RICH DIGITAL BOOKS FOR THE WEB

Rui Lopes, Hugo Simões, Carlos Duarte, Luís Carriço
LaSIGE, University of Lisbon, Edifício C6, Campo Grande, 1749-016 Lisboa, Portugal
rlopes@di.fc.ul.pt, hsimoes@di.fc.ul.pt, cad@di.fc.ul.pt, lmc@di.fc.ul.pt

Keywords: Rich Digital Books, Automated Book Production, Behavioral Dimensions, Web User Interfaces.

Abstract: This article presents an architecture for the production and delivery of Rich Digital Books on the Web. These books are transformed and enriched with supporting media, like images and sound, pursuing the goal of reaching broader audiences and enlarging usage possibilities. The architecture affords for the production process' automation, and thus has the potential to increase digital books' availability. The architecture also enjoys a high flexibility degree, allowing the production of books that can be read with Web-based technologies, such as Web browsers.

1 INTRODUCTION

Before digital libraries, several barriers were in the way of everyone's right to information, ranging from availability to information retrieval. But digital libraries radically changed the way people look at books. Nowadays, space congestion problems and maintenance costs are reduced, in comparison to traditional libraries. On the other hand, users are able to read books in the comfort of their homes and quickly search for information.

Since the ground basis for digital libraries has been widely deployed, other issues became more relevant. Digital libraries and digital books should be accessible for any user, in any usage scenario. Visually impaired persons, children, students, and average users should have access to information. However, current digital book technologies do not cope with these specificities. Consequently, digital books must be tailored and enriched to users' needs. As it is impossible to have a single environment to cope with all usage scenarios, different user interfaces must be provided by different reproduction platforms.

With this range of possibilities, manually producing Rich Digital Books (RDBs) will be time consuming and error prone. Therefore, it should be as automated as possible, driving the focus of manual tasks to specialized activities (e.g., describing multimedia contents' semantics). Having the ability to deliver automatically multi-device Web-based solutions enables a wider acceptance and dissemination of produced books.

This paper focuses on an architecture for automated production of RDBs targeted for Web-based reading activities, covering the different requirements on multimedia content, users' profiles and usage scenarios. Some examples of Web-based reproduction platforms are also presented.

2 REQUIREMENTS

Reading is highly influenced by the reader's goals. Whether reading a novel for entertainment purposes, or studying a textbook, these activities engage the reader with different levels of commitment and attention. To portray different kinds of reading, a categorization of reading situations based in two dimensions (nature of engagement, and the activity's breadth) was proposed (Schilit et al., 1999): passively reading a single text; passively reading multiple texts; actively reading a single text; and, actively reading multiple texts.

While understanding the text is a common goal for all reading situations, they pose different problems. Situations encompassing multiple texts entail the need to manage multiple documents and the difficulty of finding needed information. Active reading (Adler and Doren, 1972) involves underlining, highlighting and annotating, either on the text or in a separate notebook, thus demanding for annotation management.

Digital books and digital libraries contribute to mitigate some of these problems. The latter make it possible to manage large book collections, and to create and explore relations between books, while books offer the possibility to record, organize and search annotations, increasing the possibility of sharing personal annotations within a community (Kaplan and Chisik, 2005).

Besides this, the book's digital support opens up the possibility of enriching its content with supporting media (Carriço et al., 2003). If allowed by the reproduction platform, the book can have its content enriched with additional multimedia content. In such a platform (e.g., Web browser), the book's content can also be narrated in addition or in alternative to the visual presentation, similar to Digital Talking Books (DTBs) (ANSI/NISO, 2002).

RDBs must be able to reach a heterogeneous audience in a variety of situations (researchers, students, occasional readers, children, elderly, the blind, etc.). Thus, reproduction platforms should be tailored for usage patterns and user profiles, coping with a specific set of features: support advanced annotation mechanisms (multimedia annotations, filters, etc.), advanced navigation (table of contents, lists, etc.), situational awareness, and interaction with a multimedia repository in order to augment RDBs in playback time.

All these requirements will directly influence books' characteristics. To be able to meet them, the book must pass a production stage. This will ensure not only that the book's format will be presentable in the selected reproduction platform, but also that additional enriching materials will be coherently selected.

