A Multi-layer Digital Library for Mediaeval Legal Manuscripts

Monica Palmirani and Luca Cervone

University of Bologna, CIRSFID
via Galliera 3, 40121 Bologna, IT
{monica.palmirani,luca.cervone}@unibo.it

Abstract. This paper presents the results of the MOSAICO project, an Italian Government research project (2008–12) funded by the Italian Ministry of Education and Research, and carried out by an academic consortium.[1] The goal of the Mosaic project (http://mosaico.cirsfid.unibo.it) is to create a thematic and specialized digital library, relying on the Web 2.0 and the P5 TEI XML standard to manage heterogeneous descriptions of medieval codex images. The portal is designed for scholars of medieval legal history and emphasizes the intellectual path of the academic experts.

1   Introduction

The European Commission is currently devoting much attention to the digital library goal[2] as a complex method for favouring the access to rare materials, for guaranteeing the long-term preservation of the cultural heritage, and for sharing knowledge by overcoming physical limitations. In the domain of medieval manuscript digitalization we find outstanding projects by libraries, institutions, and universities (Manuscripta Medievalia, a German consortium;[3] e-codices virtual manuscript library of Switzerland;[4] the Max-Planck-Institut für europäische Rechtsgeschichte;[5] the Enrich project database;[6] the Europeana Regia project;[7] Shared Canvas, managed by the University of Stanford and Los Alamos National Laboratory;[8] etc.) that over time have digitalized the manuscripts for future generations. Even if these projects, among others, define a robust backbone of the digital library initiative, they are much too oriented toward bibliographic description and classification of the material based on librarian criteria and codicological best practices 1 2, rather than being focused on allowing scholars to annotate the precious manuscripts through their expertise.

---

[1] University of Bologna, CIRSFID (coordinator); University of Federico II, Naples; University of Roma Tre.

[2] See 6.

[3] http://www.manuscripta-mediaevalia.de

[4] http://www.e-codices.unifr.ch/en

[5] http://dlib-pr.mpier.mpg.de/

[6] http://www.manuscriptorium.com/

[7] http://www.europeanaregia.eu/

[8] http://www.shared-canvas.org/

The goal of the MOSAICO project is to provide scholars with a very rich and easy to consult platform of manuscripts and a quick way to write content and metadata related to each manuscript, piece of artwork, page, or page fragment as other works are done 5. The power reached by modern Web applications permits to us to create a full in-browser system having, as its first convenience, the characteristic of providing a unified and collaborative venue in which historical scholars from all over the world can work together to improve the catalogue. The MOSAICO platform aims to store and manage each digital resource (content, metadata, images, comments to the images, etc.) in a neutral way so as to allow a multiform access to them on three different historical points of view: Roman, medieval, and contemporary.

2   Functionalities and Requirements

The Mosaico environment includes the following functionalities arrived at through several interviews and focus-group meetings within the consortium:

1. Collecting different digital materials on medieval legal manuscripts using patterns and templates.
2. Permitting scholarly annotations by writing text and hypertext using multiple templates available through a special Web editor. This approach makes it possible to preserve the intellectual and original logical structure designed by the author. We want to go beyond the rigid architecture of the DMBS, which forces authors to organize their thought on the basis of the database's logic layer. We aim to provide a Web editor capable of marking up in XML the metadata in the hypertext template.
3. Managing a plurality of templates of historical works on the basis of the different products expected (descriptive schedules, critical editions, comparative editions, multi-layer presentation, etc.).
4. Annotating manuscripts in XML format, so as to better manage the embedded knowledge and share it with a network of libraries across Europe. Further, the metadata will be recursively annotated in P5 TEI Enrich format, making it possible to overlay comments onto other comments, either hierarchically or in multiple and simultaneous fashion.
5. Comparing different manuscripts related to the same topic, thus creating an environment for historians to build, with the support of technology, a comparative critical edition.
6. Searching each codex's incipit and explicit using the roots of the Latin vocabulary.
7. Pointing-and-clicking on any image to bring up information relative to the codex being viewed and to its history.
8. Zooming in and out of the manuscripts and isolating a portion so as to focus on it.
9. Connecting the resources stored using association expressed in RDF.
10. Using a special viewer that can manage high-resolution images and in the meantime protect them (through the pyramid processing method) from illegal processing.

