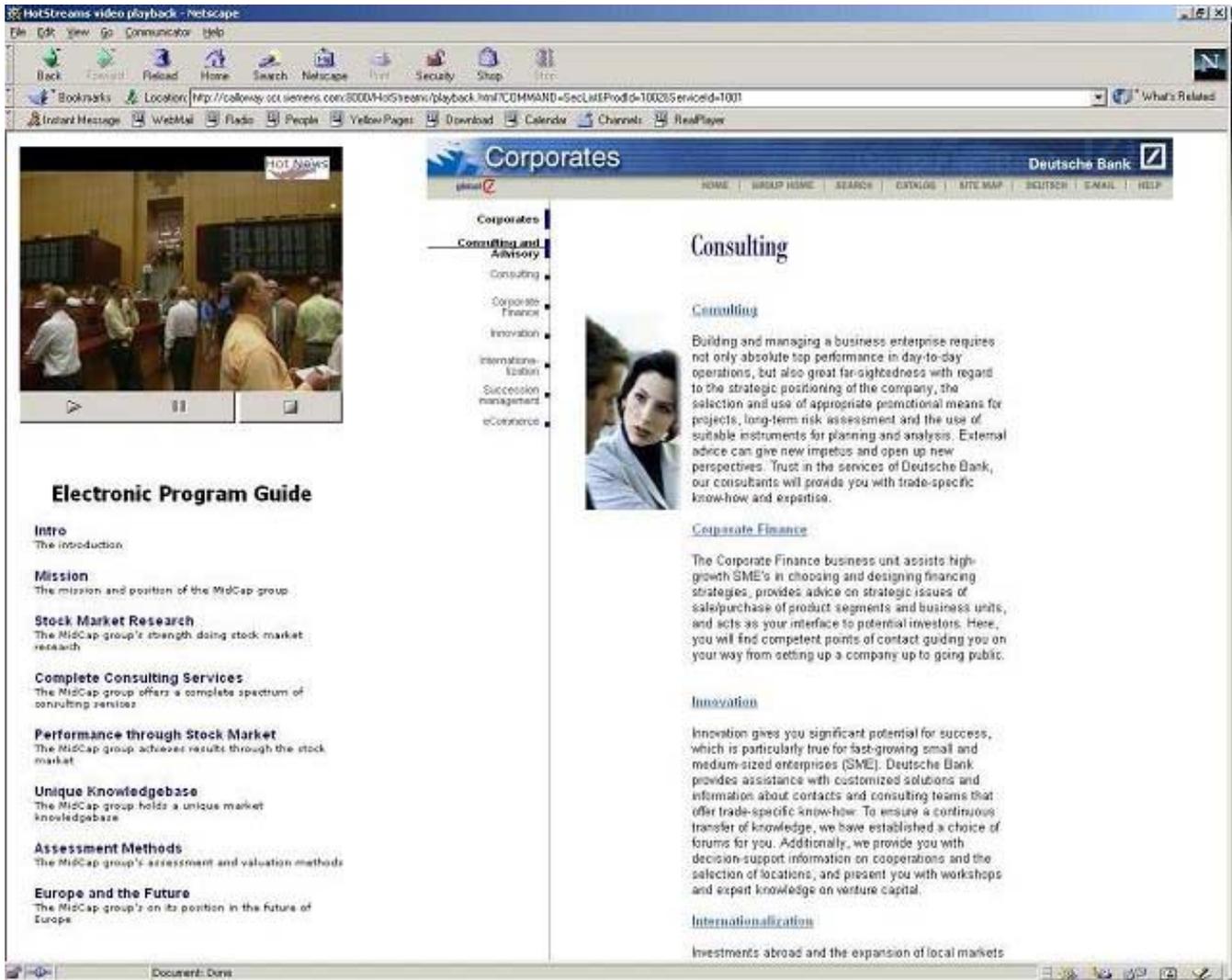


Figure 2 - A Sample Interactive Video Application Interface (Content Courtesy of the Deutsche Bank)

multimedia objects in this example, the frame to the right on the figure. The framework gives the application access to the information to be displayed in the various frames and for connecting the functionalities of the frames together – for instance to load a given web page into the frame to the right when the user activates a hyperlink.

## 2.2 Personalization

The application framework is based on a personalization model in which sequence elements, hyperlinks, and synchronization points are profiled using vectors of descriptors. The application framework provides the means for the content provider to define the descriptor schemes and the descriptors that will be used for matching the content against the user profile. A descriptor scheme may be hierarchical or unordered. The descriptors in a hierarchical descriptor scheme are ordered hierarchically. The rating system for American movies is an example of a hierarchical scheme. Any video clip rated G (general audience) can legally be shown to an R (restricted) rated audience. The framework fully supports hierarchical descriptor schemes.

