

Vladimir Filipović

Faculty of Mathematics, Belgrade

Mladen Zagarčanin

Bar County Museum, Bar

Dušan Tošić

Faculty of Mathematics, Belgrade

Sanja Stanišić

Municipality of Bar, Bar

DIGITALIZATION IN THE BAR COUNTY MUSEUM – PILOT PROJECT

Abstract. This paper describes a pilot project of digitalization in Bar County Museum. The Museum's digital resources are stored, managed and catalogued by web-based open source software for digital asset management and hosted on the University of Belgrade - Faculty of Mathematics' web server. The characteristics and trade-offs of different types of open source software related to digital asset management are analyzed, and the FocusOPEN Digital Asset Manager is proposed as the best choice in this case. In addition, the experiences in project planning and implementation are described.

Keywords. Digitalization, Museum of Bar, Museum Archives, Digital asset management, Open source software.

1. Introduction and related work

Every museum creates and maintains a catalogue of its own assets. This also applies to museum's digital resources. Nowadays, there are many ongoing initiatives aimed at digitalizing national heritage [5], [9], as well as museum collections [1], [4], [6].

2. Bar County Museum

Bar County Museum was established in 1959. It is housed in the King Nikola's Palace, i.e. the summer house of King Nikola of Montenegro, built in 1885. The Museum is situated on the seafront in the part of town called Topolica.

Since 1976 it has been operating within the Bar Cultural Centre. The Museum itself is an institution of complex organizational structure with several departments: archaeology, history, history of arts, ethnology and natural history department, and conservation laboratory, professional library and exhibition hall.

Bar County Museum holdings include several collections, the oldest ones being kept within the Archaeology Department. These collections contain finds from pre-history, ancient times and middle ages.

The History Department collects, preserve and exhibits items related to the political, cultural and military history of the town. It holds valuable pieces of craft weapons, decorations, archive documents and photographs, and a considerable collection from the People's Liberation War.

The Ethnological Department holds several collections of rich and diverse tangible and intangible heritage related to the local population. The ethnographic material is classified

according to the specific features into collections of national costumes, household items, ethnic jewelry, culture and customs heritage and economic heritage.

The History of Art Department holds collections of paintings, drawings and graphics of mostly local authors. In addition, there is a small collection of icons and a collection of items of applied art, mostly authentic objects from King Nikola's Palace.

The Natural History Department collects and preserves paleontology and a mineralogy-petrography items within several collections..

The current permanent display in the Museum was installed in 1999 and designed in such a manner to present chronologically the continuing development and specific features of tangible and intangible heritage of Bar. The displayed artifacts represent diverse periods: pre-history, ancient times, Middle Ages, the period of Turkish dominance and the reign of King Nikola.

Over the years, the Museum has established fruitful cooperation with numerous institutions of a kind, as well as with several established international universities. The cooperation with universities includes mainly joint archeological researches of rich cultural and historical heritage of Bar area. Currently the archeologists of Bar County Museum are cooperating with their colleagues from the University of Venice – Ca' Foscari (Italy) in researching the site of the Old Town of Bar, one of the greatest archeological sites located within walls in the world. The cooperation with experts from the University of Southampton (Great Britain) is aimed at exploring the underwater archeological heritage. This research is the first underwater archeology research in Montenegro.

At the time being, the Museum's collections are being catalogued in the old-fashion, paper way, as it is described on the picture that follows.

Назив установе			
КАЛЕНДАР СА ЗАБИЈОРШКАМА	Број књиге улаза	Инв. број 983	Збирка НОБ
Предмет — назив		Ранији инв. бр.	Бр. пред. у збирци
Број фототеке	Гдје се предмет налази		
О П Ш Т И П О Д А Ц И:			
1. Опис предмета		8. Димензија — тежина	
2. Аутор		9. Стање предмета	
3. Мјесто постанка		10. Конзер. препарат, радови	
4. Вријеме		11. Шта треба предузети	
5. Историја предмета		12. Изложбе	
6. Начин набавке		13. Литература	
7. Материјал и техника			
1.	КЊИЖИЦА МАЛОГ ОБОЈАНОГ КОЈА ЈЕ ПРЕСТАВЉАЛА КАЛЕНДАР ЗА 1904. ГОДИНУ. ОСИМ КАЛЕНДАРА ТУ СЕ НАЛАЗИ И МАЛИ ПОДСЕТНИК И РОСЛОВНИК. У КАЛЕНДАРУ СУ И КАРТЕ ЈУГОСЛАВИЈЕ И СРБИЈЕ - ЦРВЕНА БОЈА.		
2.			
3.	ЈУГОСЛАВИЈА		
4.	1904.		
5.	КАЛЕНДАР ЈЕ ПРИПАДАО ДР МИЛУ БОШИЊОВИЋУ.		

