A Case Study of Workflow and Version Control for Regeneration of Multimedia Systems

Jon Patrick, Edmund Balnaves, Terence Kam, Daren Ler, Timothy Yeates
Basser Department of Computer Science, University of Sydney

Abstract. To create efficiencies in the production process of revisable multimedia systems it is necessary to define processes for the control of content revision and regeneration with a workflow control of these processes. A model for managing Multimedia Run-time Systems (MRS) is presented as consisting of a *revision control strategy* for managing *primary resources*. R*egeneration processes* that move data from one process to the next incorporating *derivative resources* on the way, and ultimately producing *run-time resources*, and a *workflow control process* to regulate and maintain the integrity of the regeneration process. A case study of one approach to tackling these problems is presented. This MRS, known as the English to Basque Learning Environment (EBLE), is a reference library of three books and concomitant sound files for second language learning of Basque.

Literature Review

The technologies involved in the creation of multimedia content are still in evolution. One-off generation of systems is well-developed as a process and often follows the model of making a film. Many modern applications are more complicated than film productions processes in that they must be regenerated time and time again, as a repeated production process, often by passing the materials through computational processes, and as well, they must be supplied in forms suitable for a variety of delivery media, such as books, web page or CD-ROM run-time systems.

Both at the national and international level, multimedia content creation is considered of immense strategic importance [1, 2]. The EEC Multimedia Educational Software Observatory (MESO) final report sees a “missing paradigm” for multimedia content production: it belongs neither in the processes for book publishing, film production, software development, nor game development [3]. Publishers working in the Educational environment have made a strong contribution to the production of multimedia content; they have also been among the earliest to seek a means to manage content *reuse*, *revision* and *regeneration* [4,5]. In this paper we use this terminology to express three different concepts. *Reuse* is the use of a given set of material in a number of different publications, whether they be in a book, on the WWW (as HTML files) or in a CD-ROM run-time systems. *Revision* is the process of changing content in a given system for the purpose of publishing a new edition of the same core materials. *Regeneration* is the processes both human and computational that need to be followed to create a revised version of a publication, and in our case a Multimedia Run-Time System (MRS). Workflow control is the process of managing regeneration so as to make sure it is as automatic as possible and thereby minimize the human intervention.

The importance of effective content re-use is highlighted by the rapid obsolescence of both technology and knowledge assets, with a longevity as little as 18 months [6]. Such a rate of obsoles-

cence demands the building of content re-use and concomitant regeneration technology systems that allow not only the repurposing of content into new products, but also the effective management of resources in a broader digital asset architecture.

Indeed, the regeneration of multimedia content presents a challenge in every stage of the publication process. The once-off publication of multimedia content might, of its nature, not require an inordinate investment in the long-term management of the content itself. However, the rapid obsolescence of information, both in its design or presentation elements and the very content itself makes the regeneration of this content with revised or updated material an early exigency, particularly in the management of Website Content [7] and in the Education arena [8]. The logical conclusion is that the content creator should in the first instance collect the component elements of a multimedia publication into a database comprising the design elements, logical structure and multimedia items themselves (text, audio, visual, etc) hence multimedia capable databases are gaining some prevalence in the industry.

However, such a content database presents not only a storage conundrum, but also a retrieval dilemma. The Information Retrieval of multimedia content has been most extensively explored in the Library Science literature. Early research and trials by Lancaster [9] led to the first large bibliographic databases such as MEDLARS in the early 1970's. The 1990's saw a movement from a communication model of Information Retrieval to a behavioural model of information seeking [10], with detailed analysis of this process by McCreadie and Rice [11, 12] and more recently applied in the multimedia context [13]. However, the analysis of Internet-based search behaviour has tended to remain isolated from the discipline of Information Retrieval Behaviour, with recent papers such as Wang, Hawk et al [14] focussing on a communication model of information search behaviour on the Internet. Research constructs described by Mann [15] in 1993 in his “methods of searching” model have been extended to the analysis of discipline-specific “material mastery” in the behavioural elements of digital library usage [16]. However, the complex interaction involved in Information Retrieval is difficult to translate to an effective user interface, and it has been argued that such models for IR are more useful as a reference guide during the prototyping design than as a concrete instrument for conceptual design [5, 17]. Nevertheless, over large multimedia content databases, the challenge of effective indexing of content is still being explored [18, 19]. Content indexing of video and sound archives present particular problems [20]. The behavioural elements in the process of information retrieval present a basic challenge not only in information retrieval, but also in the maximisation of the re-use of content.