The personalization process consists of the following steps:

1. The end-user passes an interest profile to the server and selects the interactive video of interest. The interest profile consists of one or more descriptor values for the schemes defined in the system. The user enters the profile by manually selecting the proper values.
2. The server accesses the database to retrieve all the sequence elements and scenes in the video that matches the user's interest profile.
3. The server similarly accesses the database to retrieve all the hyperlinks and synchronization points in the video that matches the user's interest profile.
4. The server determines how many video commercials to insert, selects the proper in-points in the video stream for inserting the commercials, and selects the ones that match the user's interest profile and the content profile.
5. The server generates a table of contents for the personalized version of the video.

### 3. System Architecture

The HotStreams system implements the application framework presented in this paper. It is built on top of the Java 2 Enterprise Edition (J2EE) [23] platform to provide a robust scalable system for the content provider to manage and deliver personalized presentations. HotStreams uses a multi-tier web based architecture consisting of a web tier, middle tier, backend database tier and streaming platform. The web tier consists of JSPs that help in producing dynamic HTML, servlets for controlling the management and personalization workflows, java applets along with HTML and graphic pages. The middle tier is comprised of the Enterprise Java Beans (EJB) with the business logic in the session beans and the entity beans reflecting the database. The data used for personalization is stored in backend databases and HotStreams delivers the personalized video content from streaming video servers.

HotStreams is integrated with a payment system to allow for payment transactions for viewing content from a tiny video clip (e.g., to watch a clip related to your favorite sports team from your evening sports wrap up) to entire movies. Currently, HotStreams system is integrated with Brokat X-Pay system [24] which complies with the Secure Electronic Transaction (SET) [25] standards. The Brokat SET system is integrated with HotStreams through payment EJBs and servlet. The SET components include X-Pay wallet, X-Pay server and X-Pay gateway and accessors.

#### 3.1 Personalization Server

The personalization server components run within the context of a J2EE application server (see Figure 3). The application server is providing the scalability, security and transaction processing required for a large-scale e-commerce system. The HotStreams components developed are compliant with the J2EE specifications and are tested on the J2EE Reference Implementation (RI). The core functionality of the HotStreams server lies in the session EJBs that provide the personalization of the video content based on the user criteria. Other functionalities, such as dynamic generation of SMIL files and advertisement insertion algorithms, are also encapsulated in session beans.

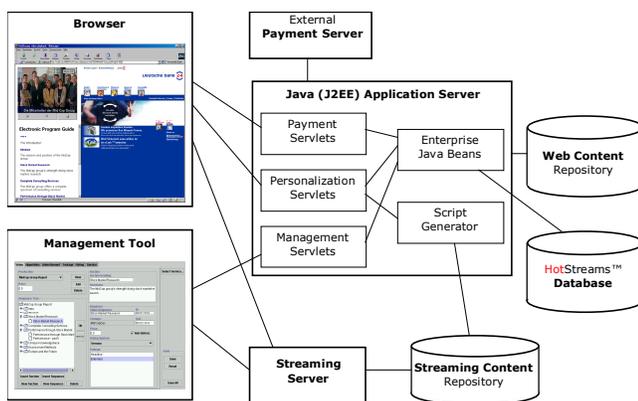


Figure 3 - HotStreams System Architecture

The personalization server interfaces with the web tier using servlets and communicates using serialized objects and XML on top of HTTP. This allows for applets to access the server through corporate firewalls. The servlets help in the workflow control by accepting requests from the clients (JSP or applets) and

forwarding them to the appropriate EJBs in the middle tier. The data is encapsulated in serialized objects and sent back and forth using HTTP. The payment servlet interfaces with the X-Pay server for payment transactions and for keeping a transaction history.

The transaction history not only keeps track of the payment but could also be used for tracking the user experience of the personalized presentation. Currently a transaction involves tracking events for a particular session of the user. Most e-commerce systems are used to trade services and hard goods. A system like HotStreams, however, is used to trade multimedia information. Such systems need to provide security mechanisms that protect the multimedia information from being accessed by non-paying users. HotStreams is handling this through an interaction between the payment servlets and personalization servlets.

The personalization files generated in SMIL consist of the filtered video clips the customer has requested along with the appropriate hyperlinks and graphics to be shown during the presentation. After the personalization query is run through the database, the SMIL generation EJB parses a SMIL template file and uses XML Document Object Model (DOM) functions to create nodes, traverse the hierarchy and change the content of the nodes. The user can easily manipulate template file for setting parameters for viewing the video like how the video should be presented on the screen and for different types like PAL, NTSC, etc. The personalization features are described in more detail in Chapter 4.

#### 3.2 Asset Management Server

The primary function of the asset management server is to provide management functionality. The asset management server contains the management service applet that is downloaded to the web browser when the content manager accesses the management server site and the management servlets (see Figure 3). The management servlet is retrieving and updating data in the database by accessing relevant EJBs.

#### 3.3 Streaming Server

The G2 server from Real Networks is used as the streaming server in the current version of HotStreams because it supports SMIL streaming. The personalized SMIL files are directly saved into the streaming server's content repository by the script generator.