The primary goal of this pilot project is to find out the optimal way of digitalizing and cataloguing museum resources, using modern information technology.

3. Digital asset management (DAM)

Digital asset management (DAM) consists of management tasks and decisions surrounding the ingestion, annotation, cataloguing, storage, retrieval and distribution of digital assets [11].

Digital asset management systems (DAMS) include computer software and hardware systems which support the process of digital asset management.

The term "digital asset management" (DAM) also refers to the protocol for downloading, renaming, backing up, rating, grouping, archiving, optimizing, maintaining, thinning, and exporting files.

Generally, the "**asset**" being managed is collected and stored in a digital format. There is usually a target version - referred to as "essence" - generally the highest-resolution and highest-fidelity representation. The asset is detailed by its **metadata**. Metadata is the description of the asset and the description depth can vary depending on the needs of the system, designer, or user.

Metadata can include, but is not limited to, the description of: asset content (what is in the package?); the means of encoding/decoding (e.g. JPEG, tar, MPEG 2); provenance (history to point of capture); ownership; rights of access; as well as many others. There exist some pre-defined standards and templates for metadata such as Dublin Core and PBCore.

If systems contain large-size asset essences, such as MPEG 2 and JPEG2000 for the cases of images and video, there are usually related "proxy" copies of the essence. A proxy copy is a lower-resolution representation of the essence that can be used as a reference in order to reduce the overall bandwidth requirements of the DAM system infrastructure.

The following broad categories of digital asset management systems may be distinguished [11]:

- Brand asset management systems, with a focus on facilitation of content re-use within large organizations.
- Library asset management systems, with a focus on storage and retrieval of large amounts of infrequently changing media assets, for example in video or photo archiving.
- Production asset management systems focus on managing assets as they are being created for a digital media production (video game, 3D feature film, animation, visual-effects shots, etc.) They usually include work-flow and project-management features.
- Digital supply chain services, pushing digital content out to digital retailers (e.g. music, videos and games).

DAM software may be open source or proprietary.

According to [3], the heart of DAM systems should have the following set of core content management functionalities (some native, some perhaps from 3rd parties):

1. The repository
This is core of any system; it builds a representation of the content utilizing a relational database or file system, or some combination. This includes basic repository services, such as version control, categorization, upload, and download.
2. The metadata index
This includes descriptors, administrative data as well as versions, and other hierarchical, peer to peer, parent child or lineage relationships.
3. The search engine
This feature performs searches against the above-defined index and repository.
4. The access and rights subsystem
Privileges and permissions that define who can see and operate with different objects.

5. The workflow or collaboration engine
 - Scheduling and definition of tasks in serial or parallel progression.

4. Desirable characteristic for successful DAMS

Nowadays, there are many comparative analyses of different digital libraries [2] and digital asset management systems [3].

Software that catalogue and manage digital resources (DAMS) should have following features (ordered from the most important to less important ones):

1. Ability to upload and store digital resources. Software which manages digital resources should have all the facilities needed to manage many different types of media across the organization. Uploaded resources should be kept safe and without loss of quality. It should also allow automatic extraction of digital resource's properties and attaching extracted properties (metadata) to uploaded digital resource (asset).

2. Ability to add various properties to uploaded digital resource. DAMS should allow end user to catalogue uploaded digital resource e.g. to attach various metadata values to uploaded digital resource. Specific 'tags' can be added to digital resource to provide users with search controls that help them to operate the system with minimal effort and expertise

3. Ability to work on the web. Digital Asset Manager should allow end user to interact with the system (upload data, catalogue data, make inquiries) by using web as communication channel.

4. Advanced Search. Digital Asset Manager should provide users with search technology that enables them to quickly and easily locate the media files they need. Also, Digital Asset Manager should permit users to browse for, as well as search for media files by using 'tags'.