The intersection of the problems of information retrieval with the management of libraries of digital assets were highlighted by Wood et al [21]. The common use of one-shot queries in web-sites and similar systems depends greatly on the specificity and relevance of metadata associated with the content. The structured collection of metadata in itself can be a time-consuming task, and is itself very subjective. Santini and Jain[22] explore the use of interface design to maximise the contextual information implicit in the database to enhance the process of retrieval.

The multimedia publisher commonly employs a variety of different tools for content creation [4, 23, 24]. The process of creation, therefore, can involve the complex interaction of different content creation systems. Work by Cheung & Chanson [8] shows the added complexity of reaching and identifying multimedia resources over a heterogeneous network. They propose three different

models: configuration, user control and presentation. Complexities of the software management of multimedia content are explored by Vazirgiannis et al [25] and Agoulmine et al [26]. Multiple methods of publication extend the concern of publishers into the area of product support as well as the production and distribution of their content [7].

The production process itself is also compounded by the variety of tools and processes that go toward multimedia content generation[27, 28]. The Microsoft "White Paper" on "Content Management" explores a largely linear view of the content management process [7], with little exploration of the issues of content re-use. It does, however, examine the interrelationship between content management and workflow of the publication process. Indeed, workflow systems themselves are taking on conceptual elements of document management systems (and visa-versa) [29]. The requirements of content reuse within multimedia content systems are explored by Liu and Hsu [30] from the perspective of document management systems, whilst the high cost of multimedia content publication [31] makes content reuse important. Nevertheless, the integration of workflow systems will need to go beyond the domain of the publication generation and into the arena of rights administration. Similarly, the complexity of the heterogeneous tool-set involved in the content creation makes the computer automation of processes involved in the generation of content all the more important. While the concepts underlying workflow management of business flows are well defined, they tend to be poorly defined in the editorial management of content databases, although it is clear that many concepts relating to workflow management apply; in particular:

* availability of graphical tools
* routing capability
* queue management of tasks
* management of events
* task and process management [32].

Issues of workflow control also intersect aspects of "trouble management" described by Agoumine et al [26] in the management of Quality of Service through the use of Quality of Service Assurance Systems. An area of vital interest is the integration of problem management and QOS management processes within the content management process itself.

The "Content Management System" oriented to the generation of material specifically for website publication has its roots in management of complex document content [36]. Current models for content management (such as those offered by Microsoft [7]) focus on the primary generation and maintenance of website content, and their origins lie in simple content delivery mechanisms for web publishing [24, 37]. Content separation from design is achieved either through Template-based approaches to content publication, or more recently through use of XSL (Extensible Style Sheet) templates interacting with XML document formats[38]. This area of industry discussion is very topical, and is best followed through current lists such as cms-list [39]. The explosive growth of content management systems highlights rather than diminishes the potential for some theoretical convergence in the area of content re-use, as website publishers struggle with issues of currency, navigation and editorial consistency on their sites.

The Library of Congress *National Digital Library* program illustrates the degree of intersection between digital asset management and web-based content management [40]. Their efforts to con-

struct a web-based paradigm for information management and retrieval provide good indicators for directions that Content Management Systems themselves are likely to pursue. Finally, obviously, the issue of content reuse is active in the education community, with efforts to build systems that maximise the reuse of course material in CDROM and tutorial courseware preparation [5, 42].

Model of a Workflow for Multimedia System Regeneration

The case study that is discussed here is representative of the problem of content management for a single system rather than the reuse of knowledge for multiple systems. Hence the important processes are defined by the requirements of "revision control" and "regeneration" both in terms of the various processes involved and the coherence of the workflow necessary for ensuring the final system is assembled in the correct way.

Reference books, instructional manuals and texts of the same type will invariably be used with secondary texts to serve as an enhancement of the material within the first source. In our case study we focus on the problem of regenerating a multimedia system which incorporates an instructional text for a reader in some field, and their need to refer to other reference books and multimedia resources that enhance the detail of the topic under study.