If book production can be automated, all books in a digital library can be converted to a common format, and benefit from the possibilities offered by the reproduction platform. Having this automation on production also opens the way on information repurposing and creative combination of media contents, allowing authors and publishers to manage different book editions.

To fit all these different aspects in an RDB creation process, production architectures must be modular to leverage book creation and maintenance. Consequently, the following production time requirements must be taken into account: provide a modular and composable content processing configuration; define a strict content format, rich enough to support multimedia composition; add content to a repository, regarding transclusion scenarios; support addition of new material to the initial content; provide a clear separation between book content and its user interface; define a reusable specification of user interfaces, enforcing coherence amongst usage scenarios; and ease prototype features testing.

Having flexibility in the book production process raises issues regarding production time users. As such, three user profiles must be supported: top level user, power user, and developers. The first is a user with less technical expertise, whose tasks relate to manage and annotate content, or control bookset production. The second relates to those who have high knowledge over digital publishing, requiring a full control of book production to create production profiles. Lastly, developers are specialized in creating digital publishing components.

In order to support different RDBs scenarios, a multimedia repository must be available both on production and reproduction times. Consequently, the following activities must be supported regarding multimedia content management (Cybulski and Linden, 1999): continuous identification, classification, and organization of multimedia items; converting, and structuring data into a normalized format, in order to be indexed and stored in the repository; establish relationships between media items, based on different criteria (e.g., semantic, composition based, etc.); and support online query and retrieval of those items, to be integrated into manual and semi-automatic content production tools.

3 RDB PRODUCTION

Based on these requirements, a flexible architecture for automated production of RDBs has been defined. Figure 1 presents the proposed production architecture, divided into different concerns.

The initial set of inputs is fed to the production architecture, where it is transformed, augmented, and/or simplified according to a given production profile, through content processing and structure repurposing concerns. A multimedia repository is available to both concerns as a

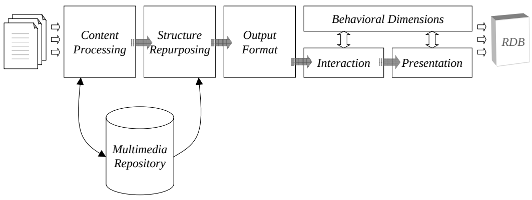

Figure 1: The production architecture.

way to enhance the book's content. Afterwards, the target reproduction platform is chosen, by specifying the required output format concern. Finally, to increase the flexibility of the production architecture, a set of behavioral dimensions can be filled by interaction and presentation concerns, or left to be dealt by the target RDBs player. At the end of the architecture, an RDB is available for the selected reproduction platform and user profile. Having modular concerns as mechanisms to handle the different aspects found along the production process, meets the production requirements gathered previously.

Each production time user's specific issues are supported by the production architecture. At a lower level, developers define processing tasks for each concern. On top of it, these tasks are aggregated into production profiles, regarding specific requirements (e.g., user profile, publisher's presentation specificities). Finally, top level users control batch production of books, selecting appropriate concerns or production profiles.

3.1 Content Processing

The increase on production and use of rich contents requires an efficient, and reliable multimedia content management. However, this presents unique questions, such as the wide variety of complex formats, or the need to associate these with the proper application information. To handle these issues, the processing architecture's first concern deals with different tasks centred on book content processing. As a wide range of data formats is potentially available as input (e.g., DTB, HTML, PDF, timed text, etc.), an initial content format normalization task is required. This normalization uses a book content format rich enough to cover the complex tasks to be applied later, along the lines of hypermedia reference models (Hardman et al., 1994).

After this step, content reasoning tasks are performed. These can be classified as manual, semi-automatic, or even fully automated, depending on the content's complexity. For instance, a semantic analysis of a book excerpt is difficult to be performed automatically, while a syntactic analysis requires little to none user intervention. Therefore, a multimedia repository was created, to sustain these tasks on RDB production, mainly through its multimedia content indexing and retrieval facilities. This eases RDB's rich content access and distribution.

Integrating such a repository of semantically indexed media will assist the production of media enriched books. Figure 2 illustrates how the content reasoning tasks were designed. This repository needs to be able to store both raw (e.g., acquired from the Web) and processed items (e.g., obtained from classification and composition components). Moreover, a multimedia content manager component needs to provide repository indexing and retrieval facilities.