5. Powerful Media Processing Features. Digital Asset Manager should handle most common media types, including: photos, video, audio and print artwork. It should auto-generate thumbnails and previews for most supported media types including bandwidth intensive media like high definition (HD) video or high resolution artwork.

6. Easy To Use. User interfaces should be designed to be intuitive and utilize GUI models that users are already familiar with. It should also offer a comprehensive and context-sensitive help system that will guide even the most novice user through key tasks and system functions.

7. Flexible and Configurable. Media and metadata management tools should be flexible, which means that media files can be hidden or shown to certain users, can be displayed with or without a watermark (includes video) and can be restricted from download or flagged for approval before publication. Likewise, administrators should have ability to revise and add metadata fields to media files with far greater ease and control, ensuring that media is catalogued, presented and made available to users exactly as the business requires.

8. Scalable. Offered solution should have very scalable nature and the whole system have to perform well under very heavy load (e.g. hundreds of concurrent user). Also, it should perform well even in the case when demands arise very quickly. The architecture of Digital Asset Manager should be robust, and implementation should be without 'narrow throats'.

9. Possibility of automatic interchange. Offered software should have a way to interchange digital resources and/or metadata in automatic manner. In other words, the system should have possibility to communicate with other systems, or with digital resources register, like the register that is proposed in [10].

10. Ability for further changes and improvements. Digital Asset Manager will be changed and improved during years. In order to achieve that goal, source of the system should be available for further improvements and modifications.

11. Multiple brands under multiple domains in the same core system. Digital Asset Manager should be structured in such a way that each brand can have its own styling (logo, images, colors), a unique domain name (rather than a sub-domain), different metadata, terminology and home page. All digital resources and users from one brand are separated from those in other brands, so end users do not ever need to be aware that the system is shared.

12. Single sign-on. The system should provide a single sign-on services to the entire organization. It should simultaneously support access from approved external users without internal network access.

13. Streamlined Cataloguing. Administrators should be able to catalogue media easily and quickly, yet still produce the comprehensive metadata needed for fast and accurate search results. The streamlining features include: digital asset cross-referencing, metadata templates, scheduled uploading and bulk uploading.

14. Copyright and Brand Control. To better protect the copyright and reduce the risk of license breach or brand misuse the Digital Asset Manager should enable administrators to control media use - protecting against unauthorized downloads, applying watermarks or suppressing media previews and files for selected user groups on a per asset basis. Furthermore, the system should support publications and expiry features.

15. Comprehensive Statistics and Reporting. Reporting feature enables administrators to rapidly generate reports on media and system usage, such as login and registration statistics, media download statistics and user audit trails. This information assists administrators in troubleshooting and assessing the need for media or system features into the future.

5. Open source software

The paper [7] states that there are three main types of software product firms:

1. **Closed source firms** own all competitively differentiating software components their products are based on (e.g. : Microsoft Windows, Oracle's database and SAP Business Suite).
2. **Single-vendor open source firms** also own all competitively differentiating components, but they make some of these components available as open source (like: Alfresco, Jaspersoft's BI, etc.).
3. **Open source distributors** integrate a large set of open source components and distribute the assembly for free. The distributors typically don't own these components (for instance, Red Hat's Linux).

An open-source software is available in a source code form: the source code and certain other rights normally reserved for copyright holders are provided under a software license that permits users to study, change, improve and also to distribute the software [12].

An open source software is very often developed in a public, collaborative manner. It doesn't just mean access to the source code. The distribution terms of an open-source program must comply with the following criteria:

According to [7], in order to be an open source, all of the terms below must be applied together, and in all cases. For example, they must be applied to derived versions of a program as well as the original program. It's not sufficient to apply some and not others, and it's not sufficient for the terms to apply only some of the time.