#### 3.4 Payment Server

As the content provider (hotel chain, business news service) might wish to charge the user for viewing certain content. HotStreams provides for a pricing structure of fine granularity where the content provider could charge for tiny clips to large videos like movies. Moreover, there could be several payment systems like subscription where the users can subscribe to a channel or have a pay-per-view system similar to cable TV today. The main difference being that the user pays for the personalized presentation and only for the content one is interested in. As a HotStreams production is composed of sections that are further decomposed into sequences, one can associate a price from the level of sequence to that of production or to the entire service itself.

The user might charge the presentation to a particular account possibly associated with a corporate accounting system or be able to do micro payments. HotStreams has integrated with Brokat SET system as described earlier.

## 4. Delivering Personalized, Interactive Video

This section describes how a HotStreams server creates and delivers personalized, interactive content to the end-user. The current implementation of the system is creating SMIL files that represent the personalized content. In this section, we will describe how the content provider may customize the SMIL layout, how the outcome of the personalization is mapped to SMIL constructs, and how the table of contents is created.

We will start by revisiting the personalization workflow that we briefly presented in Section 2.2. Figure 4 shows how the personalization module and the script generator interact to generate a personalized version of the interactive video in the form of a SMIL file. The following subsections will describe the personalization process in more detail.

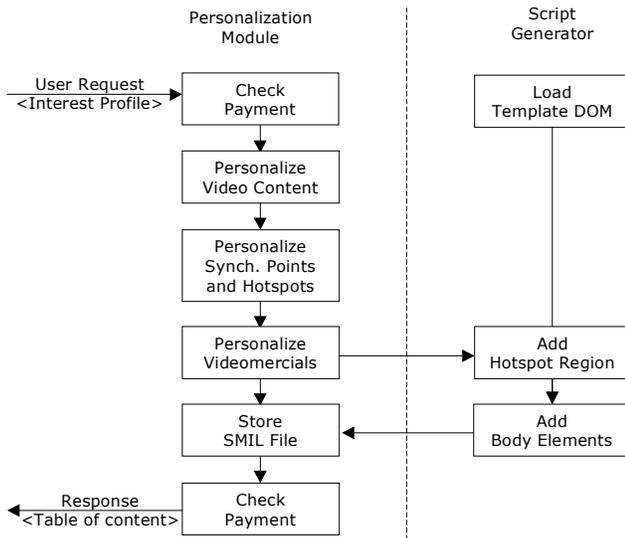


Figure 4 - The Personalization Process

### 4.1 Customizing the SMIL Layout

SMIL files are logically divided in two parts – the layout part and the body part. A HotStreams server will add SMIL body elements to a personalized SMIL file. The layout part, however, is defined when the application is configured. A HotStreams server loads a template SMIL file when starting up. The template file should define the layout element and leave the body element empty. The following code sample represents a typical template file for HotStreams – it defines a 176 by 144 pixel video region with the fill attribute set to the fit value.

```
<?xml version="1.0"?>
<smil>
<head>
<layout>
<root-layout height="144" width="176" />
<region id="vcr" top="0" left="0" height="100%"
width="100%" fill="fit" z-index="4" />
</layout>
</head>
<body>
</body>
</smil>
```

The content provider may change the application layout, for instance, to resize the video region or set a different fill attribute, by just modifying the template SMIL file and restart the server.

### 4.2 Personalized Video Content

An interactive video is composed of a sequence of video elements, grouped into sections. The aim of the personalization process is to select the sequence elements and the sections that match the end user's interest profile. The *personalization module* forwards the sequence elements and sections that "survived" personalization to the *script generator*. The script generator adds elements to the body of the resulting SMIL file as illustrated in the following SMIL sample:

```
<sec id="section_0" title="Intro"
abstract="The introduction">
<video src="rtsp://theServer/thePath/theFile1" region="vcr"
clip-begin="0ms" clip-end="30000ms" />
<video src="rtsp://theServer/thePath/theFile1" region="vcr"
clip-begin="75000ms" clip-end="90000ms" />
<video src="rtsp://theServer/thePath/theFile2" region="vcr"
clip-begin="0ms" clip-end="30000ms" />
<video src="rtsp://theServer/thePath/theFile3" region="vcr"
clip-begin="0ms" clip-end="60000ms" />
</sec>
```

Each section of the interactive video is represented as a `<seq>` element. The server generates an identifier for each of the sections to enable direct access to the start of the section. The server also fills in the `title` and the `abstract` attributes with values retrieved from the database. The server concatenates contiguous sequence elements and adds one `<video>` element for each such concatenated group. The server sets the values of the `clip-begin` and `clip-end` attributes with the actual database values.

