BROWSING A STRUCTURED MULTIMEDIA REPOSITORY

Rui Lopes and Luís Carriço
LaSIGE, University of Lisbon, Edifício C6, Campo Grande, 1749-016 Lisboa, Portugal

Keywords: Multimedia Repository, Hypermedia, Semantic Browsing, Web User Interfaces.

Abstract: Browsing large amounts of assorted multimedia poses several challenges on creating a supporting user interface. Even further, if metadata, structures and relations are defined on top of multimedia fragments, the complexity of a multimedia repository increases. However, this is also a path for exploiting new browsing mechanisms. This paper presents insights on which aspects should be available on a task-oriented user interface for browsing a structured multimedia repository, centred on speeding up interaction through media independent navigation schemes and user advisory.

1 INTRODUCTION

Digital media support changed radically the way information is consumed. Nowadays, accessing digital contents is a common task, especially through the Web and its overwhelming size. Also, recent trends give the user an active role on producing contents on different media (blogs, photographs, podcasts, videos) and establishing relations between media components, instead of just passively consuming information.

With such a massive amount of information available, new challenges arise. Recent efforts (Berners-Lee et al., 2001) envision a Web of semantically described information, to be shared by different applications. With this framework, pieces of information can be mixed, repurposed, and interlinked according to different criteria (such as complementing an e-learning environment with content from an online encyclopedia).

To support such scenarios, there is a need for a storage and retrieval capable repository for multimedia contents, metadata, and relationships. Ever further, these contents should not be perceived as black-boxed pieces of information. With the imposition of content structuring, each fragment may have its own semantics associated, allowing for its reuse in other contexts.

In such a repository it is critical to have a large and rich multimedia corpora, and its corresponding metadata and relations. To enable this, some tasks can be automated (e.g., crawling for new content, infer knowledge), but some tasks may only be performed by humans (e.g, fixing incorrect information originated from automation).

To cope with the amount of contents and the complexity inherent of having metadata and relations in the repository, its management user interface must be carefully crafted. This paper discusses these browsing aspects in a web-based interface for managing a structured multimedia repository that supports the storage and description of multimedia information fragments.

2 GOALS AND REQUIREMENTS

The effectiveness of the user interface for browsing a structured multimedia repository is based on achieving the following set of goals:

- *Simple user interface*: as a huge amount of contents may need manual editing of their corresponding metadata, the user interface must not be an obstacle on performing these tasks;
- *Ease browsing of multimedia fragments*: browsing tasks should be centred around content frag-

Lopes R. and Carriço L. (2007).
BROWSING A STRUCTURED MULTIMEDIA REPOSITORY.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Web Interfaces and Applications*, pages 429-432
DOI: 10.5220/0001282404290432

ments, their siblings, children, etc.;

- *Alternate navigation methods*: having different methods for browsing the repository allows the user to select the best way for finding or reaching some multimedia content;
- *Overall views*: providing an overall view of the repository's contents helps the user to choose what tasks have to be performed;
- *Advice to the user*: the user must be advised with some pointers that otherwise would take too long to be found through exploration;
- *Collaboration*: the user interface has to promote multi-user activities, as a single user is unable to perform all required tasks.

These goals can only be achieved by fulfilling the following set of requirements:

- *Large scale*: the repository requires efficient forms for visualizing and browsing multimedia contents, as traditional efforts are geared towards single media types, or do not scale up;
- *Multitasking*: many tasks in the repository may trigger actions that will take some time to complete (e.g., uploading a video). This time can be used to perform other tasks;
- *Media agnostic*: the browsing capabilities should be independent from media types, resulting in a coherent user interface where users will perform better on any task;
- *Cope with structures and relations*: as the repository also contains structures and relations, these concerns should be explicitly present on the browsing interface;
- *Filtering*: the browsing interface should provide means for narrowing the browsable set of contents through filtering.