Satisfying this requirement has led us to develop a model of process control and management that contains three sub-systems each of significant size in their own right. The first is a run-time system for users, which integrates various text and sound resources (or any other media pertaining to that text) in a seamless way. The second is a revision control system, which allows modifications in the documents or manuals to be efficiently managed while satisfying the needs of multiple modes of publication. The third is a workflow control system, which will automate and control as far as possible the process of regeneration of the run-time resources after the primary resources have been updated.

Revision Control

Revision Control deals with the management of modifications of the source documents so that the needs of multiple modes of publication, such as a book, the web or CD-ROM are met. Different document types (e.g. primary resources that are linked to media resources such as text, sound, figure and video) may need different kinds of revision control techniques. Our model of revision control presents three types of resources that requires management and are subject to computing processes. The first is a *primary resource* or *document* considered to be created autonomously of other resources, and its existence or meaning is not dependent on other resources in the system. The second is a *derivative resource* is one which is derived, typically but not necessarily by computation, from the content of a primary resource. The computation may exploit other knowledge that is external to the primary resources. The third is a *runtime resource* or *version* which is a resource in the format it requires to participate in the runtime system. It may be changed from that format during utilisation at runtime, for example database records changed for display purposes. Examples from our case study are:
Primary resource - the three reference books,
Derivative resource - the parts of speech of each word in each example sentence necessary for retrieving the correct meaning of a word from the dictionaries,
Runtime resource - the XML versions of the primary resources.

An *entry* whether it be in a primary, derivative or runtime resource is a single element, such as a word or sentence, or their matching media files, that is the subject of some computation.

A Case Study - The English to Basque Learning Environment (EBLE)

The implemented system is called the English to Basque Language Environment (EBLE) and consists of a workbench of reference materials for second language learning, in this case for learning Basque as an English speaker. It consists of three books, a reference grammar, and Basque-English and English-Basque dictionaries. The reference grammar is a substantial book describing the grammar of the basque language and contains some 4500 examples of Basque sentences and their English translations (about 9000 sample sentences in total). The dictionaries are standard multi-lingual dictionaries with about 25,000 headwords each. The aim of the project was to build a system that linked each word in the Basque & English examples to their respective entries in the dictionaries. This would enable a user to immediately access the dictionary entry for any example word. Furthermore, it was decided to prepare voice recordings of all the examples and make them accessible to the user at run-time in the same way as the dictionaries are directly accessed. These concepts produce a seemingly remarkably simple run-time system, that is there are 3 books, each can be read at any time, a word selected in an example sentence will trigger the opening of the corresponding dictionary entry and the sound track for an example sentence can be played at any time. However the simplicity of the user environment belies the complexity of the system regeneration processes and the workflow control to manage them. (Fig 1. Screen shot of user system - to be inserted here)

Structural Issues in System Regeneration for EBLE

The system regeneration process consists of the conversion of 3 books into formats for appropriate run-time presentation. The books are stored in desktop publishing software formats. A major processing requirement is the necessity to extract the sample sentences from the grammar book to regenerate:

(i) Sound recordings of the sentences, or any other desired media material.

(ii) For each word in a sample sentence; a. its lemma form so that the correct headword form can be identified in the dictionaries, b, the part of speech so that the correct meaning of the headwords can be found in the dictionary entry. These processes have to be performed independently and off-line for the sentences in each language, Basque and English.

The dictionaries also require specialist processing. such as:

(i) Extracting headwords and their parts of speech and inserting cross-reference tags from the grammar book. Other content analysis is desirable to identify other fields of interest but it is not possible to achieve a general solution to that problem at the moment.

(ii) The dictionaries are owned and maintained by another party so there can be only be limited presumptions about its specific structure. Also the dictionaries are such detailed documents that there are many idiosyncratic features so that one cannot reliably predict the organisational structure of at least 20% of the headword entries.

The structural issues that emerge from the maintenance of the primary resources is:

(i) Delivery of the primary resources for both book publication and run-time system storage and retrieval are required and necessitate they be maintained in desktop publishing

environments for book publishing needs and then generation into run-time resources by some means.

(ii) Maintenance of the primary resources by their owners by mechanisms and procedures outside control of the Workflow regeneration system.

(iii) Changes to specific types of content in the primary resources that have flow-on consequences for the run-time system, such as, changes to the sentence examples that require changes to media files, part of speech tags, lemma and dictionary cross-references.