### 4.3 Personalized Hotspots

The sequences that passed the first step of the personalization may be anchor points for hotspots and synchronization points. These will be matched with the user's interest profile. Unfortunately, SMIL does not support synchronization of media being displayed outside the SMIL player. Hence, synchronization of media to be displayed in the web browser has to be done externally. The hyperlinks that pass the personalization are forwarded to the script generator. The script generator needs to add a layout region for the hotspot image – if not already existing. Video clips without hotspots will be generated as illustrated above. Video clips that contain hotspots will look like this:

```
<par>
<video src="rtsp://theServer/thePath/theFile1" region="vcr"
clip-begin="0ms" clip-end="30000ms" />

<anchor href="http://www.siemens.com/" show="new" />
</img>

<anchor href="show="new"
"&&theFrame&&http://theSrvr/thePath/thePage.html" />
</img>
</par>
```

Two hotspots will be shown during the playback of this video clip. The first one will be shown in the region named `aiu_1` during the first 15 seconds of the clip. The second will be shown in the region named `aiu_2` throughout the whole clip. The `href` attribute of the second `<anchor>` element is “decorated”: The browser will interpret the text appearing between the two pairs of ampersands as the name of the frame in which the hyperlink is to be loaded. Hence, the Siemens web page will be loaded in an *external* browser window if the user clicks on `theIcon1` and the `thePage` will be loaded in the frame named `theFrame` if the user clicks on `theIcon2`.

#### 4.4 Personalized video commercials

The insertion of video commercials is currently based on a value. The application page designer may define a fixed amount of video commercials to be inserted in each video stream or may allow the end user to select. In both cases, the amount of video commercials to insert is determined by a currency value. Each video commercial is also given a price value, and the server will select as many video commercials as necessary to bring the total value to the amount selected by the end user. The server will insert video commercials at one or more of the predefined insertion points (see Section 5.2). The server will also select video commercials that match the profile of the video content.

#### 4.5 Personalized Table of Content

The personalization module creates a Java Bean object that gives the Table of Content JSP access to information about each of the sections in the personalized video, i.e., the name of the section, the abstract, and the URL that can be passed to the SMIL player to start playback from the beginning of the section. The URL is in the form: `rtsp://theSrvr/thePath/theFile.smi#section_nn`, where `nn` is the index of the given section.

### 5. Media Asset Management

The application framework proposed in this paper contains a complete set of management tools for assembling, profiling, and managing interactive video content. We already discussed the need for management tools briefly in Section 1.3. We identified three key requirements that an interactive video management system should fulfill. The management tools we have developed address these requirements in the following way:

1. **Easy-to-use.** The management tools have been designed in a modular way. Management functions are divided into coherent groups and each group of functions can be accessed from one management tool panel (see Figure 5). It uses user interface widgets like directory trees, drop-down selection boxes, etc. that most users will find familiar.
2. **Easy-to-brand.** The management tool allows the content manager to worry about how to compose and profile the content and does not need to know how the application's frames are designed.
3. **Remote management.** The management tools are implemented as a set of applets that interact with each other and with the SMIL player. Hence, any authorized user may use the management tools remotely.

Figure 5 shows the management tools. The user interface is divided in three major parts: The *service manager* applet is to the left on the interface. This applet provides all the management

functions the user needs to manage the service and the content. The embedded RealPlayer® from Real Networks, Inc. [26] is shown in the upper right part of the interface. The *playback control* applet to the lower right is interacting with the RealPlayer.

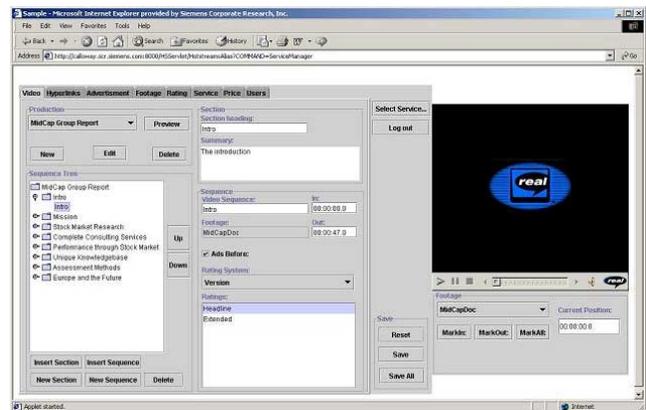


Figure 5 - HotStreams Management Tools

The playback control can be used by the content manager to set in and out points in the video. The playback control retrieves the current play position from the RealPlayer when the content manager clicks either button. The corresponding value is automatically set in the service manager applet.

The communication between the management tools and the server is carried by HTTP. Currently the applet-servlet protocol is based on passing of serialized Java objects. We will in the following describe the protocol in terms of XML elements because XML is a convenient and precise way to describe the content of the objects being passed.

The asset management protocol defines four types of applet requests:

1. **List** messages are used to request individual objects or sets of objects from the server. These messages are typically invoked when the applets are loading and are querying the server for meta-data.
2. **Add** messages are passed when the applet requests the server to store a newly created object for the first time.
3. **Update** messages are passed when the applet requests the server to save modifications to existing objects.
4. **Delete** messages are passed when the applet requests the server to permanently delete one or more objects.

