This paper appeared at Australasian Conference on Information Systems ACIS'99, Wellington, New Zealand.

Designing Multimedia Development Environments with Reuse in Mind

Jacob L. Cybulski
Tanya Linden

Department of Information Systems
University of Melbourne
Parkville, Vic 3052, Australia
E-mail: j.cybulski@dis.unimelb.edu.au
t.linden@dis.unimelb.edu.au

Abstract

This paper describes the `Analyse-Organise-Synthesise` *methodology of multimedia development. The proposed methodology prescribes a systematic approach to developing multimedia applications. It emphasises reuse of multimedia objects and at the same time addresses quality and productivity of the development process. We illustrate our approach with examples drawn from the use of the MATE system - Multimedia Assisted Teaching Environment - a software tool that was designed with reusability in mind. We believe that using the* `Analyse-Organise-Synthesise` *methodology will improve the productivity of multimedia developers and will enhance the quality of individual components and of the entire multimedia products.*

Keywords

Reuse, multimedia, repository

INTRODUCTION

Development of multimedia products is extremely labourious and costly. It often requires expensive software tools and professional services [Buford1994]. Multimedia design methods are still very immature and informal, hence they sometimes lead to incomplete, inconsistent, difficult to use and ill-documented products.

Multimedia design methods currently in use have been adopted from the fields of publishing, technical writing and art design. Traditionally these methods focus specifically on artwork but they largely ignore the development process. At the same time multimedia products are very complex. They have sophisticated structure and interdependencies that necessitate use of software tools and large repositories of components. Design methods adopted from publishing and art design may not be fully effective when applied to large multimedia projects where a more rigorous engineering approach is more appropriate. To deal with the complexity of multimedia products, developers need special design methods and tools to support them. Although design approaches and tools were developed and used with some success in other fields, e.g. software engineering (see Box 1), they are not always appropriate for dealing with multimedia artefacts. We need a more systematic approach to multimedia development, which results in high quality products and increased productivity.

A typical approach to the construction of multimedia products is to develop them entirely from scratch. Considering that multimedia developers usually have at their disposal large collections of ready-made components, such an approach is not optimal. We believe that pro-active reuse of multimedia can provide significant savings in time and production costs.

Effective reuse of multimedia will have to address issues related to artefacts' analysis, organisation and synthesis [Cybulski+1998; Cybulski1996b], which imply the aspects of artefacts' representation, classification, storage and sharing, search, retrieval and composition of new multimedia products - all of which relate to the issues of multimedia repository.

Every big organisation produces and archives large volumes of legacy documents. Over the years, organisations involved in training, consulting and publishing establish vast archives of multimedia documents and components. For example, educational institutions store collections of teaching and assessment material, which includes lecture presentations, printed notes, tutorial exercises, laboratory examples, demonstrations, case studies, project descriptions, assignments and exams. Such information comes in a variety of electronic media types, e.g. text, images, sound, videos and executable programs. Alternatively, some material may be available in hardcopy form only. Although such documents may not be suitable for the new projects, some of their components can be utilised and reused. When creating new products, multimedia developers need to effectively and efficiently find, combine and create multimedia components.

The majority of work in reuse to date has been done primarily in the area of software engineering. Initially the emphasis of reuse effort was on code reuse, both in the binary and source form. Recently, however, reuse found many new applications such as reuse of analysis and design work products [Biggerstaff+1987; Li1993; Lubars1988], reuse of educational material [Ip+1997; Roschelle+1997], multimedia components [Cybulski+1997] and processes [Rossi+1997].

Generally speaking, reuse is the use of previously acquired concepts or objects in a new situation. It involves matching of new and old situations, duplication of already developed objects and actions, and their adaptation to suit new requirements [Prieto-Diaz+1987].

A reusable artefact is a certified, fully developed, possibly generic component, available for the integration into a multimedia product under development. Certified means that a reusable component satisfies some quality and reusability criteria. Generic means that the reusable component is abstract enough to be adapted to particular situations, e.g. a template. [D'Alessandro+1993]

A controlled collection of reusable artefacts constitutes a reuse repository. Reuse repositories are expected to provide certain types of services to their users, e.g. storage, searching, inspecting and retrieval of artefacts from different application domains, loading, linking and invoking of stored artefacts, specifying artefacts relationships, etc. [Prieto-Diaz+1987].

Domain analysis is the process of identifying, organising and reusing knowledge about some class of problems -- the problem domain -- to support the description and solution of those problems [Prieto-Diaz+1991].

Box 1: Reuse Concepts and Terminology

Organisations producing multimedia products need to share and reuse multimedia components to reduce the product development costs, to increase developers' productivity and to enhance the quality of the resulting product. Designing multimedia components, or artefacts, with reuse in mind would enhance their sharing within the organisation. For ease of access and distribution these components may be stored in a special purpose multimedia repository.