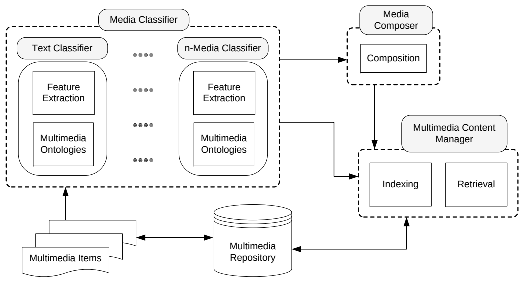

Figure 2: Classifying and storing multimedia items.

The media classifier component aggregates a wide variety of dedicated classifiers, each one accountable for a specific media type (e.g., text, image, video, etc.). Each classifier performs two tasks: extract content features and create or reuse existing multimedia ontologies, providing a semantic description for the media item. The first task is performed by a feature extraction component, responsible for content reasoning at different levels. This task is geared towards text categorization, understanding portions of an image, analysing an audio item, or establishing relationships between different elements.

The multimedia interpretation and annotation capabilities must be supported either by manual, semi-automatic, or automated tools. Hence, concerns dealing with semantic multimedia analysis and annotation must be followed. For instance, in the case of image annotations, pattern features extraction for edge detection, regions or texture analysis must be employed. Afterwards, decisions have to be made on how to represent the extracted features, and describe methods for their representation. To do this, a multimedia ontologies component provides an adequate way for representing the generated annotations.

To support these media annotations, this component is expected to follow a compliant format and allow authoring of semantically annotated documents. In this context, knowledge is represented with RDF and ontologies. A set of ontology derived semantic tags must be created to describe annotated media content features. This component may use new ontologies for media-specific domains, but it can also import and extend already existing ontologies (such as MPEG-7 Visual Part to describe and relate media features). Having this normalization assists interoperability and information reuse and availability through multimedia ontologies.

Another component relates to media composition, regarding future usage scenarios. This component is able to compose raw or already composed multimedia items. The rules for the composition process can be defined by simple attributes (such as matching metadata), or by complex algorithms (e.g., semantic inferences). Afterwards, the resulting composition is stored in the repository, for further processing instances.

The last component of content processing tasks is the multimedia content manager. This component has to support semantically indexed media, and adequate retrieval facilities in a flexible and efficient way. This information is stored and indexed in a multimedia repository, coupled with their structured descriptions. Additionally, media should be retrieved based either on simple queries, or even based on semantic relationships of different media contents.

3.2 Structure Repurposing

The second concern relates to content structure repurposing. Initially, different tasks should provide powerful content reasoning features, for instance, multimedia repository bi-directional feeding, enforcing content reuse mechanisms and repository enrichment with the RDB's content (to be used in other processing instances).

Afterwards, structure extraction tasks can be applied to the normalized content. A simple example is the extraction of a table of contents, an image list, or a table list, as independent structured content modules. Decoupling these structures can be helpful for content navigation.

Lastly, some control over content structures could be performed, regarding the deepness of these structures. This type of task should be applied, for instance, if a book is being processed towards lower computation resources.

3.3 Output Format

The third concern in the production architecture relates to output format conversion. This concern must define tasks for the conversion of normalized content structures, regarding the requirements for the target reproduction platform. As different scenarios must be taken into account, different formats must be supported. Richer formats allow the creation of specific interaction and presentation capabilities, whereas more limited platforms require simpler content formatting. Examples of target output formats are HTML+TIME (Schmitz et al., 1998), SMIL (Jeff Ayars, Dick Bulterman, *et al.*, 2001), or DTB.

Afterwards, different tasks can be applied to the current processed book content state, integrating user custom constructs (such as skeleton structures for bookmarking and annotation). These tasks should be applied if applicable within the chosen output format platform language.

This concern provides the minimum output to be played on an RDB reproduction platform, as some platforms are rich enough to provide flexible interaction and presentation capabilities. Therefore, the specificities for these two concerns are optionally used later on.