1. **Free Redistribution.** The license may not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license may not require a royalty or other fee for such sale.
2. **Source Code.** The program must include source code and must allow distribution in source code as well as in a compiled form. Where some form of a product is not distributed with the source code, there must be a well-publicized means of downloading the source code, without charge, via the Internet. The source code must be the preferred form that a programmer could modify the program.
3. **Derived Works.** The license must allow modifications and derived works, requiring that derived code will be distributed under the same terms as the license of the original software.
The software has little use if you can't maintain it (fix bugs, port to new systems, make improvements), and modification is necessary for maintenance. The intent here is to allow modifications of any sort. It must be allowed for a modified work to be distributed under the same license terms as the original work.
4. **Integrity of the Author's Source Code.** The license may restrict source code from being distributed in a modified form only if it allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time.
The license must explicitly permit distribution of software built from modified source code. The license may require that derived works have a different name or version number from the original software.
5. **No Discrimination against Persons or Groups.** The license must not discriminate any person or group of persons.
6. **No Discrimination against Fields of Endeavor.** The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.
7. **Distribution of License.** The rights attached to the program must apply to all to whom the program is redistributed, without the need for execution of an additional license by those parties.
8. **License Must Not Be Specific to a Product.** The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.
This means you can't restrict a product that is identified as open source to be free only if you use it with a particular brand of Linux distribution, etc. It must remain free if you separate it from the software distribution it came with.
9. **License Must Not Contaminate Other Software.** The license must not place restrictions on other software distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

6. FocusOPEN Digital Asset Manager

In the case of digitalizing the assets of the Bar County Museum we decided to implement the pilot project with open source software for DAM, instead of building our own proprietary software.

The requirements that should have a successful DAMS, enlisted in section 4, are hard to achieve for the software build from the scratch. To be more precise, that requirements are almost impossible to achieve in adequate time frame and with resources that are available to us.

On the other way, the limitations arising from open source license don't prevent us from upgrading and changing downloaded solution. In other words, we can upgrade and change downloaded open source solution, without breaching license agreement.

During evaluation of possible candidates, one open-source product has drowned our attention – an open source version of FocusOPEN Digital Asset Manager fulfills almost all requirements enlisted in section 4.

According to the classification from the beginning of Section 5, the selected software product is created by a single-vendor open source firm, which owns all competitively differentiating components, but it makes the FocusOPEN available as open source.

FocusOPEN Digital Asset Manager is a digital asset management solution designed to help businesses make more of their media, from managing brands to distributing marketing collateral or other media assets.

FocusOPEN is based on Windows server technology. The main requirements are:

- Windows Server 2003 or 2008
- IIS 6,7 or 7.5
- SQL Server 2005 or 2008
- ASP.NET 3.5

It is possible to run FocusOPEN on PWS or IIS7 on Vista/Windows 7 and SQL Server Express 2005 or 2008.

FocusOPEN is delivered using a multi-tier architecture that can be distributed across one or more hosts:

- Web Application: The core ASP.NET Digital Asset Management system where most of the business logic is contained
- Database: SQL Server RDBMS
- File system: Assets are stored in the file system so any accessible file storage device can be used
- Media Processing: A dedicated Asset Processing Server (APS) is provided as a Windows service

FocusOPEN supports server farms, clustering, redundancy, and other High Availability technology. For storing digital assets, FocusOPEN supports multiple storage points and it is possible to store assets in multiple locations simultaneously without consolidating them.

The APS (Asset Processing Server) is a Windows service and uses WCF (Windows Communication Framework) to exchange messages with the FocusOPEN web application via a designated TCP port. It is a self-contained sub-system and can also be used independently with other WCF compliant applications.

FocusOPEN has a number of unmodified free and open source dependencies that are used for media processing and extracting metadata:

- ImageMagick
- FFmpeg

- FLVTool2
- LAME
- GhostScript
- EXIFTool
- TagLib#

FocusOPEN uses the following free and open source client-side components:

- FlowPlayer (video)
- MP3 Player (audio)

FocusOPEN is an ASP.NET 3.5 web application authored in C#. All editions include copies of the source code.

Catalogue Assets: Step Two [ADMIN] - Windows Internet Explorer

http://liss.matf.bg.ac.rs/DigitalCatalog/Admin/Assets/AssetForm.aspx?assetid=18

File Edit View Favorites Tools Help

Catalogue Assets: Step Two [ADMIN]

FocusOPEN

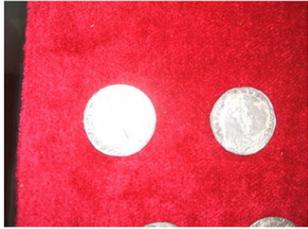
DIGITAL ASSET M

« Catalogue | Admin Home | Korisnici | Porudžbine | Resursi | Sadržaj | Alati | Izveštaji | [Sanja Stanišić](#) | [Logout](#)