MULTIMEDIA REPOSITORY

In today's multimedia systems, especially the commercial ones, sharing of artefacts is based on informal repository facilities, e.g. ClipArt databases and World Wide Web. Typical access to these collections is usually achieved with the use of simple methods, which include hyperlinks, bookmarks, and query languages. Existing search engines such as Alta Vista, Yahoo, Lycos and InfoSeek, though more complex, are heavily dependent on textual representation of searchable artefacts and are, hence, less applicable to multimedia objects. Even Yahoo Image Surfer requires entering textual descriptions and keywords for

classification of images. Visual information retrieval systems, e.g. QBIC[1] and Virage[2], analyse not only text but also visual features, such as shapes, colours and their position, to query image repositories. However, these systems are not widely used as yet. Virage also produced a sophisticated video search system VideoLogger that automatically extracts information about the video content in real time. It slices the video into segments based on changes in the visual content, such as scene cuts, camera pans and zooms. Then it extracts distinct keyframes to represent each segment, generating a digital storyboard and provides a frame accurate index that gives immediate, non-linear access to any segment of the video.

Research tool Himotoki [Hirata+1996] also offers facilities allowing processing of textual information and media-based attributes. The system projects media contents into media-specific attributes which may be difficult to express as meaningful keywords but which are nevertheless powerful clues for media specification. Then it uses these attributes to specify the semantics of the media representation and stores the attribute values as media-indexes. Subsequently user queries are matched against the attributes from the media indexes and the candidate artefacts are retrieved.

REUSE IN MULTIMEDIA DEVELOPMENT

In a typical approach to reuse, development takes place in two distinct phases: development-for-reuse and development-with-reuse. The focus of the first phase is on development of reusable components, their detailed description, classification and storage in the repository for future use. The second stage focuses on the construction of products with the aid of reusable components drawn from the repository. The two phases are hard to separate as developers discover new components in the process of product development.

We recommend an approach, which addresses issues of *reuse* and *repository* in multimedia development. It describes and prescribes a number of activities in the artefacts' life cycle as adopted from software development. Each activity proposes several tasks that may be undertaken in the processing of an artefact. These activities may be performed individually, in a sequence, or in parallel.

1. *Continually analyse the existing multimedia products:*
   - Identify potentially reusable components in existing multimedia and legacy systems.[3]
   - Describe and represent the features of artefacts identified as potentially reusable.
   - Generalise, if necessary, and document artefacts selected for reuse to widen the scope of their applicability to suit development of new multimedia products.
2. *Organise a repository of reusable artefacts:*
   - Classify and index artefacts to be stored in the repository. This can be based on facets, keywords, enumerated schemes or media-indexes (See Box 2).
   - Store reusable artefacts in a multimedia repository.
   - Search for the candidate artefacts suitable for the new project. The search techniques are dependent on the repository facilities and proper documentation of artefacts.
   - Retrieve the candidate artefacts for further processing determined by the search.

---

[1] Description and download http://wwwqbic.almaden.ibm.com/

[2] Demo available at http://www.virage.com/cgi-bin/query-e

[3] Although the components may have been built without any reuse consideration they can be altered and generalised to meet the requirements of future multimedia systems.

3. *Synthesise a new multimedia product:*
    - Select the necessary artefacts from the collection of candidate artefacts.
    - Adapt the selected artefacts, if necessary, to meet the requirements of the current project.
    - Create new artefacts as required, document them and add them to repository.
    - Integrate selected and newly created artefacts into a final product.

When to use this methodology:

✓ When creating a multimedia product, e.g. a Web site, multimedia presentations, multimedia materials on CD-ROM.

✓ When adapting existing multimedia documents and their components for reuse.

✓ When assembling and maintaining a large collection of multimedia artefacts.

✓ When searching for artefacts suitable for inclusion in the new multimedia products.

Pro's:

- Reusability will lead to multimedia repositories of high quality reusable multimedia artefacts that are easily adaptable, possibly generalised, and properly documented. These components can be shared between multiple products.
- The acquisition of artefacts is conducted on a continuing basis.
- Multimedia reuse increases developers' productivity in the long term. The richer the repository, the more likely it is that the developer will find most of the required artefacts in there.
- Reusability based multimedia development facilitates search of the repository and selection of matching artefacts.

Con's:

- Copyright issues should be considered before storing multimedia artefacts in the repository, modifying them and using them in products.
- Some developers have strong opinions against reusing components produced by others, e.g. they may feel that the quality of their own components is higher.
- Adoption of reuse practices may initially increase developers' workload.
- The multimedia repository may be costly to manage in the short term. As for any repository maintenance, setting it up and entering data may seem to be boring.

METHODOLOGY APPLICATION

In this section we describe the activities prescribed by the `Analyse Organise Synthesise` methodology as implemented by Multimedia Assisted Teaching Environment (MATE), the original tool built to support design and delivery of multimedia teaching material (Cf. Figure 1). We illustrate the workings of the MATE system with examples drawn from development and offering of courseware in object-oriented systems analysis and design.[4]

[4] Such a course is currently being developed with the use of MATE.