3.4 Interaction and Presentation

After the playback platform is chosen and the book's content is transformed into an output format, behavioral dimensions are introduced in the production architecture. These dimensions define how a reproduction platform should handle book interaction and presentation concerns. These are introduced as mechanisms to handle playback platforms' limitations around these behaviors in production time.

The first concern to be applied after book content output format choice relates to interaction mechanisms tasks. If the previously selected output format allows the specification of interaction, specific tasks implement different navigation interaction mechanisms regarding specific interaction devices (e.g., mouse, keyboard, speech). Two types of interaction capabilities can be defined: the first enables the user to jump towards specific points in a book (e.g., direct click on the content), being the latter based on navigation patterns (e.g., table of contents interaction triggers a shift in the content presentation focus).

This concern must take into account the different limitations on interaction defined by production profiles. These can relate to overly simplistic output formats, reproduction device capabilities, reader limitations and disabilities, or even the reader's surrounding environment. Nevertheless, these limitations can be overcome by introducing tasks, for instance, to limit speech recognition vocabularies in crowded environments.

Finally, the last concern in the book production defines how an RDB is going to be presented to the user, based on miscellaneous constraints. To ease presentation configurability, and to keep user interface coherence amongst different output formats, the architecture uses *presentation profiles*. Each profile is defined by a set of presentation rules, applied to the RDB's current content state. By combining different rules, different profiles can share common presentation features, thus enforcing user interface coherence.

Presentation rules implement a rich set of features based on users' requirements and device capabilities. Different patterns for presentation are defined by each rule (e.g., sound volume, coloring, dimensioning, resource limitations, etc.).

As a result, processing tasks must be provided in the presentation concern, per output format. These must be selected accordingly to apply a selected presentation profile into the RDB's current state. Moreover, by selecting richer presentation profiles, stricter behavioral dimensions are fed to the reproduction platform.

4 RDB REPRODUCTION

The flexibility of the production process allows output of digital books tailored to a reader's desired format. This means that rich digital books can be available early for immediate presentation, tailored to different devices and Web-based platforms. These include output formats that combine text with audio (e.g., SMIL), or even HTML+TIME documents, presentable using Microsoft Internet Explorer, as shown in Figure 3. Here, a book is presented with direct content navigation, as well as navigation capabilities of table of contents and sidenotes., coupled with synchronization guidance between text and audio.

Towards minimal playback platforms, SMIL is adequate for RDBs. It integrates no navigation

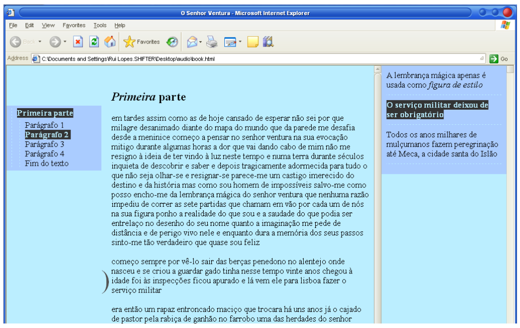

Figure 3: HTML+TIME book in Internet Explorer.

mechanisms, although the display of current table of contents item is allowed, next to the main content. The display of these contents is synchronized with the audio file, regarding the granularity chosen at the structure repurposing level.

Audio-only books can also be reproducible on a SMIL player. This kind of presentations introduces an audio guide (in the form of a beep played in parallel with the main audio), to help on synchronization aspects. Sound volume leveling between the spoken text and the synchronization guides is also taken in consideration. Other document formats, targeted at more specific user populations, may also be generated, like a Braille version of the book, for print disabled readers, demonstrating the production process flexibility.

5 RELATED WORK

Nowadays, DTB production is usually performed by experts, through automated or manual methods. As DTBs are mainly targeted to the blind, frameworks do not take into account other user profiles or usage scenarios. On the automated approach, text-to-speech is used to generate audio tracks and synchronize text with audio, leading to bad acceptance from users due to its robotized voices, and to ambiguous interpretation of textual content. In opposite, manual production becomes too expensive for book collections, due to synchronization efforts.

RDBs can be generalized to the notion of time-based hypermedia, having different content sources with linking capabilities and time composition. Based on this, automated architectures have been proposed (van Ossenbruggen et al., 2001), allowing for a constraint-based automated creation of media-centric presentations (e.g., devices, user models, presentation specifications).

