

DiTaBBu - Automating the Production of Time-based Hypermedia Content

Rui Lopes
LaSIGE/University of Lisbon
Campo Grande, Edifício C6
1749-016 Lisboa, Portugal
rlopes@di.fc.ul.pt

Luís Carriço
LaSIGE/University of Lisbon
Campo Grande, Edifício C6
1749-016 Lisboa, Portugal
lmc@di.fc.ul.pt

Carlos Duarte
LaSIGE/University of Lisbon
Campo Grande, Edifício C6
Lisboa, Portugal
cad@di.fc.ul.pt

ABSTRACT

We present DiTaBBu, Digital Talking Books Builder, a framework for automatic production of time-based hypermedia for the Web, focusing on the Digital Talking Books domain. Delivering Digital Talking Books collections to a wide range of users is an expensive task, as it must take into account each user profile's different needs, therefore authoring should be dismissed in favour of automation. With DiTaBBu, we enable automated content delivery in several playback platforms, targeted to specific user needs, featuring powerful navigation capabilities over the content. DiTaBBu can also be used as testbed for prototyping novel capabilities, through its flexible extension mechanisms.

Categories and Subject Descriptors

H.5.2 [Information Interfaces and Presentation]: User Interfaces; H.5.4 [Information Interfaces and Presentation]: Hypertext/ Hypermedia; I.7.2 [Document and Text Processing]: Document Preparation

General Terms

Design, Human Factors.

Keywords

Hypermedia, Digital Talking Books, DiTaBBu, Automatic presentation generation, Multimodality, Accessibility.

1. INTRODUCTION

As the world switches from analog to digital media, blind or print disabled persons have the chance to access richer talking books, through a larger feature set and better sound quality, in comparison to analog talking books (i.e. analog audio tapes). As such, the Digital Talking Books (DTBs) specification [3] appears as a solution for information technology targeted to visually impaired persons, based on subsets of XHTML and SMIL [4]. DTBs can also be a base for production of rich DTBs [1], where new multimedia content and rich navigation capabilities are incorporated into the main content. These can be targeted to different user groups: visually impaired users can benefit from audio-only books navigation features; partially sighted users will have

an increased user experience through visual book presentation enhancement; reading or writing disabled persons (e.g. children) can learn how to read, spell and speak correctly; users in constrained environments benefit from alternative book versions, such as audio centred books for situations where visual focusing is hard (e.g. while travelling by train). Based on each user group, different content features should be made available in a rich DTB, such as full audio and text, advanced navigation control, bookmarks, highlights and annotations. Such diversity must be reflected into a DTB player and dealt in design time [2] (e.g. cues for audio tracks, multimodal inputs, etc.). Lastly, to increase rich DTB availability, different devices and platforms should be targeted by production frameworks.

Having such diversity on rich DTB targeting, for an immense available corpora it is unfeasible to author them by hand. Current DTB automated production is based on text-to-speech technologies, resulting in ambiguous interpretation of textual content [6] and low quality DTBs delivery. Web based multimedia presentation engines [8] are capable of delivering limited DTBs, as these engines do not cope with dynamic layout flow (as DTBs are typically text based).

As such, rich DTBs should be produced automatically, by specifying high-level processing tasks accordingly to each target user group, enabling consistent book collections production. Therefore, a set of requirements must be taken into account, including: provide a separation between book's content and reusable UI specifications; integrate content along a timeline; cope with different input sources and output formats; lastly, allow different tasks to be performed by different users (i.e. a digital library maintainer selects content and triggers production, a designer specifies production profiles, a developer creates processing components).

2. DITABBU FRAMEWORK

DiTaBBu has been created as a framework for modular production of DTBs, in a fully automated process. It is composed by four conceptual stages (as seen on figure 1): structure repurpose, output format definition, interaction, and presentation. Each stage has a specific set of processing steps available (defined in XSLT through XML pipeline technologies [5]), used to perform transformations over an initial content, towards book final output.

The structure repurposing stage is responsible for different structure manipulation tasks: content normalization, structure extraction, and granularity control. Content normalization allows supporting different input content formats (e.g. PDF, XHTML, etc.). The normalized content is defined by

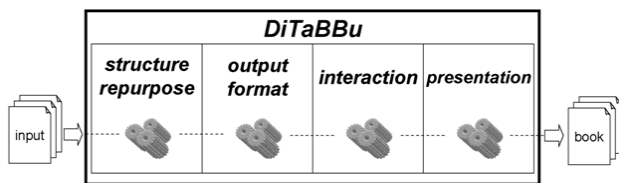


Figure 1: DiTaBBu stage definition and execution

the book’s main content and complementary contents and structures (such as table of contents or indices). Linking and time synchronization is provided through XLink based documents. Structure extraction tasks infer over the normalized content to find implicit navigation structures (e.g. producing a table of contents from the book’s content structure). Lastly, granularity control tasks allows for content structures’ flattening (geared towards low resources devices), reducing playback platform’s synchronization efforts.