MATE Architecture

MATE has a number of sub-systems responsible for various services offered to different classes of its users, i.e. a librarian (repository services), an author of multimedia material (authoring services), a teacher (on-line broadcasting and delivery services) and a student (reception and projection services).

The MATE's authoring tool helps the author to compose various types of audio-visual components into slides consisting of text, images, animation, video, sound, applets and ActiveX components. Artefact attributes, such as colour, size, position or the speed of presentation, can be altered to suit the presentation. Artefacts selected for presentations may be organised into a presentation plan, which consists of topics, lectures and courses. The authoring tool can also aid the author in publishing teaching material on CD-ROM, to facilitate its distribution to course participants.

The presenter follows a predefined lecture plan to broadcast course material to on-line student participants. The bulk of presented material is shown to students from the CD-ROM. Any additional material and course updates are fetched from the web or the teacher's repository. Students attending on-line lectures (Cf. Figure 2) interact with the presentation, by solving small problems, doing drills and quizzes, or sending queries directly to the lecturer. In response to student's questions, the lecturers may alter the course of their lectures or they can answer student queries by incorporating the explanation into the body of the lecture

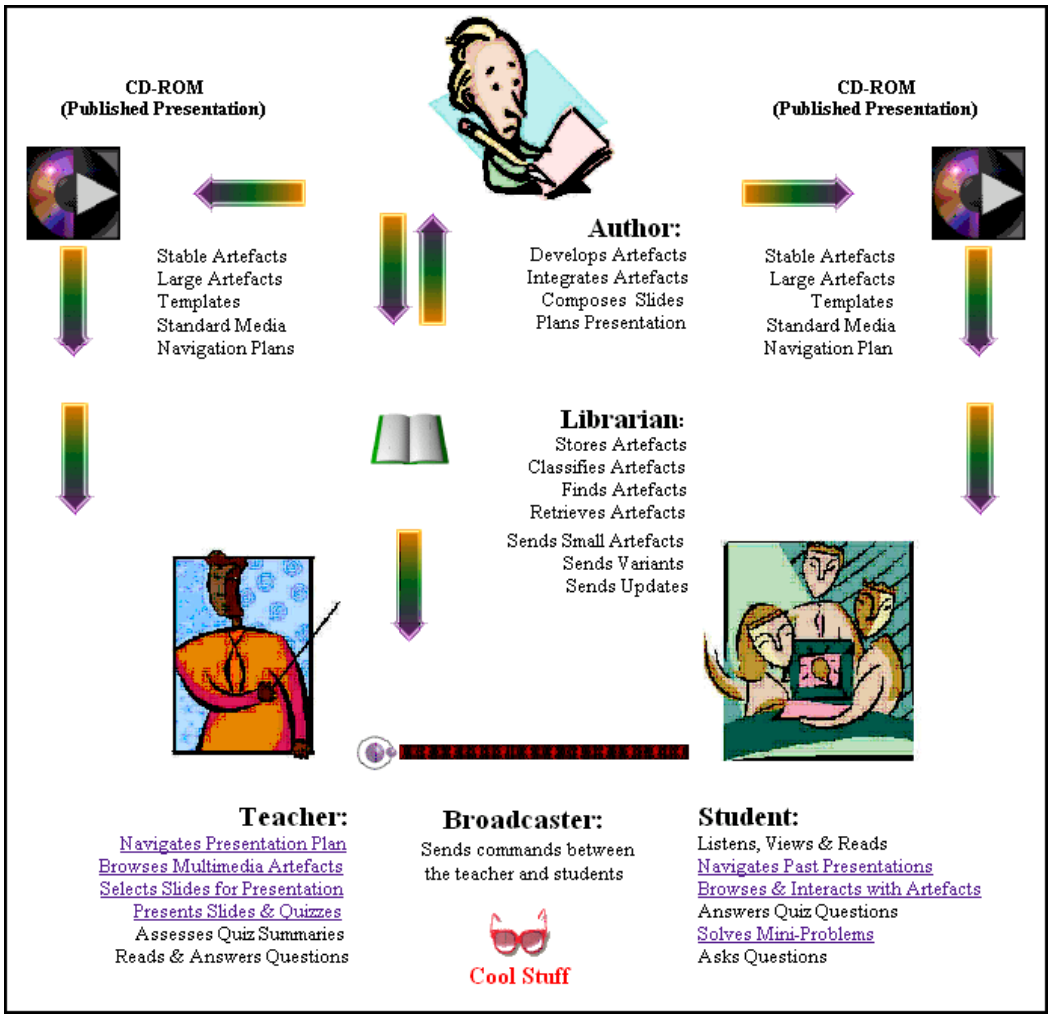

Figure 1: MATE Architecture

material.

