Multimedia Process Quality Pattern

Jacob L. Cybulski and *Tanya Linden*
j.cybulski@dis.unimelb.edu.au *t.linden@dis.unimelb.edu.au*
http://www.dis.unimelb.edu.au/staff/jacob *http://www.dis.unimelb.edu.au/staff/tanya*
ph: +61 3 9344 9244 *ph: +61 3 9344 9250*
fax: +61 3 9349 4596 *fax: +61 3 9349 4596*

Department of Information Systems
University of Melbourne
Parkville, Vic 3052, Australia

Abstract

This paper describes a systematic approach to multimedia application development that emphasises reuse of multimedia objects and at the same time addresses quality and productivity of the development process. The approach is described with the aid of a process quality pattern `Analyse Organise Synthesize.` The pattern describes a methodology that addresses issues of multimedia product development and discusses maintenance of a multimedia repository. By applying this pattern developers may improve their productivity and enhance quality of individual components and entire multimedia products.

1. Introduction

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. Reuse of multimedia artefacts 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 [2, 4], 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.

When we started developing multimedia material for teaching Systems Analysis and Design we realised that a properly maintained multimedia repository will bring benefits into teaching not only this subject but will also improve sharing of examples and concepts across related subjects. 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. Hence, the systematic approach to organising multimedia repository and reuse of multimedia components will help developing high quality multimedia products, increasing teachers' productivity, saving time and reducing development cost.

We adopted a development process, which comprises three distinct activities, i.e. *analysis* of multimedia products and legacy documents, *organisation* of reusable artefacts and maintenance of a multimedia repository, and finally the *synthesis* of new products of reusable components. We recorded this process in the form of a pattern.

2. Multimedia Process Quality Pattern

Problem

Need for multimedia development process. Multimedia design methods currently in use have been adopted from the fields of publishing, technical writing and art design. These methods may not be fully effective when applied to modern multimedia development. Traditionally they focus on artwork but largely ignore the development process. Thus they may not be fully effective when applied to large multimedia projects where a more rigorous engineering approach is more appropriate. We need a more systematic approach to multimedia development, which results in high quality products and increased productivity.

Context

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. When creating new products, multimedia developers need to effectively and efficiently find, combine and create multimedia components.

Sharing and reuse. 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.

Project requirements. Any new multimedia project normally starts with the statement of requirements. In a traditional teaching environment, such requirements are usually high-level and informal, e.g. a course outline, a subject syllabus, a lecture or tutorial plan, or a project guide. The complexity of courses offered on-line, enforces a more rigid planning strategy where the traditional course design may be enhanced with some operational plan, such as an interaction script, user scenarios, a story board or a navigation map.

Forces

Component understanding. Successful reuse of product components needs tool support, however, apart from component storage and retrieval, full automation of multimedia reuse is difficult. Activities such as describing, indexing, classifying and finding multimedia components require developer intervention.

Component identification. Reuse of multimedia components relies on their effective finding and identification in existing multimedia products, and the subsequent representation of their information. These tasks, however, are difficult to accomplish for graphics, sound, movies, animation or executable programs.

Component access. Sharing and reuse of components in multimedia projects increases productivity but when components are scattered across the organisation it is hard to gain access to shareable components.

Component malleability. Multimedia artefacts have to be adapted to suit new project requirements, however, multimedia components that have not been specifically designed for reuse do not easily yield to modification and generalisation.

Solution

Development of multimedia products necessitates reuse. Multimedia reuse can be facilitated with a central repository of shared and reusable components. Since multimedia reuse cannot be fully automated, it is developers who have to be aware of multimedia reusability in all of their development tasks. We, hence, suggest structuring these tasks into a development process that will embrace the construction and maintenance of shareable repository and will enforce component reuse. We, henceforth, describe and prescribe a number of activities that should be performed in the process of multimedia development (either individually, in a sequence, or in parallel) to achieve high reusability of multimedia contents.