#### 5.1 Media Content Management

The management tool is used to “register” video files in the database. The content manager can register video files by using the *footage panel*. The server transfers file and directory information to the panel such that a directory tree of all available files can be shown to the content provider. The content provider may associate any of the available files with a *footage* object. A footage object is an abstract representation of a video file. Multiple files can be associated with the same footage objects. Different files may store the video content in a different format, bit rate, or location than the other source files for the same content. This feature can be used to provide a content that can be streamed to a diverse range of devices, such as a PDA connected to a mobile network, a remote PC connected to 56 K modem, and a set-top box connected to ADSL.

Using World-Wide Web terms, the footage is a resource with a unique URN that may have to exist under one or more URLs. The footage panel is shown in Figure 6.

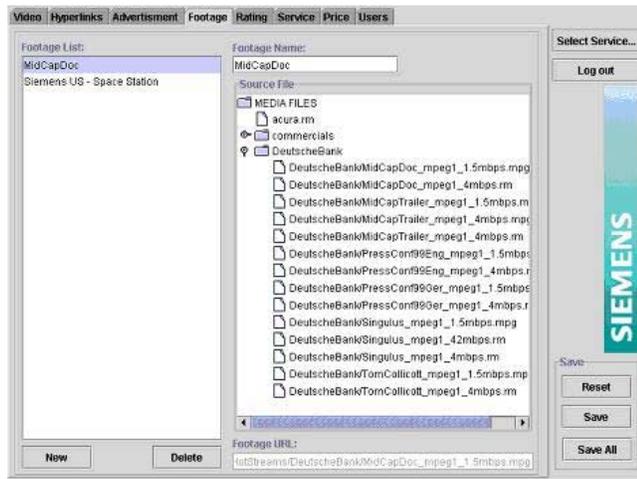


Figure 6 - Footage Panel

The following XML element shows sample footage information being passed between the applet and the servlet. The footage contains some shots from New York City in CIF (352x288) PAL format (25 fps).

```
<footage id="NY_2000_10">
  <name>New York City shots, Oct. 2000</name>
  <duration>302500</duration>
  <sourceFiles>
    <sourceFile fileNo="0">
      <url>rtsp://theServer/thePath/theFile.mpg</url>
      <height>288</height>
      <width>352</width>
      <bitrate>25.0</bitrate>
    </sourceFile>
  </sourceFiles>
</footage>
```

## 5.2 Video Composition and Profiling

The video panel is the main panel for creating, structuring, and profiling video productions. The panel is shown in Figure 7.

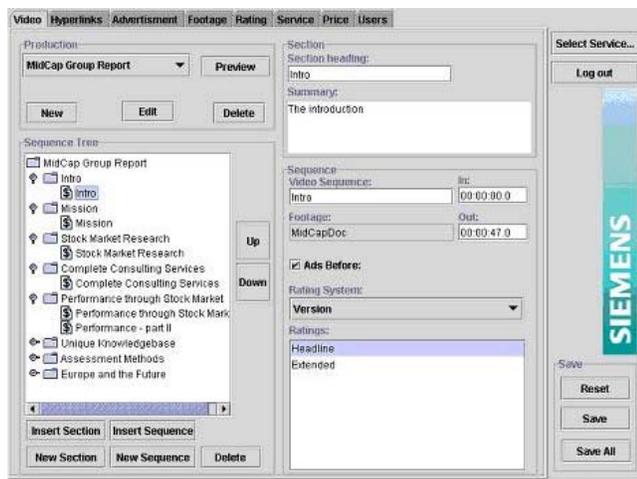


Figure 7 - Video Panel

The left-hand side of the panel is used to create and structure the video production. The content manager may add or delete sections and sequence elements and may modify the order in which sections and sequence elements appear.

The upper right hand side of the panel is used for the selected section's title and abstract. The server will retrieve these strings to create the table of content entry for the section (see Section 4.5). The content manager may add or delete sections and sequence elements and may modify the order in which they appear.

The lower right hand side of the panel is used to describe the individual sequence elements, i.e., to what footage it belongs, the in (clip-begin) and out (clip-end) points, the rating (profiling) information, and whether the beginning of the sequence element is a good place for insertions of video commercials. The dollar sign appearing in the sequence element icons in the tree indicates that the *Ads Before* checkbox is being checked for the given sequence elements.

The following XML element shows sample production information being passed between the applet and the servlet. The production has only one section that contains two sequence elements. The second sequence element is specifically targeting Norwegian tourists. Video commercials might be inserted in front of the first sequence elements but never between the elements. The XML sample also specifies pricing information; both sequence elements are delivered free of charge.

```
<production id="NY_Guide">
  <name>New York City Travel Guide</name>
  <sections>
    <section id="NY_Guide_Sec0" orderNo="0">
      <title>Arriving Manhattan by train</title>
      <abstract>Penn Station is located on 34th Street.
      </abstract>
      <seqElements>
        <seqElement id="NY_Guide_sqE10" orderNo="0">
          <name>Intro</name>
          <adsBefore checked="true" />
          <footage ref="NY_2000_10" />
          <in>0</in>
          <duration>20000</duration>
          <price>0</price>
          <ratings>
            <rating id="type_tourist" />
            <rating id="country_all" />
          </ratings>
        </seqElement>
        <seqElement id="NY_Guide_sqE11" orderNo="1">
          <name>Norwegian Consulate</name>
          <adsBefore checked="false" />
          <footage ref="NY_2000_10" />
          <in>185000</in>
          <duration>15000</duration>
          <price>0</price>
          <ratings>
            <rating id="type_tourist" />
            <rating id="country_Norway" />
          </ratings>
        </seqElement>
      </seqElements>
    </section>
  </sections>
```

</sections>  
</production>

### 5.3 Managing Interactivity

The content manager will use the hyperlink panel (see Figure 8) to create hyperlinks and specify profiling information.