All presented information is stored in the MATE repository. The repository is capable of supporting all the system's users, i.e. the librarian in artefact classification and storage, an author and a teacher in searching, inspecting and retrieval of artefacts, and specifying artefacts' relationships, and students in gaining access to on-line material.

In our organisation teachers are working under pressure to be more productive, so that they could effectively and efficiently deliver educational courses to constantly expanding community of students. In particular, IT-related subjects need to be continually updated and tailored to the specific needs of various student groups. Teachers of related subjects need to be systematic about their course structure, sharing of teaching material, and cross-referencing their educational wares across the curriculum. Productivity, quality and the systematic approach are the three essential components of our proposed methodology.

Domain Analysis

We adopted a development process, which starts with preliminary domain analysis. In this phase of our work, we identify the required and pre-existing artefacts in a given subject area. These artefacts come in a variety of forms, e.g. text files, hand-written concept documents, storyboards, images in electronic and paper form, videos and animations, sound files, etc. On occasions, we capture paper-based information into electronic documents by scanning their text and images. At this point of time, we also consider the requirements of a multimedia repository to be used in a given teaching domain. We decide on the size, access and the future operation of a database system to support a domain repository. For personal use we suggest the use of Microsoft Access or Jasmin, and for the enterprise use we consider Oracle or Microsoft SQL.

These preliminaries establish the necessary pre-condition for the effective reuse of multimedia components in later product development.

Artefacts Analysis

As many of the existing artefacts share some of their structural features and contents, the designers may opt to modify or generalise them before their final storage in a repository. The

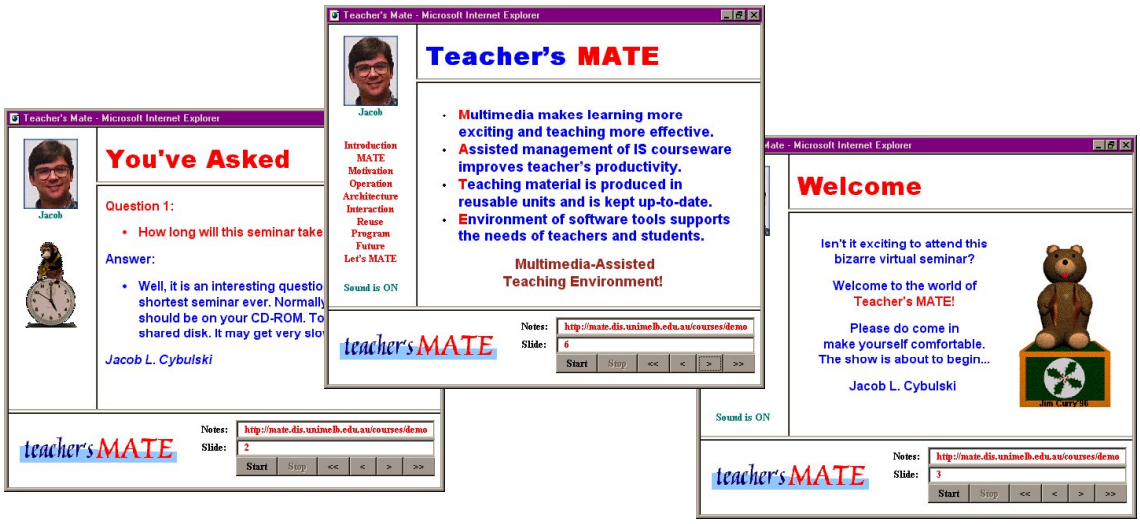

Figure 2: Presentation of lectures with MATE

simplest case of multimedia generalisation is to construct a partially filled template, which imposes a consistent layout and defines shared components to be repeated in all instances generated from the template (See sample lecture slides in Figure 2). Special-purpose applets capable of displaying systems analysis models (UML) for various case studies have also been developed to enhance artefact reusability by means of parameters-driven model building. It should also be mentioned that we consider the artefact building processes to be reusable as well, hence, a sequence of actions leading to the artefact construction, such as artefact composition, template filling or passing model parameters, may also be reapplied in the development of lecture slides, examples, quizzes, etc. Such reusable processes are treated as reusable artefact and stored in the MATE's repository.

Keyword based classification aims at identifying words and expressions that can be used as indicators of artefact content.

Enumerated classification structures information in a hierarchy of concepts based on concept types.

Faceted classification relies on several well-defined sets of attributes, called facets. Facets specify a controlled vocabulary used in the artefact classification. To assist match artefacts, each facet specifies a metric to determine closeness of classification terms, which are commonly broadened with the use of synonym lists and a thesaurus [Cybulski1996a].