11. Including in each fragment of the zoom process the original library's watermarking.
12. Exporting metadata into P5 TEI Enrich XML format, making it possible to share material by way of digital-library initiatives across the world.
13. Using thumbnails any time the text cites an image.
14. Dynamic creation of interconnection tables among the digital resources on the basis of RDF assertion.
15. Managing the glosses and "tracce d'uso."[9]

The platform can manage security and IPR issues, block access to images, and track illegal misuse. Before to accessing the digital library collection, the user has to accept the terms of use.
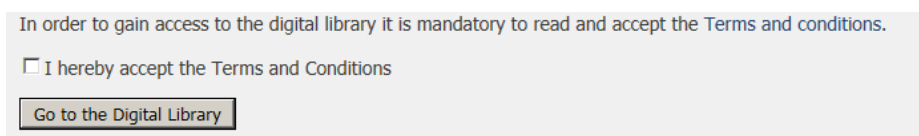
In order to gain access to the digital library it is mandatory to read and accept the Terms and conditions.

☐ I hereby accept the Terms and Conditions

Go to the Digital Library

Fig. 1. Legal terms and conditions

3   Patterns and Templates

One of the most important features is to use patterns and templates for the content, so as to lead the author in organizing the material.

With the help of the consortium partners, we have identified five patterns:
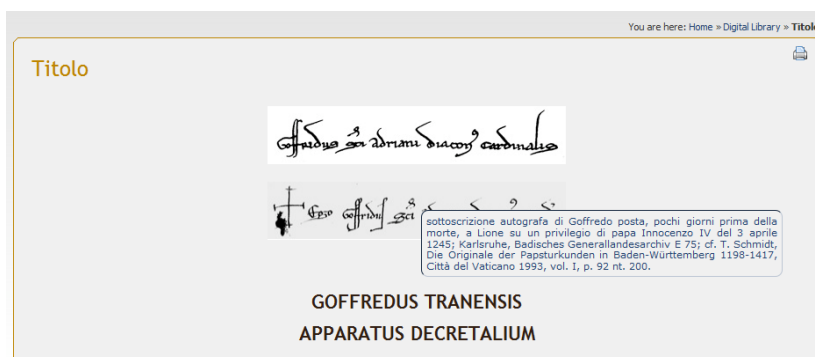
You are here: Home » Digital Library » Titolo

Titolo

sottoscrizione autografa di Goffredo posta, pochi giorni prima della morte, a Lione su un privilegio di papa Innocenzo IV del 3 aprile 1245; Karlsruhe, Badisches Generallandesarchiv E 75; cf. T. Schmidt, Die Originale der Papsturkunden in Baden-Württemberg 1198-1417, Città del Vaticano 1993, vol. I, p. 92 nt. 200.

GOFFREDUS TRANENSIS

APPARATUS DECRETALIUM

Fig. 2. Montecassino 266 description and mouse-over function

---

[9] *Tracce d'uso* ("traces of use") are annotations that students make on the code. They often record comments a professor has made during lecture. They are invaluable for scholars the medieval legal history, who can deduce the use and the interpretation of the code by the different schools of law.

1. Monographic and Hypertext Description of One Manuscript

In this template, the author describes the manuscript as a book and connects the images with the text using a hypertext model. The Montecassino 266 manuscript is an example of how Bertram described the images and connected them to the novellae.
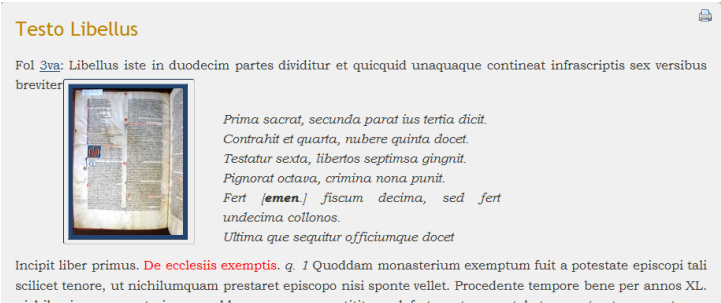
Image thumbnails are included in all parts of the text.