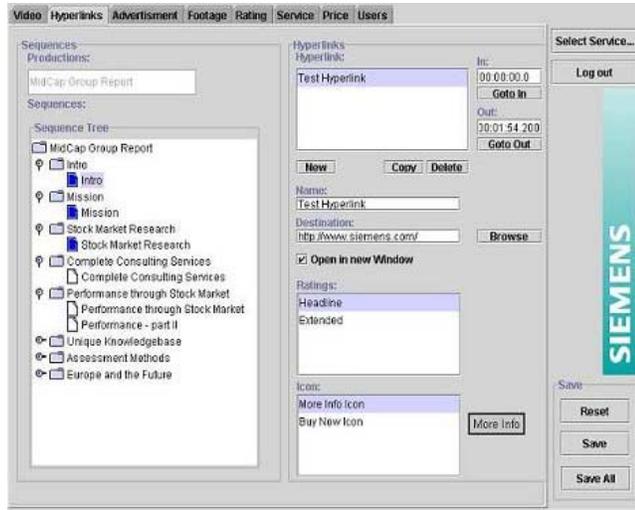


Figure 8 - Hyperlink Panel

The left-hand side of the panel is used to visualize the temporal range of the selected hyperlink. The sequence elements during which the hyperlink will be displayed are shown in the sequence tree with dark colored icons.

The right-hand side of the panel provides means for creating hyperlinks, defining their appearance and target, and setting profiling descriptors. The panel also enables the content manager to specify in what frame the target object is to be loaded. In this specific configuration, only one frame is defined by the application (see also Figure 2). Hence the tool provides a checkbox for the content manager to check when a specific target is to be loaded in a new browser window.

The following XML element shows sample hyperlink information being passed between the applet and the servlet. The hyperlink is covering 20 seconds of one sequence element. It will be shown to all kinds of users from the US. When shown, it will be linked to the `More_Info` icon that will be located in the `Upper_Right` region, which have been created in the database during application setup. When clicked, the hyperlink will load the Grand Central Terminal home page in an external browser window.

```
<hyperlink id="NY_Grand_Central">
  <name>New York Grand Central Station</name>
  <url>http://www.grandcentralterminal.com/</url>
  <startSeqEl ref="NY_Guide_sqEl0" />
  <startOffset>0</startOffset>
  <endSeqEl ref="NY_Guide_sqEl0" />
  <endOffset>20000</endOffset>
  <region ref="Upper_Right" />
  <icon ref="More_Info" />
  <targetFrame />
  <ratings>
    <rating ref="type_all" />
  </ratings>
</hyperlink>
```

```
<rating ref="country_US" />
</ratings>
</hyperlink>
```

### 5.4 Video Commercial Insertion

The content manager will use the advertisement panel to create hyperlinks and to classify the video commercial. The panel is shown in Figure 9.

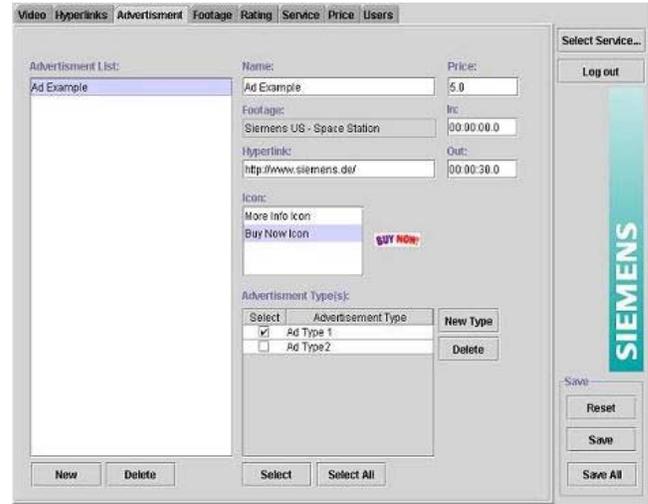


Figure 9 - Advertisement Panel

The panel provides the means for the content manager to create new video commercials from existing footage. The content manager may also specify the URL of the advertiser's or the product's home page and a hotspot icon to be used for creating a hyperlink for the advertiser in the video. The content manager may further define the value (price) of the selected video commercial and may assign a advertisement type classification.