Media-indexes allow users using media clues for retrieving media objects. [Hirata+1996]

Box 2: Classification terminology

Artefact Organisation

All of the collected artefacts need to be stored in a common multimedia repository. To facilitate efficient search and retrieval of reusable artefact from the repository, they need to be classified, indexed and structured. We considered several approaches to the effective classification, including keywords like in Microsoft Clipart, enumerated schemes and facets [Prieto-Diaz+1987; Sorumgard+1993] and attribute values (textual representation). In MATE

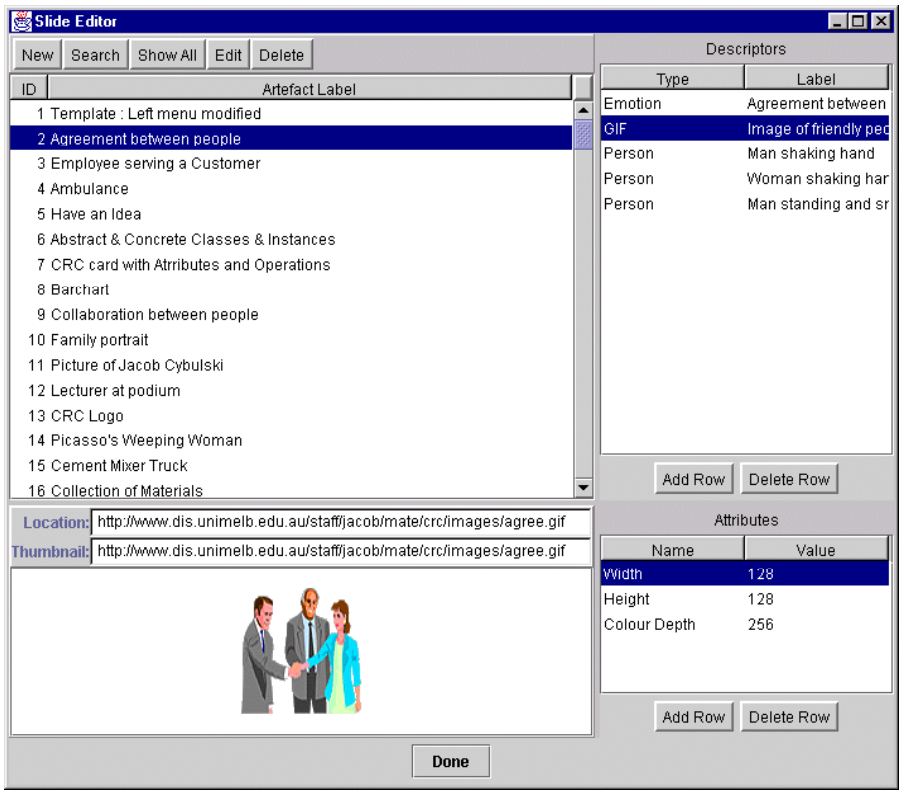

Figure 3: Artefact with associated descriptors

we classify artefacts by attribute values. Some of these attributes are common to all types of artefacts, e.g. their URL, name and thumbnail. Others are organised into attribute groups, which we call descriptor types. Subsequent search utilises this classification and thus search is based on textual representation of artefacts' contents.

Let us consider the classification of an image depicting a group of people striking a deal (Cf. Figure 3). This image has a number of descriptors attached to it based on its physical properties and its contents. In this example, the image has descriptors characterising its image format, people and the roles they play in a given situation or activity, and emotions associated with the image:

Descriptors		
Image (GIF)	**Person**	**Emotion**
♦ File type	♦ Gender	♦ Settings
♦ Height	♦ Person role	♦ Process
♦ Width	♦ Activity performed	♦ Outcome
♦ Colour Depth	♦ Special-features	♦ Effort

Search in the repository is implemented using queries. A window for query definitions is shown in Figure 4. Search allows joining query criteria with logical operations, such as AND, OR or NOT. All information is in text format because our classification relies only on textual attribute values. This can be improved in the future. For images, a visual information retrieval (VIR) system may be incorporated to use visual properties of images such as colour, texture and shape, as well as textual keywords. Some commercial VIR systems such as VIR image engine[5] by Virage and QBIC[6] by IBM may be added to the repository in the future. For video search Virage VideoLogger can also be incorporated.