Fig. 3. Image thumbnail mentioned in the incipit the text
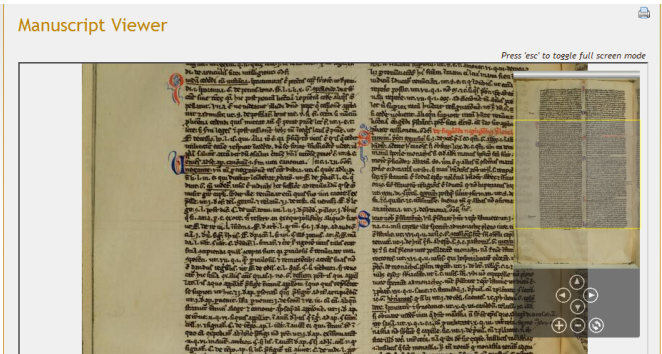
Clicking on the thumbnail will give access to the full page.

Fig. 4. Full page of Manuscript 266

Fig. 5. Comparative table of different manuscripts related to the same subject (*Authenticum*)

2. Comparison of Different Manuscripts on the Same Subject

The *Authenticum* includes 28 descriptions of the same subject. In the table below it is possible to see the different manuscripts of the *Authenticum* that Loschiavo se-

lected and described. The legend indicates the presence of images, related material, other previous historical descriptions, and a bibliography related to the code.

Fig. 6. Description of a manuscript from the *Authenticum*

Using RDF relationships among the different digital resources (bibliography, other multimedia material, images, etc.) it is possible to dynamically create a chronological table of the different available material in the database related to a specific resource.

Fig. 7. Chronological dynamic table of the resources in the database

3.1 List of Descriptions

This model (Fig. 8) is quite similar to applications that use relational databases. In this model users have a search mask in which they can see a list of tabs with metadata regarding several works. The user can do searches in the list in order to reach the needed work and can then click on the name of the work and read the tab. At this point you can also see the scans of the work using the reading tool described in the previous paragraph. This is a "multi editor" model in which there are several editors that write the metadata pertaining to the works.

3.2 Comparison of Different Transcription on the Same Subject

This model (Fig. 9) is used when there are several manuscripts that make up a unique "meta-manuscript." Also, there are several editors that write the metadata related to the manuscripts. In this case, users need to read the different pages and data in a