At the end, only SMIL contents are delivered, therefore lacking text-based formatting.

Regarding reproduction, a list of playback devices capable of simple DTB reproduction was made available (Daisy Consortium, 2006). However, no player meets all the requirements gathered before, and, in a previous evaluation of their features, some usability and accessibility flaws have been uncovered (Duarte and Carriço, 2005). This set of limitations is overcome by the reproduction platforms described in this paper. Extending these platforms outside the Web has been done previously (Duarte and Carriço, 2006), to be able to cope with richer reproduction scenarios (i.e., real-time user interface adaptation engines).

6 CONCLUSION

To increase digital books availability, it is essential to upgrade production processes towards automation. This evolution will increase digital books acceptance and, combined with Web-based reproduction platforms, can lead to a greater adoption. This paper presented an RDB production architecture to move us closer to such a vision. The proposed architecture supports goals such as providing the same "brand" for a digital library, or preparing special editions of books targeted to impaired audiences, people with learning disabilities, or children learning to read.

To allow for the production of such books, the architecture is concerned with mechanisms to normalize content, repurpose structures, and output formatting. The preparation of special editions is made possible with a close integration of a multimedia repository, to enrich books' contents. Moreover, books' contents are also added to the repository for future uses.

Different capabilities are provided by reproduction platforms, pushing the limits of each Web-based technology supported by the production platform. This enforces RDB adoption, increases the spectrum of readers and reading situations, as presented in this paper.

ACKNOWLEDGEMENTS

This work is being funded by Fundação para a Ciência e Tecnologia, through grant POSI/EIA/61042/2004, and scholarship SFRH/BD/29150/2006.

REFERENCES

Adler, M. J. and Doren, C. V. (1972). *How to Read a Book*. Simon and Schuster, New York.

ANSI/NISO (2002). Specifications for the digital talking book. Available at http://www.niso.org/standards/resources/Z39-86-2002.html.

Carriço, L., Guimarães, N., Duarte, C., Chambel, T., and Simões, H. (2003). Spoken books: Multimodal interaction and information repurposing. In *Proceedings of HCII'2003, International Conference on Human-Computer Interaction*, pages 680–684, Crete, Greece.

Cybulski, J. and Linden, T. (1999). Designing multimedia development environments with reuse in mind. In *10th Australasian Conference on Information Systems ACIS'99*, pages 235–246, Wellington, New Zealand.

Daisy Consortium (2006). Playback tools. Retrieved January 18, 2006 from http://www.daisy.org/tools/playback.asp.

Duarte, C. and Carriço, L. (2005). Users and usage driven adaptation of digital talking books. In *HCII '05: Proceedings of the 11th International Conference on Human-Computer Interaction*, Las Vegas, Nevada, USA.

Duarte, C. and Carriço, L. (2006). A conceptual framework for developing adaptive multimodal applications. In *IUI '06: Proceedings of the 11th international conference on Intelligent user interface*, Sydney, Australia. ACM Press.

Hardman, L., Bulterman, D. C. A., and Rossum, G. (1994). The Amsterdam hypermedia model: adding time and context to the Dexter model. *Commun. ACM*, 37(2):50–62.

Jeff Ayars, Dick Bulterman, *et al.* (2001). Synchronized Multimedia Integration Language (SMIL 2.0). W3C Rec. http://www.w3.org/TR/SMIL2.

Kaplan, N. and Chisik, Y. (2005). In the company of readers: the digital library book as "practiced place". In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries*, pages 235–243, Denver, CO, USA. ACM Press.

Schilit, B. N., Price, M. N., Golovchinsky, G., Tanaka, K., and Marshall, C. C. (1999). As we may read: The reading appliance revolution. *Computer*, 32(1):65–73.

Schmitz, P., Yu, J., and Santangeli, P. (1998). Timed Interactive Multimedia Extensions for HTML (HTML+TIME). W3C Note. http://www.w3.org/TR/NOTE-HTMLplusTIME.

van Ossenbruggen, J., Geurts, J., Cornelissen, F., Rutledge, L., and Hardman, L. (2001). Towards second and third generation web-based multimedia. *The Tenth International World Wide Web Conference in Hong Kong*, pages 479–488.