3 RELATED WORK

Typically, multimedia browsing is centred either on a single media type (Hürst and Stiegeler, 2002; Albanese et al., 2004). The fact that real scenarios must take into account all media types, poses severe difficulties on interacting with a multimedia repository. Even further, search tasks consume more time, as the user has to perceive the semantics of each content (instead of using metadata to provide this information instantly).

Recent studies (Lew et al., 2006) on content-based multimedia information retrieval have identified what challenges have not been achieved: creating new human centred methods based on exploratory interaction; enforce collaboration efforts; and provide multimedia assets taxonomic classification and browsing. Consequently, the architecture for a new multimedia repository must take into account these challenges.

4 OVERALL ARCHITECTURE

An architecture was defined as the ground basis for an environment based on structured multimedia fragments stored in a repository (Figure 1).

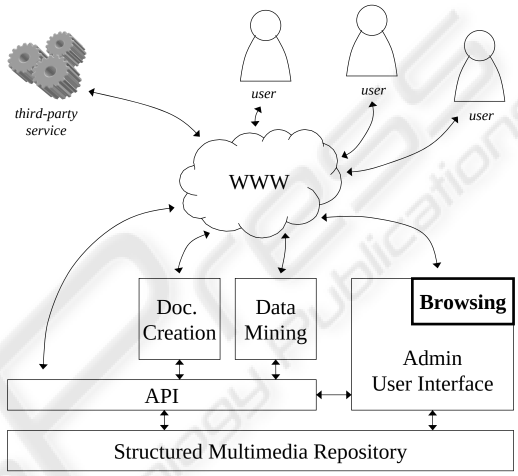


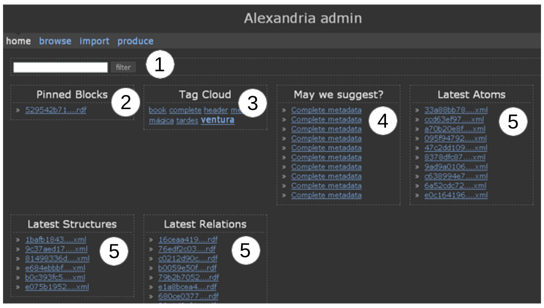
Figure 1: Overall System Architecture.

The main architectural decision relates to the Web's distributed nature as its central point. Every aspect within the architecture is a producer and/or consumer of resources through services with specific interfaces: an API is provided for developing these services on top of the repository (e.g., documents creation or third-party services), whereas a user interface provides repository management (including the browsing aspects).

5 BROWSING INTERFACE

A Web-based user interface was created for managing the structured multimedia repository. This interface supports different tasks, such as importing new content in different media formats, triggering multimedia document production tasks, editing content metadata, or creating structures and establishing relations between multimedia fragments. However, to perform these tasks successfully, users need to grasp the repository's contents through different points of view. Therefore, several browsing aspects within the user interface were defined and classified into two main concerns: *overviews* and *instances*.

Faced with millions of multimedia fragments, users may feel daunted on choosing a starting point for working on repository's contents. *Overviews* provide different possibilities for browsing the repository and start new tasks, as seen on Figure 2. Five overviews are provided:



Pinned Blocks:
- 529542b71....rdf

Tag Cloud: book complete header me... mágica tardes **ventura**

May we suggest?:
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata
- Complete metadata

Latest Atoms:
- 33a88bb78....xml
- ccd63ef97....xml
- a70b20e8f....xml
- 095f94792....xml
- 47c2dd109....xml
- 8378dfc87....xml
- 9ad9a0106....xml
- c638994e7....xml
- 6a52cdc72....xml
- e0c164196....xml

Latest Structures:
- 1bafb1843....xml
- 9c37aed17....xml
- 81498336d....xml
- e684ebbbf....xml
- b0c393fc5....xml
- e075b1952....xml