Hence the problem space consists of needing to use heterogeneous tools to generate the system and providing for the automatic flow of data from one tool to the next up to the point of system creation. Each process of exploiting a tool to serve an input to output transformation presents its own issues about errors in data, data storage formats for primary resources, derivable resources and their run-time equivalents, and data extraction processes for temporary materials awaiting further processing in the workflow chain.

Data Storage Issues for the Primary Resources

A generic view of revision control creates an initial focus on the storage mechanisms of the primary data sources. In this case study these consist of the two dictionaries and the grammar book. A serious question about the organisation of the revision control system is the form of data storage to be used, principally a choice between a sophisticated DBMS or a directory structure containing files. The choice of a solution at this level has a serious impact on many practical maintenance issues, particularly the human processes developed to interact with the workflow system. A DBMS solution would normally produce an easier pathway to computer controlled workflow and revision control but limit the flexibility of the human agents in the processes. A file system strategy as is common in the publishing industry will increase human flexibility in the revision process but make exact replication of processes more uncertain and usually lead to unnecessary work duplication and a fragile regeneration process.

In this case study the grammar book poses the greatest challenge, as the data within it (apart from the table of contents and the index) cannot be sorted into some generic form or structure. A major component of the system specification is that the example sentences have to be identified so links to other both primary and derivative resources are automatically generated. While there are many possible strategies we enforced a requirement upon the source data so that such ‘tagging’ already existed when it initially entered into the workflow system. This creates a requirement on the author to produce the document with this ‘tag’ formatting completed as a manual part of the workflow process, (hence the first stage of the process is the use of the word processor itself). This strategy means the ‘tags’ would exist in the primary resource before it entered the formal workflow process. Hence, the example sentences were given a special format structure in the primary document which enabled them to be extracted automatically at an early point in the workflow by an automatic process. Although we have the cost of creating the tool to extract these specially tagged entries it has a high reuse value.

In the next stage of the workflow control the extracted sample sentences are delivered to three other processes. Firstly, they are placed in a repository to be used for cataloging sound and media recordings, and then the Basque and English versions are sent respectively to separate off-line processors that extract the lemma and part of speech for each word in each sentence. The subse-

quently generated data is returned back to the Workflow control process where linking programs operate to insert the results as run-time resources.

The above strategy was considered to be more useful than an alternative where the whole document has links created wherever the relevant material occurs (e.g. if a link were to be constructed for a certain word, then any occurrence of that word would cause a link to be formed). This method makes the linking system more generic (linking is not restricted to only certain regions) but will then also clutter the run-time system with many unwanted links.

The two other primary resources are the multi-lingual dictionaries. Data storage issues for them where more difficult to determine as there are two options available. Firstly the dictionaries can be kept in their original format and linking information inserted in the grammar book. Alternatively, all the information in the entries could be extracted and the data housed within a database.

The first option is more restrictive than the second in the sense that the data is still sequential and manipulating it would be less efficient and more difficult. Alternatively, if we decided to have a search engine which could look up any category of data with any given entry from the primary resources, the database would obviously be more efficient. The links with a database would also be more efficient than that within a sequential document as finding a single link in a database is trivial as opposed to finding it in a sequential document. The use of the database would however incur the added cost of processing the data so that it is loadable into the database.

With a single file of an amended version of the primary source documents which includes all the required linking information, the output can simply be that file. It may however be more productive to split the output file into several smaller ones to ease the scale of processing. This in turn leads to more output issues as we would like to maintain the attributes of a conventional book (scrolling or flipping throughout the book seamlessly). Likewise, with having the dictionary stored in a database, the method of having several smaller files may be adopted, however the cost of processing them may be higher.

Yet another alternative is to auto-generate the output as required. By this we mean having a virtual semblance of having the output ready for the user. Should the user then select any entry, the runtime system would display the relevant portion around that entry with no boundaries or limits attached to this so that the user may flip through the whole document. Although the processing cost of implementing this at run-time is high, several methods such as caching (as we are used to with web browsers) can be employed to curb the cost.

The practical implementation of EBLE came down to deciding that primary resources and the information resource are best stored in the XML format even though the HTML format may suffice for the purpose. The purpose of storing this information in the XML format is to allow easy navigation and viewing by users through a simple web-browser user interface, which is embedded into the MRS. Also, the XML format is preferred to HTML because additional information description tags can be stored in it, hence providing more efficiency and capabilities in the retrieval process of the MRS. Storing each book in one file rather than as separate chapters has the advantage of allowing the user to scroll the whole book, without the need to add mechanisms for

maintaining a multi-part book. The multimedia resources (for example, WAV sounds, GIF pictures or AVI movies) are each stored in their own native format as individual files.