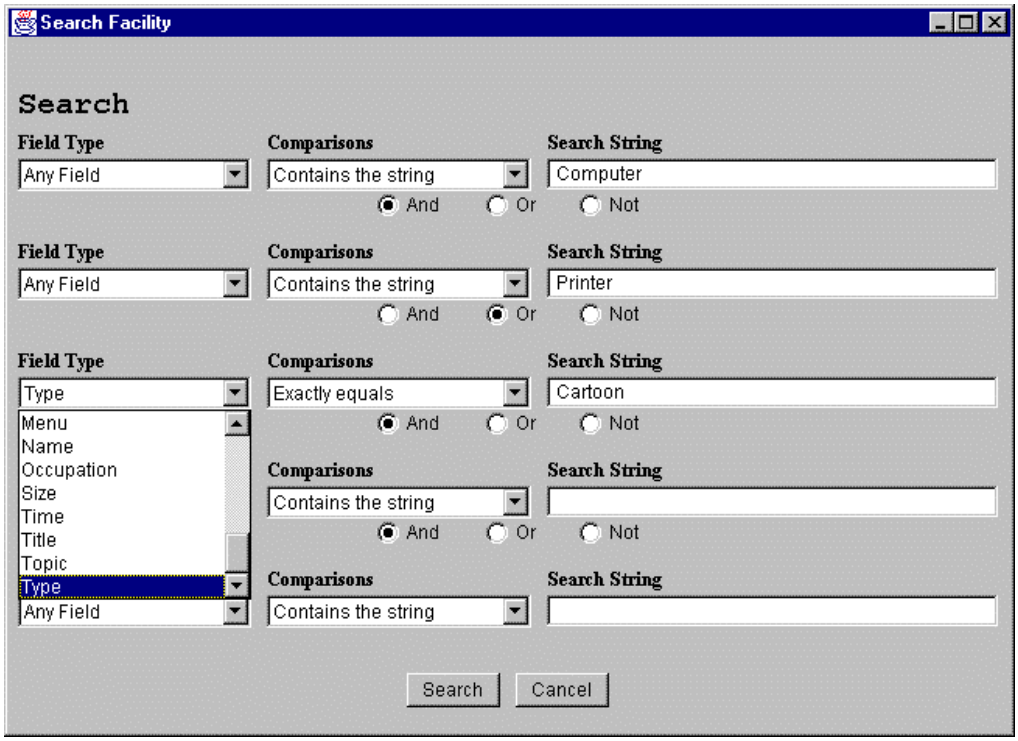

Figure 4: Specifying a query to search for artefacts

[5] Demo is available at http://www.virage.com/cgi-bin/query-e

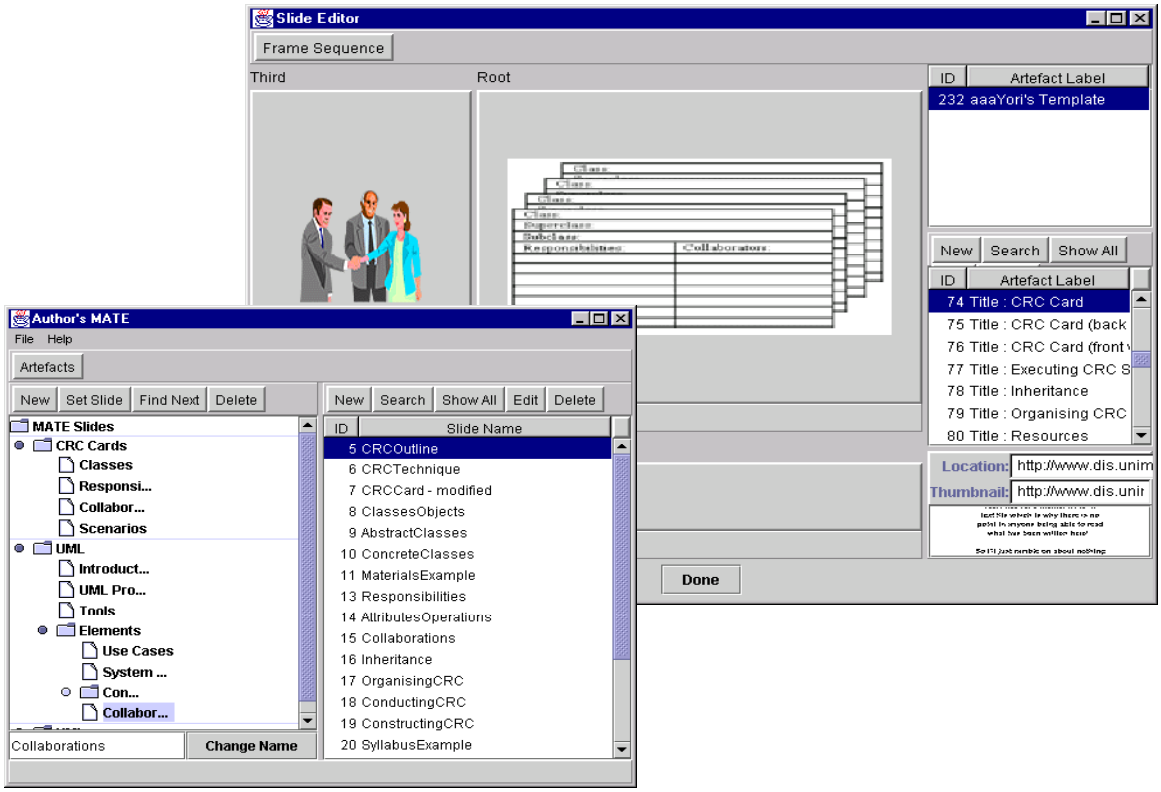

Figure 5: Composing artefacts into slides, lectures and courses with MATE

In MATE, the selection of artefacts from amongst the list of candidates identified by the search is based on artefacts' description, preview, classification and their contents representation. For example, while searching for a video with a requirements gathering interview, it is impractical to view all available videos. Instead, the video contents is indexed and subsequently searched based on the interview location, names of interviewer and interviewees, the purpose of the interview, the system under construction, etc.