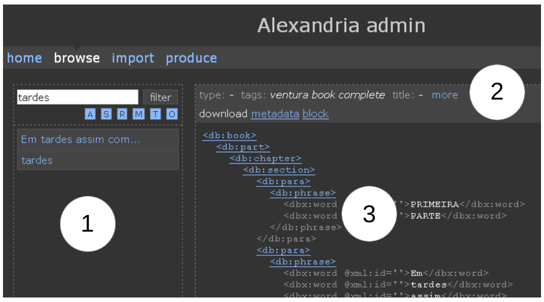
Latest Relations:
- 16ceaa419....rdf
- 76edf2c03....rdf
- c0212d90c....rdf
- b0059e50f....rdf
- 79b2b7052....rdf
- e1a8bcea4....rdf
- 680ce0377....rdf

Figure 2: Browsing *Overviews*.

1. *Filter*: a keyword-based filtering capability is provided to the user, allowing to dynamically adapt all other overviews accordingly. This feature enables the user to explore different perspectives and subsets of the repository without browsing a specific repository instance, helping the user to better perceive which contents the repository has;
2. *Pinned Blocks*: users are provided with mechanisms for pinning contents and viewing all pins, thus leveraging collaboration efforts;
3. *Tag Cloud*: with tags, users can freely classify a content according to their own perception. The cloud feature displays all tags that have been applied by users, with font sizes relative to each tag's count. This results on visualizing a user-created taxonomy of the repository's content;
4. *Suggestions*: a mechanism searches the repository for incomplete metadata, advising the user to fill the blanks. This feature enforces the completeness of the repository;
5. *Latest*: the user is presented with the latest contents and relations added to the repository, new starting points for browsing activities.

After choosing a starting point for browsing, the user is presented with its respective *instance*, as seen on Figure 3. Three different aspects can be perceived within the browsing instance:

1. *Search*: while browsing, the user has the option to search the repository. However, instance search feature enables more complex exploratory tasks, such as multiviews;
2. *Metadata summary*: to quickly grasp the subject of a given instance, the interface presents its corresponding ontology, tags and title, links for downloading content and metadata. Both ontology and tags summaries are navigable objects mapped into search tasks, providing the user with other browsing possibilities;
3. *Content*: this section presents the current instance's content. As the repository is based on structuring multimedia fragments, each structure is dynamically built and presented, enabling the user to grasp its full content. Also, each structure node is itself a navigation point for browsing its children within the repository.



tardes

Em tardes assim com...

tardes

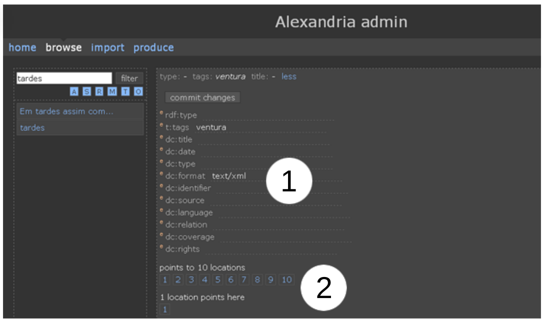
type: - tags: *ventura book complete* title: -

download metadata block

```
<db:book>
  <db:part>
    <db:chapter>
      <db:section>
        <db:para>
          <db:phrase>
            <dbx:word                 '>PRIMEIRA</dbx:word>
            <dbx:word                 '>PARTE</dbx:word>
          </db:phrase>
        </db:para>
        <db:para>
          <db:phrase>
            <dbx:word @xml:id=''>Em</dbx:word>
            <dbx:word @xml:id=''>tardes</dbx:word>
            <dbx:word @xml:id=''>assim</dbx:word>
```

Figure 3: Browsing *Instance*.