In summary, retaining the retaining the run-time resources as replicas of the primary resources is cheaper with respect to processing and resource usage for both the run-time and revision control systems. On the other hand, with a database design a lot more functionality and ease in the workflow control aspects would be obtained. In our case study we chose to use the original source documents in XML format as the equivalent run-time resource. This is satisfactory for the grammar book but not for the dictionaries. The dictionaries turned out to be too large to retain in memory when the grammar book is open. Furthermore, only access to limited points in the dictionary is required. A database storage mechanism for the dictionaries would have been a better strategy.

Revision Control for the Regeneration of Derivative Resources

In the run-time system, each derived multimedia resource has to be linked to some run-time element. Hence, the revision control system must deal with co-ordinating changes to the primary resources and their dependent media resources, and their conversion to a *run-time repository* whether as databases or files. However, regardless of how the central repository is implemented, its main purpose is to provide facilities for the storage and retrieval of all the data required by the run-time system. It must be stressed here that the run-time repository does not contain the primary documents as they are the responsibility of the original authors or the external revision control system and maintained elsewhere.

When the primary resource is modified, it is inevitable that the entire runtime version will have to be regenerated as cross-index information needs to be regenerated and the entire primary document rescanned for the creation of run-time resources. Since, it is not possible to detect precisely which sections of the primary document have been modified, incremental updates to the runtime resource is not the easiest alternative. For the purpose of revision control, overwriting existing files in the run-time repository is immaterial.

One significant issue must be addressed when the primary resource that is linked to media resources is being reprocessed. When that primary document is modified and reprocessed, any system of cross-indexing generated for the runtime resource version (such as hyperlinks) will no longer be valid. However recreation of all derivative resources would create a nonsensically inefficient system for our 5,000 media files representing all the example sentence entries. Therefore, a strategy must be implemented to maintain or automatically re-link the media resources to the entries. We have devised a strategy we call *functional-mapping* to achieve this task. In the *functional-mapping* strategy, a function that produces a one-to-one mapping from the primary resource's entries to the named media resources is used, where the name is produced by some automatic process not open for human intervention. In other words, for every unique primary resource entry that requires a derivative resource, the function will map it to a unique derivative media resource name and subsequently physical file. In the Basque grammar book, every example sentence is linked to a WAV sound file. The function that associates the example sentences with a WAV sound file is as simple as taking the example sentence, converting all punctuation marks into underscores and then appending a ".wav" to the end. Therefore, the function will map the example sentence: "Are you well?" into a WAV sound file name: "Are_you_well_.wav". The function can be of any form and complexity, but its imperative requirement is to produce a one-to-one

mapping of names from the primary resource entries to the media resources. Now, with the function in place, whenever the primary document is reprocessed, the following algorithm can be performed to keep the primary resource entries co-ordinated with the named derivative media resources entries:

1. For each primary resource entry, apply the function to produce a list, *M*, of named media resources that should be in the central repository.
2. For each named media resource in the central repository, if it is not in the list *M*, it is 'orphaned' and should be purged from the central repository because it represents primary resource entry that has been deleted.
3. Those media resources that remain in the central repository must hold links to the primary resource entries.

This algorithm is intrinsically different to a hyperlink mechanism. Our principle is that the actual text in an entry becomes the name of the derivative media file, not an extra separately defined text string inserted into a tagged string of, say HTML, as is commonly done with hyperlinks. After creating this string for the derivative media file name the connectivity between the source text entry and the media file could use any method of implementation including hyperlinks but that is a separate run-time implementation decision independent of the total system regeneration issues.

After the algorithm is applied, only the newly inserted and modified primary resource entries will not have corresponding named media resources. These media resources will have to be created or extant media resources linked to the new entries. Hence, with the *functional-mapping* strategy, whenever the primary document is modified and re-processed, a clean-up and re-linking of the media resources in the central repository will be performed and as a result the integrity of a coincidence between primary resource entries and their named derivative media files will be maintained. After identification of the need for the derivative media files it is a matter of the constraint model of the workflow control system as to whether the workflow requires the user to create the appropriate files or not. In the case of compulsory files the workflow may not stop the regeneration administrator from immediately performing other tasks, even partially, but must by definition constrain the administrator at some point. On the other hand, non-compulsory files do not inhibit the administrator from generating a full run-time system albeit with missing run-time resources, despite the presence of the executable functionality for those resources.