Admin > Manage Assets > Catalogue Assets: Step Two

CATALOGUE ASSETS: STEP TWO

Complete the following information for this asset.
Note: fields marked * are required. Roll over field titles for a explanation of each. | [help](#)



[download original](#) | [replace asset file](#) | [change preview settings](#) | [delete asset](#) | [audit log](#)

Asset Reference: 18
Datum umetanja: 17 September 2011
Dimensions (pixels): 1728 (h) x 2304 (w)
[View asset file metadata \[+\]](#)

Ilaziv ustanove: * Zavicajni Muzej Bar

Predmet - naziv: * DSCF0781

Inventarski broj: *

Panji invent. broj: *

Broj knjige ulaza: *

Datum poslednjeg inventarisanja: *

Inventarisanje izvrsio: Not Specified

Odeljenje: * Not Specified

Zbirka: * Not Specified

Broj fototeke: *

Gde se predmet nalazi: Not Specified

Opis predmeta: *

Autor: *

Mesto postanka: *

Vrijeme: *

Istorija predmeta: *

Porijeklo: Not Specified

Iacin nabavke: *

Materijal i tehnika: *

Internet | Protected

7. Outline of the pilot project and gained results

The pilot project is planned and implemented according to recommendations discussed in paper [8]. The Faculty of Mathematics' part of the team has as a task the adaptation and localization of the open source software, and part of the team from Bar is supposed to upload real data from museum collections into digital asset management system.

People from Faculty of Mathematics are to configure DAMS based on the FocusOPEN open source software and set it on the Faculty's web server. In that way, this part of the team enables uploading and cataloguing Bar County Museum digital resources.

Due to its open-source nature, the solution implemented in this pilot project can be easily made interoperable with Cultural heritage register, similar to register described in [10].

8. Conclusion and future work

The pilot project shows that open source software for DAM FocusOPEN can be successfully applied in Bar County Museum digital resource management.

We configured DAMS based on the FocusOPEN open source software, set it on web server at Faculty of Mathematics and uploaded and catalogued some of Bar County Museum digital resources. After the successful completion of this pilot project, we are going to upgrade and localize FocusOPEN open source system and translate all user interfaces into Serbian.

In order to do so and not breach the license agreement, we are going to set a repository for modified source code and make that repository accessible from every web page of modified solution. Assets that are already uploaded into DAMS will be preserved – changes of the software will not influence previously uploaded digital assets.

References

- [1] Branković T., Šega M., *Digitalization of Collections in the National Museum of Požarevac*, Review of the National Center for Digitization 16, pp. 55-58, 2010
- [2] Dobрева M, Feliciati P., *User-centric Evaluation of Digital Libraries: Three Case Studies*, Review of the National Center for Digitization 18, pp. 1-15, 2011
- [3] Magan A., *Intro to Digital Asset Management: Just what is a DAM?*
<http://www.realstorygroup.com/Feature/124-DAM-vs.-DM>, 2005
- [4] Mijajlović M., Stojanović A., *Digitalization process in Belgrade City Museum*, Review of the National Center for Digitization 17, pp. 50-56, 2010
- [5] Mijajlović Ž., Pejović N., Ognjanović Z., *Doctoral Dissertations in NCD Virtual Library*, Review of the National Center for Digitization 18, pp. 15-18, 2011
- [6] Mirčić A., Jeremić N., *Project of Digitization of Art Documentation Department Museum of Contemporary Art, Belgrade*, Review of the National Center for Digitization 15, pp. 77-80, 2009
- [7] Perens B., *Open Sources: Voices from the Open Source Revolution*, ISBN: 1-56592-582-3,
<http://oreilly.com/catalog/opensources/book/perens.html>, 1999
- [8] Riehle D., *Control and Steering Open Source Projects*, IEEE Computer, July 2011, pp. 93-96, 2011
- [9] Zorić I., *Processing of Documents and Digitalization of Photographs from Nikola Tesla's Personal Fund*, Review of the National Center for Digitization 15, pp. 41-46, 2009

- [10] Životić S., Malkov S., Mitić N, Mijajlović Ž., *A Cultural Heritage Register Prototype*, Review of the National Center for Digitization 17, pp. 1-8, 2010
- [11] http://en.wikipedia.org/wiki/Digital_asset_management, 2011
- [12] http://en.wikipedia.org/wiki/Open_source_software, 2011