The following XML element shows sample video commercial information being passed between the applet and the servlet. The 30 seconds commercial can be found in the beginning of the `Siemens_S10` footage. Its price value is 0.05 US dollars and it can be used where commercials for electronics or cellular products are appropriate or for insertion in content targeting young people.

```
<advertisement id="Siemens S10">
  <name>Siemens S10 Cellular Phone</name>
  <price>0.05USD</price>
  <sections>
    <section id="Ad_S10_Sec0" orderNo="0">
      <title />
      <abstract />
      <seqElements>
        <seqElement id="Ad_S10_sqEl0" orderNo="0">
          <name />
          <adsBefore checked="false" />
          <footage ref="Siemens_S10" />
          <in>0</in>
          <duration>30000</duration>
          <price>0</price>
          <ratings />
        </seqElement>
      </seqElements>
    </section>
  </sections>
</advertisement>
```

```

</section>
</sections>
<adTypes>
  <adType ref="ad_electronics" />
  <adType ref="ad_cellular" />
  <adType ref="ad_youngster" />
</adTypes>
</advertisement>

```

## 5.5 Other Management Functions

The current version of the management tool offers more management functions than the ones described so far:

- ❑ The **Rating Panel** provides functions that enable the content manager to define and manage content description (rating) schemes and descriptors.
- ❑ The **Price Panel** provides functions that enable the content manager to price sequence elements, video productions, and services – depending on the price model used.
- ❑ The **Service Panel** provides functions that enable a service manager to configure a new service or content channel.
- ❑ The **Users Panel** provides functions that enable a service manager to create users with management authorization.

It is beyond the scope of this paper to describe the details of these panels and the applet-server communication.

## 6. Conclusion and Further Work

### 6.1 Conclusion

This work contributes to the web application user interface by demonstrating how the design of the user interface for an interactive video application can be completely separated from content creation. This allows a content provider to “brand” all their interactive video content in one common way. It also allows the content manager to focus on the task of deciding the best way to create and profile the content without worrying about too much about the application interface design. This separation of user interface design from application logic was accomplished by providing a JSP based user interface and by use of a SMIL template file.

The proposed application framework shows that it is possible to implement personalized, interactive video systems that are easy to use for users who want to add interactivity to their video content but who do not have the technologies or the training to use professional authoring tools. The major contributions made regarding interactive video composition, personalization, and management can be summarized as:

- ❑ The application framework supports personalization of individual sequences and hyperlinks within an interactive video presentation; this compared to previous work, such as the work reported by Merialdo et al. [16], that provides personalization of complete sections of video only.
- ❑ The application framework supports micropayment (per-video-clip payment) of interactive video.
- ❑ The application framework contains a set of management tools that combines content creation with content profiling.
- ❑ The management tools support the reuse of video content by allowing the same footage to be part of multiple video productions.

- ❑ The management tools are completely web-based – i.e., they do not require any application specific software or hardware to be installed. The tools only require a web browser and a video streaming client with SMIL capabilities.
- ❑ The management tools allow the content manager to focus on content preparation rather than advanced HTML and SMIL programming. The application framework takes care of the proper rendering.
- ❑ The application framework provides a sophisticated scheme for inserting video commercials in an interactive production based on the end users interest and profile, the appropriateness of the given video commercial within the context of the video, and the predefinition of potential target points within the interactive video where the video commercial could be inserted.

### 6.2 Further Work

The current system is based on SMIL 1.0. There are several other formats that could be used to provide interactive video, including SMIL 2.0 and Microsoft ASX. We will be looking into how to best structure the *Script Generator* to potentially support several script file formats.

The current system is designed to run on ordinary computers. We will be looking into the implications on the system to support other devices, such as TV’s, PDA’s, etc.

The current system provides the basis for resolving footage objects (URNs) into source files. We will be investigating more sophisticated methods that may take content, network, and user characteristics into consideration when resolving footage URNs.

The current system provides the basis for resolving footage objects (URNs) into source files. We will be investigating more sophisticated methods that may take content, network, and user characteristics into consideration when resolving footage URNs.

Information is exchanged between the applet and the servlet in the form of serialized Java objects. Hence, the client has to be a Java applet or application. An XML Schema [27] based implementation of the protocol will open the management interface for a larger variety of clients.

More research is required in the area of video personalization, for instance, to determine how the content manager can make sure that the video appears in a meaningful form after being personalized.

Finally, we will be looking at different ways to package the management tool. Currently, all panels are included in the applet sent to the user. There are security as well as performance reasons for not downloading all panels to all users. A targeted packaging may also make the user interface user friendlier because the user would only be seeing the panels he or she really needs to get the work done.

## 7. ACKNOWLEDGMENTS

We would like to thank the Deutsche Bank for being able to having access to multimedia content for our experiments.

We would also thank Sanjeev Segan, Amogh Chitnis, and Xavier Léauté for their help in implementing and refining the current version of the software.