"side-by-side" model. So they can open a page on the left side of the reading tool and another page, probably belonging to another manuscript, on the right side so as to compare them and read the different metadata written for each of the pages.
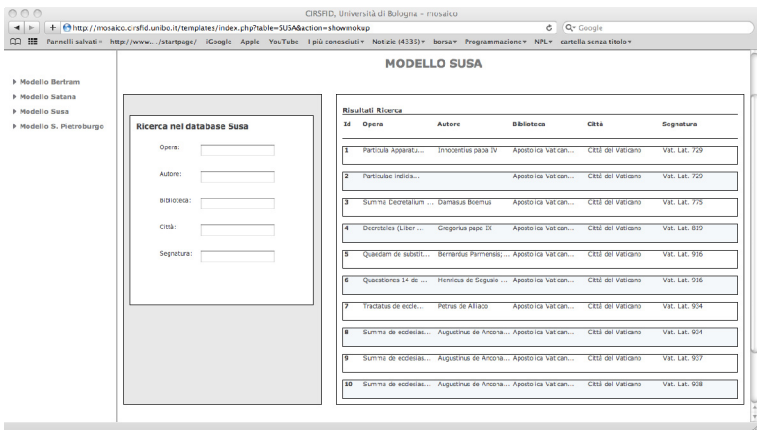
Fig. 8. List of descriptions
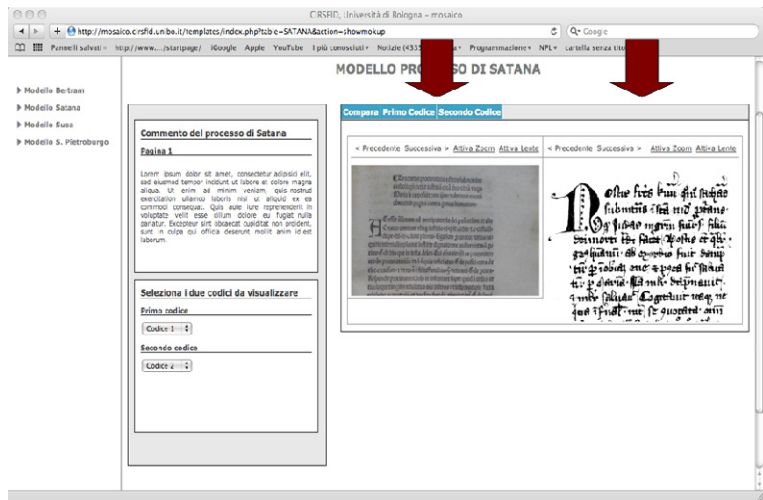
Fig. 9. Satana manuscripts: comparison windows

3.3   Temporal Sequence of Digital Material

The Saint Petersburg model" (Fig. 10) is used when there are different manuscripts and different metadata written by different editors over time. It is the most complex model for viewing metadata because there is a horizontal level of metadata and a vertical one. In this model the users can open a set of manuscripts and they can navigate them over time, so they can see the different metadata written over time navigating the

manuscript's vertical level, and they can see the different manuscripts navigating through the manuscripts' horizontal level.

Fig. 10. Authenticum's Saint Petersburg Manuscript with related material by Biner

4   The MOSAICO Architecture

MOSAICO is a Web application comprising two main elements: the server-side component and the client-side component. The server-side component is charged with performing ordinary operations on the database (data retrieval, saving, updating, deleting), ma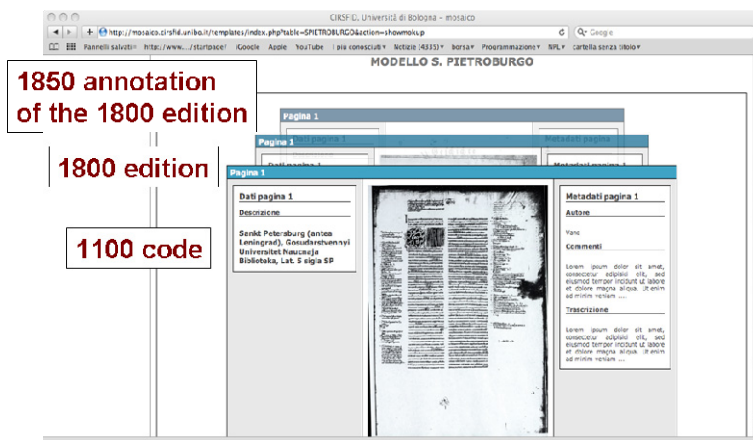king all the computation requested, and displaying the final results to the client side. The client-side component is charged with accepting user requests (made in a human-readable manner), sending them to the server side, retrieving the results, transforming them in a human-readable format, and, finally, presenting them to the users. In the MOSAICO project both the client and the server sides are made of other macro components.

4.1   The Server-Side Component

The server side component of the MOSAICO project is actually composed of three layer for corresponding three servers.

There are two servers that host the MOSAICO data repositories: the first one contains the XML repository and the second one the image repository. The main sever contains the MOSAICO application core and the packages that are used to communicate with the repositories and with the application's client-side component.