Afterwards, the output format stage is responsible for transforming the normalized content structures into different output formats, as well as integration of bookmarks, highlights and annotations mechanisms. DiTaBBu currently supports three output formats, HTML+TIME [7], SMIL and DTB, but any other target format can be plugged into this stage, extending DiTaBBu’s delivery range. As each format has different features, an output format must be chosen regarding the feature-set a book producer wants to deliver.

On the interaction stage, input device support (currently mouse, keyboard and speech) integration is performed in the book’s content and navigation structures. Two interaction types may be applied: direct content navigation, and content navigation through auxiliary structures. The first enables the user to jump towards any location in a book content, whereas the second is based on predefined navigation patterns (e.g. table of contents). As each output format has its specificities and constraints, different interaction mechanisms have been implemented for each format. Having this stage separate from the output format stage enables better configuration of DiTaBBu’s automation process.

DiTaBBu’s last processing stage defines how a book is presented to the user. The approach taken is based on the *presentation profiles* concept. These profiles are defined as a set of presentation rules, with an agnostic language regarding the chosen output format from the previous processing stage. These rules are applied to the book’s contents and navigation structures, implementing a rich set of presentation features, thus opening the way for automated creation of DTBs with coherent UIs. Each rule specifies a presentation pattern for one of five settings: dimensioning, colouring, synchronization guidance, sound volume, and sound items.

3. EXPERIMENTATION

To test DiTaBBu’s flexibility, three different versions of an existing book (“O Senhor Ventura”, by Miguel Torga) were produced. Each version focused on different aspects on the automatic generation of DTBs, proving DiTaBBu’s flexibility. The initial book content was composed by an audio file with the spoken text, the book textual content and a set of sidenotes, and synchronization information. This synchronization was performed automatically with speech recognition software. The first version created was defined

to be available on an HTML+TIME player (Internet Explorer 6), with rich navigation mechanisms, synchronization presentation and complete presentation profile. The second version was targeted to a minimal playback platform (SMIL was the chosen output format). Content structures were trimmed down to paragraphs. No navigation mechanisms were integrated, albeit displaying the current presentation status concerning the table of contents. Regarding presentation, a stripped down profile was selected, with dimensioning for handheld devices. Lastly, the third configuration was defined to be audio only, geared towards visually impaired users. This version was also stripped down to its bare minimums, although reproduceable on a SMIL player. After transformation into the SMIL output format, an audio cue was introduced (in the form of a beep played in parallel with the main audio), to help on navigation tasks.

4. CONCLUSIONS AND FUTURE WORK

Accessibility is becoming a crucial issue for web based multimedia content, such as DTBs. As authoring tasks are error prone and time consuming regarding different kinds of users, as well as different usage scenarios, there is a need to automate production. DiTaBBu enables the production of complete book collections in an entirely automatic manner, with coherent UIs and interaction mechanisms, thus delivering more content to several target audiences.

Based on DiTaBBu, future directions are being delineated: transclusion mechanisms to support the integration of different kinds of external contents; integrate other content formats both on the initial input content and the final output format a DTB will be delivered; lastly, work is being done towards seamless integration between DiTaBBu and adaptive rich DTB playback platforms.

5. ACKNOWLEDGEMENTS

This work is partially supported by FCT (Fundação para a Ciência e Tecnologia), grant POSC/EIA/61042/2004.

6. REFERENCES

- [1] L. Carriço, C. Duarte, R. Lopes, M. Rodrigues, and N. Guimarães. Building Rich User Interfaces for Digital Talking Books. In *Computer-Aided Design of User Interfaces IV*. Kluwer Academic Publishers, 2005.
- [2] L. Carriço, N. Guimarães, C. Duarte, T. Chambel, and H. Simões. Spoken books: Multimodal interaction and information repurposing. In *Proceedings of HCI’2003, International Conference on Human-Computer Interaction*, pages 680–684, 2003.
- [3] Daisy 3.0 Specification, 2005. <http://www.daisy.org/z3986/2005/z3986-2005.html>.
- [4] Jeff Ayars, Dick Bulterman, et al. Synchronized Multimedia Integration Language (SMIL 2.0). W3C Rec., 2001. <http://www.w3.org/TR/SMIL2>.
- [5] R. Lopes and L. Carriço. APP - Architecture for Pipelined Processing. *IADIS International Conference WWW/Internet 2005*, 2005.
- [6] I. J. Pitt and A. D. N. Edwards. Improving the usability of speech-based interfaces for blind users. In *Assets ’96: Proceedings of the 2nd annual ACM conference on Assistive technologies*, pages 124–130, New York, USA, 1996. ACM Press.
- [7] P. Schmitz, J. Yu, and P. Santangeli. Timed Interactive Multimedia Extensions for HTML (HTML+TIME). W3C Note, 1998. <http://www.w3.org/TR/NOTE-HTMLplusTIME>.
- [8] J. van Ossenbruggen, J. Geurts, F. Cornelissen, L. Rutledge, and L. Hardman. Towards second and third generation web-based multimedia. *The Tenth International World Wide Web Conference in Hong Kong*, pages 479–488, May 1-5 2001.