In practice the creation and linking of physical media files with primary resource entries has been made straightforward by the workflow control methods we have put in place. The system administrator accesses a module which has the list of all example sentence entries. They select an example and push a button to open the sound recording module to record the sound track. On closing the recording software the sound file is saved to disk with the appropriate file name.

Operationally the whole workflow system is contained in a menu of 6 items (Fig 2). The flow of activity is not totally regulated as anyone of the first 5 steps can be executed multiple times in any sequence. However doing so does not ensure the full integrity of the system. Particularly, Step 4 requires the completion of the off-line processing of the lemma and parts of speech of the grammar book examples. Absence of the files from this processing will mean that the book examples cannot be linked to the dictionaries. Likewise failure to perform step 3 before step 5 means that any new primary resource examples will not be available for sound recording. However this lack

of restriction is not entirely unwanted. The system allows the freedom of working with parts of the system and testing them without the time cost and weight of preparing the full system. This has proven most useful in the commissioning stages of EBLE.

1. Process English-basque Dictionary
2. Process Basque_English Dictionary
3. Select Grammar Book Source Documents
4. Select Part_of_Speech Files
5. Link grammar Book to Media Files
6. Generate run-time files

Fig 2. Menu of Workflow Control System.

Components of the Multimedia Run-time System

This section will discuss the storage organization and the indexing and retrieval strategy of the MRS which consists of a programmed system that must deal with the various classes of input resources we have already described. The first component is the primary resources which through the workflow process need to be converted to a runtime form. The second component is the derivative resources which need to be computed in some instances (parts_of Speech of example words) or captured in other instances (sound files). The task of the workflow process is to ensure these diverse elements or linked together appropriately and to pass them into the run-time system so that they are assembled together properly, and executable according to the functional design of the run-time software. In some senses beyond the workflow process of system regeneration is the maintenance and revision process of the run-time software itself. This can be performed independently of the regeneration workflow however elements of its specification will impact on data formats of the various resources, for example whether the dictionaries are stored as database records or as PDF or HTML files.

Access Across Primary Resources

Access between primary resources, that is from the grammar book to the dictionaries, is achieved by setting up an intermediate linking mechanism. For each entry in the primary resource, that requires a cross-reference to a run-time resource external to its own document there needs to be an index to those resources so an XML anchor tag (<*A*>) with a unique label to ‘bookmark’ the location is inserted. With all the anchor tags’ labels in place for every appropriate entry in the primary resources an index file entry is created. Each entry in the index file corresponds to each entry in the primary resource. There are basically two components to the index file’s entry: the primary resource entry’s anchor tag’s label and its attributes. The index file can be implemented by sophisticated DBMS or a simple comma-separated-value text file. Whatever its implementation, its main purpose is to associate each resource entry to the external resource. The resource entry’s attribute in the index file will depend on the specification of the MRS, for example, in EBLE the attributes for grammar book entries that cross reference into the English and Basque dictionaries are the lemma and the part-of-speech. Such attributes enable the system to retrieve a unique dictionary entry based on its lemma and part-of-speech. Other kinds of primary resources may need different kinds of attributes. The purpose of the index file entry’s attribute(s) component is to create direct retrieval of an example word from the dictionary. The access mechanism will depend on the for-

mat of the resource being retrieved in identifying, querying and searching for an entry in the primary resource. Once the entry is found in the index file, its anchor tag's label can be retrieved and the MRS can display that entry by 'jumping' to that entry's location in the dictionary through the web-browser user interface.

Accessing Multimedia Files

Since each of the multimedia resource entries is stored individually as a file in its native format, using an XML hyper-link tag in the run-time version of the resource readily makes the connection from each entry to its multimedia equivalent. When the user selects the primary resource entry through the web-browser user interface, the MRS invokes the operating system to deliver (for example, display picture, movie or play the sound or music) the primary entry's associated multimedia resource. Alternatively, the MRS could handle the delivery of the multimedia resource instead of delegating that job to the operating system.