1. ***Continually analyse the existing multimedia products:***
   - Identify potentially reusable components in existing multimedia and legacy systems.[1]
   - 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. A variety of classification and indexing methods are described and used by developers, e.g. facets[3], keywords, enumerated schemes[8] or media-indexes [5].
   - 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.
3. ***Synthesize 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.

[1] 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.

Resulting Context

As a result of following the process, an organisation will benefit by obtaining an extended repository of reusable multimedia artefacts. Legacy documents will be transferred into electronic form, possibly decomposed and generalised. Any component will be properly documented before being placed in the repository, which will facilitate finding and retrieval of required components. Components will be designed with reuse in mind, which will simplify components' modification and adaptation to suit new requirements. For example educational institutions may end up with a collection of templates for lecture slides, on-line assessment, and tutorial handouts which can be shared across subjects. The same case study(s) with small changes can be used in related subjects (e.g. Programming and Systems Analysis and Design) with all elements of solution being in the repository and relevant parts extracted and used in lectures, tutorials or revision classes. With artefacts being properly documented, classified, indexed and generalised where possible, they can be easily reused and shared within the organisation, which in turn should increase productivity of product development.

When to use this pattern

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.

Examples/Known uses

Several systems in multimedia and software development support the described process of product development.

- Software development systems: REBOOT [6, 8], F3 [1].
- Multimedia systems: Himotoki [5], Intelligent Multimedia Presentation System (IMMPS) [7] and our Multimedia Assisted Teaching Environment (MATE).

3. Future Work

In our work, we focus on the construction, organisation and management of reusable multimedia components. We are aiming at the creation of a comprehensive multimedia-reuse pattern language, which provides multimedia developers with guidelines for effective multimedia reuse. This pattern language addresses six dimensions of multimedia authoring and reuse, i.e. the contents and quality of artefacts, their arrangement and presentation, and the processes leading to their construction and utilisation. In our previous work [2] we presented the patterns focusing on artefact construction, contents, arrangement and presentation. The pattern presented in this paper addresses the quality dimension. In the future, the resulting pattern language is intended to assist the users (and developers) of multimedia authoring systems to more effectively identify, represent, generalise, classify, store, search and retrieve, select, adapt and integrate multimedia components and processes that manipulate them.

4. Bibliography

1. Castano, S. and V. De Antonellis (1994): *The F3 Reuse Environment for Requirements Engineering.* ACM SIGSOFT Software Engineering Notes. **19**(3): p. 62-65.
2. Cybulski, J. and T. Linden (1998): *Composing Multimedia Artefacts for Reuse*. In *Pattern Languages of Programming*. Allerton Park, Illinois, USA: http://jerry.cs.uiuc.edu/~plop/plop98/final_submissions/P38.pdf.
3. Cybulski, J.L. (1996): *Introduction to Software Reuse*, Technical Report TR 96/4, The University of Melbourne: Melbourne.
4. Cybulski, J.L. (1996): *Reusing Informal Requirements: Review of Methods and Techniques*. In *1st Australian Workshop on Requirements Engineering AWRE'96*. Monash University, Caulfield, Victoria, Australia, p. 2.1-17.
5. Hirata, K., Y. Hara, H. Takano, and S. Kawasaki (1996): *Content-oriented Integration in Hypermedia Systems*. In *Hypertext '96*. Washington DC, USA: http://www.cs.unc.edu/~barman/HT96/P42/Hirata.htm.
6. Morel, J.-M. and J. Faget (1993): *The REBOOT Environment*. In *Second International Workshop on Software Reusability "Advances in Software Reuse"*. Lucca, Italy: IEEE Computer Society Press, p. 80-88.
7. Shih, T.K. and R.E. Davis (1997): *IMMPS: A Multimedia Presentation Design System.* IEEE Multimedia. **4**(2): p. 67-78.
8. Sorumgard, L.S., G. Sindre, and F. Stokke (1993): *Experiences from Application of a Faceted Classification Scheme*. In *Second International Workshop on Software*

Reusability "Advances in Software Reuse". Lucca, Italy: IEEE Computer Society Press, p. 116-124.