Artefact Synthesis

The selection of artefacts suitable for new multimedia projects can occur either by visual inspection of artefacts, or their attribute values, or based on previous use of an artefact. Some artefacts will need further adaptation.

For example,

- Instances can be generated from the slides template described above (Cf. Figure 2 and Figure 5) by filling template gaps with appropriate multimedia artefacts. If a template is a frame set, then the process of adding a component to a frame will generate HTML automatically.
- To fit in the template, some images may need to be resized for better visual effect.
- Colours of existing artefacts may have to be changed to suit the selected colour scheme.
- Some sequences of images may be transformed into animations using packages such as Microsoft Gif Animator or Gif Construction Set or by applying Javascript.

---

[6] Download and description available at http://wwwqbic.almaden.ibm.com/

At the same time to suit the new project requirements, additional new artefacts may have to be created - preferably with reuse in mind. These artefacts also have to be properly documented, classified and added to the repository.

As it can be seen from our examples, MATE facilitates all three aspects of multimedia reuse, i.e. artefacts analysis, organisation and synthesis. As the result of applying the approach set by the `Analyse Organise Synthesise` methodology, the teachers in our organisation will be more effective and productive in courseware development. MATE will also helps us in sharing and reusing high quality multimedia components, and co-ordination of courses by cross-referencing their contents.

SUMMARY AND CONCLUSIONS

In this paper we proposed an approach to constructing and maintaining multimedia products and repositories. Our approach has been adopted from software development. The proposed approach emphasises reuse of multimedia artefacts and addresses the issues of quality and productivity of the development process. Some of the steps described, such as identification, representation, classification, search, selection and adaptation, can be automated and that's what our MATE project is aiming for.

To evaluate our approach to structuring the multimedia development process, we compare it with the approaches taken in other multimedia tools. In Table 1 we evaluate these approaches in terms of activities described in the `Analyse Organise Synthesise` methodology. Four possible approaches are considered.

- ***Find - Apply.*** This is the simplest of the four approaches, in which a multimedia designer browses through existing products in search of useful components. Found components are subsequently incorporated, in ad-hoc fashion, in any new products.
- ***Find - Store - Apply.*** In this approach, developers maintain collections of multimedia components over long periods of time. They add new components to the multimedia collection when they come across anything that may prove to be useful in future products. The drawback of this approach is that these collections are usually maintained in the unplanned fashion without proper indexing or documentation of artefacts. Frequently the potentially useful components are stored in a system of directories and subdirectories. This strategy fails to support flexible object composition and sharing due to the general limitations of a filing system [Shih+1997b]. The search in such collections can only be conducted by artefact name or the preview of its contents.
- ***Find - Organise - Apply.*** Developers may opt to create specialised multimedia repositories where artefacts are classified and indexed, then stored and retrieved as required. Although, in general, multimedia repository provides a benefit of readily accessible artefacts suitable for the project under development, in this approach, artefacts maintenance, their adaptation and adoption is still largely unexplored.
- ***Analyse - Organise - Synthesise.*** This approach is more systematic, i.e. a developer actively analyses existing products and their collections for the opportunity to identify and adopt reusable multimedia components. Then he or she documents, modifies and generalises, classifies and indexes such artefacts to construct high quality reuse repository. The method also provides some guidelines on the construction of new multimedia products of reusable components, by means of artefact composition, template instantiation, parameter passing, etc.

Approach	***"Find-Apply"***	***"Find-Store-Apply"***	***"Find-Organise-Apply"***	***"Analyse-Organise-Synthesise"***
Example	Any browser with HTML editor, e.g. Netscape, IE, Opera	Microcraft Author, Anecdote, Microsoft PowerPoint	ClipArt, Asymetrix ToolBook, Web Robots, Spiders, Wonderes and Seach Engines, e.g. Yahoo, Excite	Himotoki [Hirata+1996], IMMPS [Shih+1997a], MATE [Cybulski+1997], REBOOT[7], F3[8]
Identify	◗	◗	◗	●
Describe	❍	◗	◗	●
Generalise	❍	❍	◗	●
Classify and index	❍	❍	●	●
Store	❍	●	●	●
Search	●	●	●	●
Retrieve	●	●	●	●
Select	❍	◗	●	●
Adapt	❍	◗	◗	●
Create	◗	●	◗	●
Integrate	◗	◗	◗	●

Table 1: Approaches to Multimedia Reuse
Support - ❍ (nil or manual), ◗ (partial), ● (full)

As can be seen from the Table 1, the last approach is the most systematic and it promotes the most efficient organisation of artefacts. Our MATE tool fully supports this process. We use the tool (and hence the process) to develop multimedia courseware for a number of related subjects in our department. Although the use of MATE is currently limited to a few multimedia applications, we find it very helpful. Full-scale empirical studies of the tool and its associated methodology will be conducted at a later date.