Our system is designed to deal with the case of a primary resource entry linked to multiple multimedia resource. This problem is dealt with by using *resource-indicator file* that contains reference to one or more multimedia resources. The system operates so that when the user selects a resource entry, the MRS will open the resource-indicator file and retrieve the appropriate multiple multimedia resources via the index in that file.

Conclusion

A workflow control system for the revision control and regeneration of multimedia systems has been conceptualized on two major axes. The first axis is that of resources, namely primary, derivative and run-time. The second axis is the processes of Revision Control and Regeneration under the rubric of a constraining Workflow Control System. Revision control concentrates on setting primary resources in such a fashion that they can be processed automatically by creating appropriate conversion programs to produce their run-time equivalents. Regeneration concentrates on the process of moving data between the heterogeneous tools required for the conversion. The workflow control ensures the conversion processes are followed in appropriate sequences, ensuring the identification or creation of derivative resources so that they feed into the regeneration process as it progresses. Issues on the maintenance and development of the user's functional features of the MRS software itself are not directly addressed.

References

1. OECD. Content As A New Growth Industry. *European Audiovisual Conference Birmingham, 6-8 April 1998.* http://europa.eu.int/eac/papers/oecd_en.html. 1998
2. Multimedia Educational Software Observatory. http://europa.eu.int/comm/education/meso/
3. Multimedia Educational Software Observatory. http://europa.eu.int/comm/education/meso/ conclusions.pdf.
4. Brett, P.A. & Nash, M. Multimedia language learning courseware: a design solution to the production of a series of CD-ROMs. *Computers & Education* Vol. 32, 1, January 1999, pp. 19-33.
5. Colbert, X. The application of process models of information seeking during conceptual design: the case of an intranet resource for the re-use of multimedia training material in the motor industry. *Proceedings of the conference on Designing Interactive systems: processes, practices, methods, and techniques,* 1997, pp. 73 - 81.

6. Multimedia Educational Software Observatory. http://europa.eu.int/comm/education/meso/forecast.pdf.
7. Reynolds, J and Kaur, A. Content Management. http://www.Microsoft.com.technet/ecommerce/contmgt.asp, Microsoft, 1999.
8. Cheung, S. C. and Chanson, S. T. A model-based authorware for the construction of distributed multimedia systems, *Information and Software Technology*, Vol. 41, 11-12, 15 September 1999, pp. 715-727.
9. Lancaster, F.W. Interaction between requesters and a large mechanized retrieval system.' *Information Storage and Retrieval*, 4(2) 1968, pp. 239-252.
10. Kuhlthau, C.C. *Seeking a Meaning: A process Approach to Library and Information Services.* Norwood, New Jersey, Ablex, 1993.
11. McCreadie, M. & Rice, R.E. Trends in analyzing access to information. Part II. Unique and integrating conceptualizations. I*nformation Processing & Management*, Vol. 35, 1, January 1999, pp 77-99.
12. Kuhlthau, C. C. Role of experience in the information search process of an early career information worker: Perceptions of uncertainty, complexity, construction, and sources, *Journal of the American Society for Information Science*, Vol 50, 5, 1999, pp. 399-412.
13. Chang, X. Multimedia access and retrieval (panel session): the state of the art and future directions. *Proceedings of the seventh ACM International conference on Multimedia*, October 30 - November 5, 1999, Orlando, FL USA, pp. 443 - 445.
14. Wang, P. Hawk, W. B. and Tenopir, C. Users' Interaction with World Wide Web resources: an exploratory study using a holistic approach, *Information Processing & Management*, Vol 36, 2, 1 March 2000, pp. 229-251.
15. Mann, Thomas. *Library Research Models: a guide to classification, cataloguing and computers.* NY, Oxford University Press, 1993.
16. Covi, L. M. Material mastery: situating digital library use in University research practices, *Information Processing & Management*, Vol 35, 3, May 1999, pp. 293-316.
17. Sonnenwald, D. H. and Pierce, L. G. Information behavior in dynamic group work contexts: Interwoven situational awareness, dense social networks and contested collaboration in command and control, *Information Processing & Management*, Vol 36, 3, 1 May 2000, pp. 461-479.
18. Amato, G., Rabitti. F. and Savino, P. Multimedia document search on the Web, Computer Networks and ISDN Systems, Vol 30, 1-7, April 1998, pp. 604-606.
19.
20. Dagtas, S. Models for motion-based video indexing and retrieval, *IEEE Transactions on Image Processing*, Vol 9, 1, January 2000, pp. 88-101.
21. Wood, M E J, Campbell, N.W. and Thomas, B.T. Iterative Refinement of Relevance Feedback in Content-based Digital Image Retrieval. *Proceedings, ACM Multimedia 98, Bristol, England, Sept. 12-16 1998*. ACM, NY, 1998, pp. 13-18.
22. Santini, S & Jain, R. Beyond Query by Example. *Proceedings, ACM Multimedia 98, Bristol, England, Sept. 12-16 1998*. ACM, NY, 1998, pp. 345-350.
23. Michalas, V. X. A comparison of multimedia application development platforms towards the object web, *Computer Standards & Interfaces*, Vol 22, 1, March 2000, pp. 13-26.
24. Fraternali, P. Tools and approaches for developing data-intensive Web applications: a survey; *ACM Comput. Surv.* 31, 3 (Sept. 1999), pp. 227-263.