The core application, hosted by the main server, uses the MOSAICO repository manager to access the repository manager via HTTP, and it provides the MOSAICO API, which can be invoked by the MOSAICO portal to do simple and specific operations on the XML documents and on the images in the repository. For instance, when a reader-tool requests a page, the CMS calls, typically using a REST query, and

selects the appropriate method of the MOSAICO API, which dispatches the request to the application core. The application core, passing through the MOSAICO repository manager, retrieves the images and the XML related to the requested page, packages them to make everything readable by the reading tool, and, always using HTTP, returns the results to the client, which may carry out other formatting and presentation operations as needed and will finally reply to the user's request by supplying a human readable version of the materials related to the requested page.
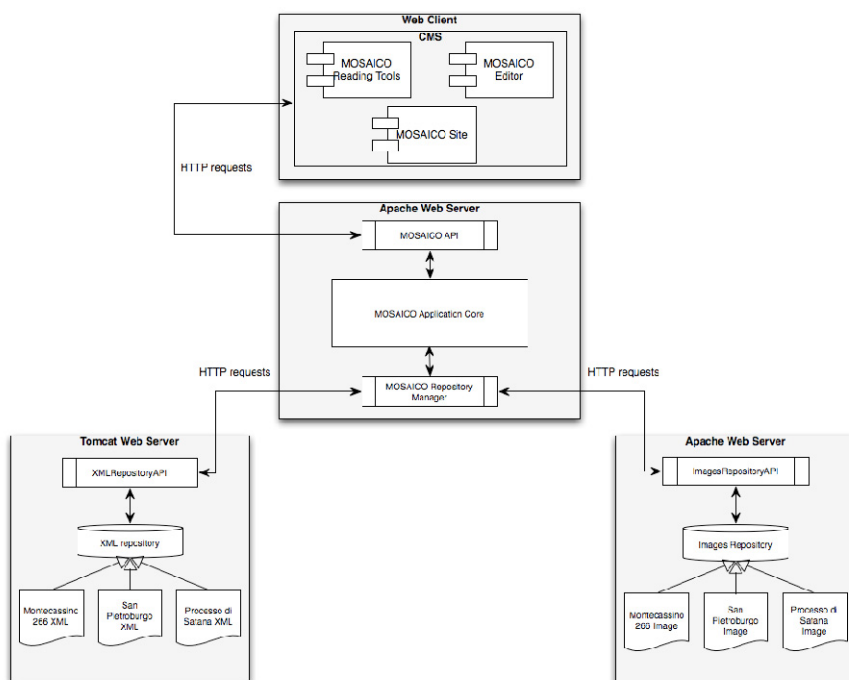
Fig. 11. MOSAICO architecture

4.2   The XML Repository

One of the main aims of this project is move past the idea of relational databases and start using an ML standard to markup the metadata relative to the manuscripts. This is because the descriptive tabs supplied by different historical experts can be formatted in very different ways. To store them in a relational database we need to identify any relevant partition, extrapolate it from the original document, and save it in an appropriate database table. But in this way we cannot preserve the structure of the original document, and that exponentially increases the risk of not being able to recreate the document. With XML we can mark up the relevant partitions preserving the original document format.

For this reason we need to use a native XML database that makes it possible to store the documents, and we also need to perform smart queries on the document collection. We choose the eXist database (http://exist.sourceforge.net/), currently the most widespread and supported native XML database. It comes with a built-in Tomcat Web server so it can be accessed through REST requests. In this way we can always keep separate the data (the document collection) from the application's other parts. So a service malfunction cannot corrupt the integrity of the document collection. Another point in favor of the system's security is that eXist is never queried by an external application but only by the MOSAICO repository manager in order to satisfy the application core requests. Another pillar of the MOSAICO project is the use of a permanent URI, after the FRBR[10] model (work, expression, manifestation, item: title/author/shelfmark), so as to have a permanent link for each resource independently of its physical storage in the image repository.

4.3 The Image Repository

The MOSAICO project collects a very large set of images that are essentially scans of the original manuscripts. These images are usually protected by copyright, so security policies are a main issue for the project. The best practice is to store them in a completely independent server accessed as a NAS (Network Attached Storage system) hosted on an APACHE Web server.