BIBLIOGRAPHY

1. Biggerstaff, T. and C. Richter (1987): "Reusability Framework, Assessment and Directions." *IEEE Software*(July): p. 41-49.
2. Buford, J.F.K. (1994): "Multimedia Systems." : ACM Press.
3. Castano, S. and V. De Antonellis (1994): "The F3 Reuse Environment for Requirements Engineering." *ACM SIGSOFT Software Engineering Notes.* ***19***(3): p. 62-65.
4. Cybulski, J. and T. Linden (1998): "Composing Multimedia Artefacts for Reuse." in proceedings of *Pattern Languages of Programming*. Allerton Park, Illinois, USA: http://jerry.cs.uiuc.edu/~plop/plop98/final_submissions/P38.pdf.
5. Cybulski, J.L. (1996a): "Introduction to Software Reuse.", Technical Report TR 96/4, The University of Melbourne: Melbourne.
6. Cybulski, J.L. (1996b): "Reusing Informal Requirements: Review of Methods and Techniques." in proceedings of *1st Australian Workshop on Requirements Engineering AWRE'96*. Monash University, Caulfield, Victoria, Australia, p. 2.1-17.

[7] REBOOT implements the software development process with reuse [Morel+1993; Sorumgard+1993].

[8] F3 (From Fuzzy to Formal) is the reuse environment for supporting the requirements engineering process [Castano+1994].

7. Cybulski, J.L. and M. Mackowiak (1997): "Teacher's MATE: Multimedia Assisted Teaching Environment." in proceedings of *"Doing IT at Melbourne" Symposium*. University of Melbourne, p. 56-61.

8. D'Alessandro, M., P.L. Iachini, and A. Martelli (1993): "The Generic Reusable Component: an Approach to Reuse Hierarchical OO Designs." in proceedings of *Second International Workshop on Software Reusability "Advances in Software Reuse"*. Lucca, Italy: IEEE Computer Society Press, p. 39-46.

9. Hirata, K., Y. Hara, H. Takano, and S. Kawasaki (1996): "Content-oriented Integration in Hypermedia Systems." in proceedings of *Hypertext '96*. Washington DC, USA: http://www.cs.unc.edu/~barman/HT96/P42/Hirata.htm.

10. Ip, A., R. Canale, P. Fritze, and G. Ji (1997): "Enabling re-usability of courseware components with Web-based 'Virtual apparatus'." in proceedings of *ASCILITE 97*. Curtin University of Technology, Perth, WA, Australia: http://www.curtin.edu.au/conference/ASCILITE97/papers/Ip/Ip.html.

11. Li, H. (1993): "Reuse-in-the-Large: Modeling, Specification and Management." in proceedings of *Second International Workshop on Software Reusability "Advances in Software Reuse"*. Lucca, Italy: IEEE Computer Society Press, p. 56-65.

12. Lubars, M.D. (1988): "Wide-Spectrum Support for Software Reusability," in *Tutorial: Software Reuse: Emerging Technology*, W. Tracz, Editor. Computer Society Press. p. 275-281.

13. Morel, J.-M. and J. Faget (1993): "The REBOOT Environment." in proceedings of *Second International Workshop on Software Reusability "Advances in Software Reuse"*. Lucca, Italy: IEEE Computer Society Press, p. 80-88.

14. Prieto-Diaz, R. and G. Arango (1991): "Domain Analysis and Software Systems Modeling." . Los Alamitos, California: IEEE Computer Society Press.

15. Prieto-Diaz, R. and P. Freeman (1987): "Classifying Software for Reusabililty." *IEEE Software.* ***4***(1): p. 6-16.

16. Roschelle, J. and J. Kaput (1997): "Educational Software Architecture and Systemic impact: The Promise of Component Software.", University of Massachusetts, Dartmouth, http://www.simcalc.umassd.edu/simcalc/library/S&SImpact.hqx

17. Rossi, G., D. Schwabe, and A. Garrido (1997): "Design Reuse in Hypermedia Applications Development.", http://www-lifia.info.unlp.edu.ar/~garrido/

18. Shih, T.K. and R.E. Davis (1997a): "IMMPS: A Multimedia Presentation Design System." *IEEE Multimedia.* ***4***(2): p. 67-78.

19. Shih, T.K., H.-C. Liu, C.-H. Kuo, and T.-F. Kuo (1997b): "A Distributed Multimedia Database Supports Collaborative Learning." in proceedings of *Midwest Computer Conference '97*. University of Illinois at Springfield, Illinois, USA: http://www.uis.edu/~mcc97/html/papers/kuo/.

20. Sorumgard, L.S., G. Sindre, and F. Stokke (1993): "Experiences from Application of a Faceted Classification Scheme." in proceedings of *Second International Workshop on Software Reusability "Advances in Software Reuse"*. Lucca, Italy: IEEE Computer Society Press, p. 116-124.