25. Vazirgiannis, X. Specifying and authoring multimedia scenarios, *IEEE Multimedia*, Vol 6, 3, 1999, pp. 24-37.
26. Agoulmine, X. Trouble management for multimedia services in multi-provider environments, *Journal of Network and Systems Management*, Vol 8, 1, 2000, pp. 99-123.
27. Duffy, K., Content management comes of age: streamlining the information creation and delivery process, *Information Management & Technology*, Vol 33, 4, 2000, pp. 183-191.
28. Liu, P. and Hsu, L. H. Multimedia Document Systems From Authoring perspective. *Internet and Multimedia: Systems and Applications.* Furht, B (ed). IEEE Press, 1999, pp. 635-667.
29. Collins, P. Document and workflow management '99. *Management Services*. Vol 44, 1, 1999, pp. 23-26.
30. Hysell, D. Meeting the needs (and preferences) of a diverse World Wide Web audience; *Proceedings on the sixteenth annual International conference on Computer documentation*, ACM, 1998, pp. 164-172.
31. Kelly, B. Will it all fit together? The need for standards and the technical challenges. *Multimedia Information and Technology*, 2000, Vol. 26, No. 1, pp.
32. Rojas. M Perez, T. Evaluation of Workflow-type software products: A case study. *Information and Software Technology.* Vol 42, 7, May 1, 2000, pp. 489-503.
33. Crispin J. Digital chaos and professional standards. *Multimedia Information and Technology*, 2000, Vol. 26, No. 2, pp.
34. Koch, E. R, J and Zhao, J. Copyright Protection for Multimedia Data. *Digital Multimedia and Electronic Publishing, edited by Earnshaw, R, Vince, J and Jones, H.* NY, Academic Press, 1996, pp. 203-214.
35. - Cox, J. E. The Changing Economic Model of Scholarly Publishing: Uncertainty, *Complexity, and Multimedia Serials, Library Acquisitions: Practice & Theory*, Vol 22, 2, Summer 1998, pp. 161-166.
36. Celentano, A., Pozzi, S. and Toppeta, D. A multiple presentation document management system; *Proceedings of the 10th annual International conference on Systems documentation*, 1992, pp. 63-71.
37. Yeh, J-H; Chang, J-Y; Oyang, Y-J. Content and knowledge management in a digital library and museum, *Journal of the American Society for Information Science*, Volume 51, Issue 4, 2000, pp. 371-379.
38. W3C. Extensible Markup Language (XML) 1.0. http://www.w3.org/TR/REC-xml.
39. Content Management Systems. http://www.camworld.com/cms [online discussion list].
40. Marchionini, G., Plaisant, C. and Komlodi, A. Interfaces and tools for the library of congress national digital library program, *Information Processing & Management*, Vol 34, 5, September 1998, pp. 535-555.
41. Cline, L. Buying electronic: the development of the electronic book market in academic libraries, *Library Collections, Acquisitions, and Technical Services*, Vol 24, 2, Summer 2000, pp. 312-315.
42. Candler C.S.; Andrews M.D. Avoiding the great train wreck: Standardizing the architecture for online curricula, *Academic Medicine*, Vol 74, 10, 1999, pp. 1091-1095.