This is helpful in two respects:

- The images are protected from system failures. In other words, neither physical nor logical failure of the system can corrupt the integrity of the images.
- The images are protected from malicious attacks. The NAS is in a private LAN network, so the images can be accessed only by the application core (managing the repository API).

The reason for choosing to host the image repository in an APACHE Web server is because this makes it possible, even in this case, to use REST queries to send the application core's request (passing trough the MOSAICO repository manager). So all communication between the project's components are sent using an homogeneous communication architecture.

4.4 The MOSAIO Repository Manager

Both the eXist database and the NAS have APIs for access to the database. eXist provides a REST API that can be used to retrieve documents, upload documents onto the database, and send simple queries to the documents collection. REST (Representational Transfer Rate) is a paradigm that makes it possible to manage resources usinf the HTTP protocol.

---

[10] Functional Requirements for Bibliographic Records,
   http://www.ifla.org/en/publications/functional-requirements-
   for-bibliographic-records

Also, the NAS can be accessed via REST interrogations.

It is important to note that the MOSAICO repository manager do not perform any computation on the resource retrieved or uploaded. Simply it has in charge to satisfy the complex requests that are performed by the application core.

5   The Client-Side Component

The client-side component is the interface that users use to request actions to the server-side component of the MOSAICO project. Essentially, it is made up of a CMS that contains three main objects:

- **The MOSAICO site.** This is simply MOSAICO project's institutional site, and the start point from which to access the reading tools and the editor and to do queries on the database.
- **The MOSAICO reading tools.** The reading tools are software that makes it possible to read the scans of the original manuscripts of the MOSAICO document set alongside the relative metadata. There are several types of manuscripts and metadata, and for this reason, and in order to provide users with a good reading experience, there are not one but several types of reading tools, and each manuscript uses the most appropriate one.
- **The MOSAICO editor.** The MOSAICO editor is used to write MOSAICO documents or to mark them up, using the MOSAICO XML P5 standard on documents already written with any other text editor. The editor is under construction.

5.1   The MOSAICO Web Site

The MOSAICO site is the institutional site of the MOSAICO project. It contains all the information about the project and the consortium. It is also the bridge the get access to the digital library. The site's content s created and updated using an open source CMS named Impress CMS (http://www.impresscms.org). In order to present the site, the client side of the MOSAICO project, simply requests the page to the server side API. The server side replies with the content of the requested page and then the client side applies to it a specific style sheet and present the page to the user.

5.2   The MOSAICO Reading Tools

The MOSAICO project permits to read the scans of the original manuscripts in several ways. This is because the manuscripts and the related metadata come from very different heterogeneous sources. In order to read the scans of a manuscript the user must access to the "digital library" section of the site and choose the manuscript to read or the model to use to see the images and the related metadata of a specific manuscript. When a user requests a manuscript, or a single page, the client side component of the MOSAICO system, dispatches the request to the server side API. As we have seen previously in this document, the server side component performs all the

action needed to retrieve the data related to the requested item and creates a MOSAICO package. When the package is returned to the client side component, the client side understands by the info contained in the package what is the model to use in order to read the objects contained in the package, instantiates and present to the user the appropriate reading tool for those objects. There are several models that are used in the MOSAICO project in order to give to the user the best reading experience. All the reading tools are, in a technical language, AJAX (Asynchronous Javascript And XML) applications that use Javascript and run time calls to the server side component and use the best Web 2.0 techniques in order to supply fast and powerful pages reading.