Expading the metadata summary, a full view is presented (Figure 4). Here, users are able to: (1) view and edit metadata fields, and (2) view different available navigation possibilities (inclusion, composition, and relations) for exploring the repository, based on the current browsing instance. It is worth noticing that metadata fields are also starting points for exploratory tasks from the user (similar to *Tag Cloud* browsing).



tardes

Em tardes assim com...

tardes

type: - tags: *ventura* title: -

rdf:type	
t:tags	ventura
dc:title	
dc:date	
dc:type	
dc:format	text/xml
dc:identifier	
dc:source	
dc:language	
dc:relation	
dc:coverage	
dc:rights	

points to 10 locations
1 2 3 4 5 6 7 8 9 10

1 location points here
1

Figure 4: Metadata view for Browsing *Instance*.

6 IMPLEMENTATION

The browsing user interface for the structured multimedia repository was implemented with Web-based technologies. As most users are comfortable on interacting with the Web, this will ease interaction with the browsing user interface. Simple multitasking is also leveraged just by opening a new window or tab in the browser, enabling its fast response. Another direct consequence is the instant availability of the user interface (no software installed on the client side), increasing its dissemination and usage.

The repository was implemented on top of eXist (Meier, 2002), a native XML database. This decision allows using XML formats for document structures (Walsh et al., 2002), metadata and relations. Also, using an XML database allows for volatile data schemes, as opposed to relational databases. Consequently, any ontology instance can be inserted on multimedia fragments metadata, thus enforcing extensibility to any knowledge inference mechanism. Executing queries on the repository is performed with XQuery (Boag et al., 2006). Different abstraction layers on information retrieval have been implemented, and may be used in the future as basic constructs for smarter mechanisms. On top of XQuery modules, the browsing interface has been implemented as a thin layer returning XHTML.

On top of the user interface, a set of unobtrusive javascript functionalities were added to improve the usability of its browsing characteristics. This way, tasks that require high computational resources (e.g., complex searching within the repository) can be triggered asynchronously, leveraging the user interface's responsiveness.

7 CONCLUSIONS AND FUTURE WORK

This paper presented new browsing aspects for managing a structured multimedia repository. These aspects were centred on speeding up exploratory activities within the repository, typically based on navigating between relations, structures, and semantic information of a huge amount of multimedia fragments. Two different browsing concerns were presented, *overviews* and *instances*, providing exploratory cues to start browsing the repository, and visualizing concrete fragments and their own browsing opportunities, respectively. These concerns are crucial, given the amount of fragments and the complexity of structures that may be related to each other.

As future work, it is desirable to support searching by example for images, sound and video resources. Also, allowing the repository metadata layer to reference any Web resource (instead of just referencing multimedia fragments stored inside of it) will ease coping with repository feeding. Furthermore, by introducing data mining techniques, better advice can be provided to users. Lastly, other collaboration techniques may improve the repository's browsing tasks, such as communities specialized in specific topics.

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

Albanese, M., Cesarano, C., and Picariello, A. (2004). A multimedia data base browsing system. In *CVDB '04: Proceedings of the 1st international workshop on Computer vision meets databases*, pages 35–42, New York, NY, USA. ACM Press.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American*.

Boag, S., Chamberlin, D., Fernández, M. F., Florescu, D., Robie, J., and Siméon, J. (2006). XQuery 1.0: An XML Query Language. W3C candidate recommendation, World Wide Web Consortium (W3C).

Hürst, W. and Stiegeler, P. (2002). User interfaces for browsing and navigation of continuous multimedia data. In *NordiCHI '02: Proceedings of the second Nordic conference on Human-computer interaction*, pages 267–270, New York, NY, USA. ACM Press.

Lew, M. S., Sebe, N., Djeraba, C., and Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19.

Meier, W. (2002). eXist: An open source native XML database. In *Proc. of 2nd Annual International Workshop on Web and Databases, NetObjectDays 2002*, Erfurt, Germany.

Walsh, N., Muellner, L., and Stayton, B. (2002). *Docbook: The Definitive Guide*. O'Reilly, 2.0.8 edition.