## 8. REFERENCES

- [1] *RealSystem G2 Overview*, Real Networks, Inc., [http://www.realnworks.com/devzone/documentation/wp\\_g2\\_overview.html](http://www.realnworks.com/devzone/documentation/wp_g2_overview.html)
- [2] *Windows Media Services 7*, Microsoft Corporation, <http://www.microsoft.com/windows/windowsmedia/en/technologies/services.asp>
- [3] *Keep Our Streams Free. QuickTime Streaming Server - No Server Tax*, Apple Computer, Inc., <http://www.apple.com/quicktime/products/qtss/>
- [4] *Media Server*, Streaming21 Inc., [http://www.streaming21.com/products/prod\\_html/prod\\_01.html](http://www.streaming21.com/products/prod_html/prod_01.html)
- [5] *Mobility Meets Multimedia*, Packet Video Corporation, [http://www.packetvideo.com/wireless\\_platform.html](http://www.packetvideo.com/wireless_platform.html)
- [6] *Television and the Web Activity Page*, the World-Wide Web Consortium, <http://www.w3.org/TV/>
- [7] Zigmond, D. and Vickers, M., *Uniform Resource Identifiers for Television Broadcasts*, Informational RFC 2838, <ftp://ftp.isi.edu/in-notes/rfc2838.txt>, May 2000.
- [8] Sawhney, N., Balcom, D., and Smith, I., "Authoring and Navigating Video in Space and Time," In *IEEE Multimedia*, Vol. 4, No. 4, Oct. 1997.
- [9] K.G. Kumar, et al., *The HotMedia architecture: Progressive & interactive rich media for the Internet*, <http://www.developer.ibm.com/library/articles/hotmedia.html>
- [10] W-H. Ma, Y-J. Lee, and Du, D.H.C., "Video-Based Hypermedia for Education-on-Demand." In *IEEE Multimedia*, Vol 5., No. 1, 1998.
- [11] Boll, S., Klas, W., and Westermann, U., "Multimedia Document Models: Sealed Fate or Setting Out for New Shores," In *Multimedia Tools and Applications*, Vol. 11, No. 3, Aug. 2000.
- [12] Tsirikos, D., et al., "Interactive Multimedia Documents: A Modeling, Authoring and Rendering Approach," In *Multimedia Tools and Applications*, Vol. 12, No. 2/3, Nov. 2000
- [13] *Synchronized Multimedia*, the World-Wide Web Consortium, <http://www.w3.org/AudioVideo/>
- [14] Not, E.. and Zancanaro, M., "Content Adaption for Audio-based Hypertexts in Physical Environments," In *Proceedings from the 2nd Workshop on Adaptive Hypertext and Hypermedia, HYPERTEXT'98*, Pittsburgh, Pennsylvania, June, 1998.
- [15] Papadimitriou, C.H., Ramanathan, S., and Venkat Rangan, P., "Information Caching for Delivery of Personalized Video Programs on Home Entertainment Channels". The *International Conference on Multimedia Computing and Systems*, May, 1994.
- [16] Merialdo, B., et al., "Automatic Construction of Personalized TV News Programs". In *Proceedings from ACM Multimedia '99*, Orlando, Florida, Oct., 1999.
- [17] Bulterman, D.C.A., et al., "GRiNS: A GRaphical INterface for creating and playing SMIL documents". In *Proceedings from the 7th International World Wide Web Conference*, Brisbane, Australia, Apr. 1998.
- [18] *Windows Media On-Demand Producer: Overview*, Microsoft Corporation, <http://msdn.microsoft.com/workshop/c-frame.htm?/workshop/imedia/windowsmedia/contcreation.asp>
- [19] Tolva, J., *MediaLoom: An Interactive Authoring Tool for Hypervideo*. Project report, <http://www.mindspring.com/~jntolva/medialoom/paper.html>
- [20] Soares, L.F.G. et al., "Modeling, Authoring and Formatting Hypermedia Documents in the HyperProp System," In *Multimedia Systems*, Vol. 8, No. 2, Mar. 2000.
- [21] Auffret, G., et al., "Audiovisual-based Hypermedia Authoring: using Structured Representations for Efficient Access to AV Documents". In *Proceedings from Hypertext'99*, Darmstadt, Germany, Feb., 1999.
- [22] *Micropayments Overview*, the World-Wide Web Consortium, <http://www.w3.org/ECommerce/Micropayments/>
- [23] *Java(TM) 2 Enterprise Edition Developer's Guide*, Sun Microsystems, Inc., <http://java.sun.com/j2ee/j2sds/techdocs/guides/ejb/html/DevGuideTOC.html>
- [24] *Electronic Payment Transactions*, Brokat AG, <http://www.brokat.com/int/products/overview/x-pay.html>
- [25] *SET Specifications*, SET Secure Electronic Transaction LLC, [http://www.setco.org/set\\_specifications.html](http://www.setco.org/set_specifications.html)
- [26] *Embedded RealPlayer® Extended Functionality Guide*, Real Networks, Inc., Dec. 21, 1998. <http://service.real.com/help/library/guides/extend/embed.htm>
- [27] *XML Schema*, the World-Wide Web Consortium, <http://www.w3.org/XML/Schema>.