5.3   The MOSAICO Editor

The MOSAICO editor is a WYSIWYG in-browser editor used to mark up document using in the MOSAICO XML format (a customization of the TEI P5 format http://www.tei-c.org/). The editor is now under construction, so we present here its main features. It will be able to perform all the basic operations of common text editors, and it makes it possible to create a new metadata tab (as well as RDF triples) on manuscripts and also to import a previously written metadata tab and mark up it in the MOSAICO XML format. The editor will be able to perform the following actions:

- **Creating a new document.** The user can create a new blank document and can use the editor to write the document natively in the MOSAICO XML format. It is important to note that, at any time while editing, the user can access the photo catalogue and see the images in the same window of the editor, taking advantage of the "side-by-side" editing feature.
- **Importing a document.** A common behavior is that users write a document (a metadata tab related to a manuscript) using third-party software, for instance, desktop software, and then they needs to mark up the document in the MOSAICO XML format and upload it in the MOSAICO digital library system. Users can use the editor to import the document and can tag the different parts of the document assigning to each part the appropriate MOSAICO XML tag. However, this operation, is not completely manual, this because the editor also has a parser that, when a document is imported, tries to understand the relevant parts of the documents and tries to pre-mark up this parts.
- **Enriching documents stored in the XML repository.** Of course it is possible to save in the XML repository a document that has not completely been marked up and open it later to complete the markup. The MOSAICO editor also has a versioning system and makes it possible to manage RDF relationships. A group of users can cooperatively create a document, so when one of these users tries to open one of these documents, he or she can view the document's latest version, in which all recent changes are highlighted. It is also possible to open a specific version of the document.
- **Saving a document.** A user can save a document at any time, when the markup is complete or when it is incomplete. If the user thinks the mark up is complete and tries to save it in the XML repository, the editor validates it to see if it belongs to

the MOSAIO XML schema. If the document is valid, the saving action is performed and the document is instantly made accessible in the digital library. If the document is incomplete, or if it is partially and voluntarily marked up, the editor returns to the user all the problems found in the validating operation so that the user can correct them until the markup is complete and valid.

- **Exporting a document.** All the documents stored in the XML repository can be exported in the most common third-party formats. When this operation is requested, the client-side component asks the server-side API to translate the document.

6   Conclusion

The MOSAICO project outstrips the current state of the art in digital library hosting manuscripts. Which is to say that the MOSAICO environment seeks to support scholars of mediaeval legal history in creating multimedia and hypertext content, thus preserving and even enriching the digital manuscripts heritage and the connected material. The idea is to not impose any rigid template to the authors but to provide a flexible environments using XML and RDF models that effectively manage the metadata, the semantic parts of the text, and the relationships among digital resources. In this way we make for new ways of using medieval legal history materials (e-books, critical editions) and can create new technical tools (comparison tables) for supporting research in this domain.

References

1. Agosti, M., Mariani Canova, G., Orio, N., Ponchia, C.: A Case Study for the Development of Methods to Improve User Engagement with Digital Cultural Heritage Collections. In: Grana, C., Cucchiara, R. (eds.) MM4CH 2011. CCIS, vol. 247, pp. 166–175. Springer, Heidelberg (2012)
2. Baechler, M., Ingold, R.: Medieval Manuscript Layout Model. In: Procs. of ACM Symposium on Document Engineering, Manchester, UK, pp. 275–278 (September 2010)
3. Enrich Final Conference Proceedings, National Library of Spain, Madrid, November 5-6 (2009),
   `http://enrich.manuscriptorium.com/files/enrich/`
   `ENRICH_WP8_D8_5_Proceedings_Web.pdf`
4. Sanderson, R., Albritton, B., Schwemmer, R., Van de Sompel, H.: SharedCanvas: a collaborative model for medieval manuscript layout dissemination. In: JCDL 2011, pp. 175–184 (2011)
5. Second progress report on the digitization and online accessibility of cultural material and on digital preservation in the European Union (November 2010),
   `http://ec.europa.eu/information_society/activities/`
   `digital_libraries/doc/recommendation/`
   `reports_2010/2010%20Digitisation%20report%20overall.pdf`
6. TEI Consortium (eds.): TEI P5: Guidelines for Electronic Text Encoding and Interchange. TEI Consortium (November 2007), `http://www.tei-c.org/Guidelines/P5/` (accessed: January 31, 2012)