

Projet ANR- 11-IS02-001

MEX-CULTURE/ Multimedia libraries indexing for the preservation and dissemination of the Mexican Culture

Deliverable

Final report on multimedia platform

Programme Blanc International II- 2011 Edition

A	IDENTIFICATION	2
B	INTRODUCTION	3
C	MEX-CULTURE PLATFORM	3
	C.1.1 Java Enterprise Edition	4
	C.1.2 Apache Server	4
	C.1.3 Liferay Portal	4
D	DESIGN OF THE PORTAL	4
	D.1 Gallery section	5
	D.2 Search by category	5
	D.3 Video based search	6
	D.4 Image based search	9
	D.5 Search by texte	12
	D.6 Display video selection	13
	D.7 Summary navigation of a selected video	15
	D.8 About us	18
E	CONCLUSIONS	18
F	REFERENCES	18

A IDENTIFICATION

Project acronym	MEX-CULTURE
Project title	Multimedia libraries indexing for the preservation and dissemination of the Mexican Culture
Coordinator of the French part of the project (company/organization)	Centre d'Etude et de Recherche en Informatique et Communications – Conservatoire National des Arts et Métiers
Coordinator of the Mexican part of the project (company/organization)	Instituto Politécnico Nacional- CITEDI
Project coordinator (if applicable)	Michel Crucianu : France Mireya Saraí García-Vázquez : México
Project start date	01/01/2012*
Project end date	30/04/2016
Competitiveness cluster labels and contacts (cluster, name and e-mail of contact)	Cap Digital Paris-Région Philippe Roy Philippe.Roy@capdigital.com
Project website if applicable	http://mexculture.cnam.fr

* The Mexican partners are only financed since November 2012.

<i>Coordinator of this report</i>	
<i>Title, first name, surname</i>	<i>Alejandro Ramírez Acosta</i>
<i>Telephone</i>	
<i>E-mail</i>	<i>ramacos10@gmail.com</i>
<i>Date of writing</i>	<i>25/02/2016</i>

Redactors :	Alejandro Ramírez-Acosta (IPN) Mireya Saraí García-Vázquez (IPN)
--------------------	---

B INTRODUCTION

The scalable methods for indexing, summarization and retrieval in large cultural archives of Mex-Culture project are implemented in a common development platform. This platform has an open architecture based on web services in order to facilitate the independent development of complementary content description and retrieval components.

As it mentioned in the deliverable ID4.4, the multimedia infrastructure is based on a Service Oriented Architecture (SOA) [B.1, B.2]. It is a component model which interrelates the different functional units of a solution, named “services”, through well-defined interfaces between those services. The interfaces are defined in a neutral manner, independent of the hardware platform, operating system or programming language in which the service is implemented. A service is thus a function that is self-contained and immune to the context or state of other services.

C MEX-CULTURE PLATFORM

Mex-Culture as multimedia infrastructure, is based on SOA and the standards of Web Services and the Semantic Web. The Mex-Culture architecture is based in Web Service specifically designed for multimedia processing components. The figure C.1.1 shows the model that integrates the approaches outlined in [C.1, C.2, B.2] for SOA architecture. It can clearly see an independence between the layers, as mentioned in the SOA philosophy. Therefore, Mex-Culture includes five layers allowing an efficient implementation and operation:

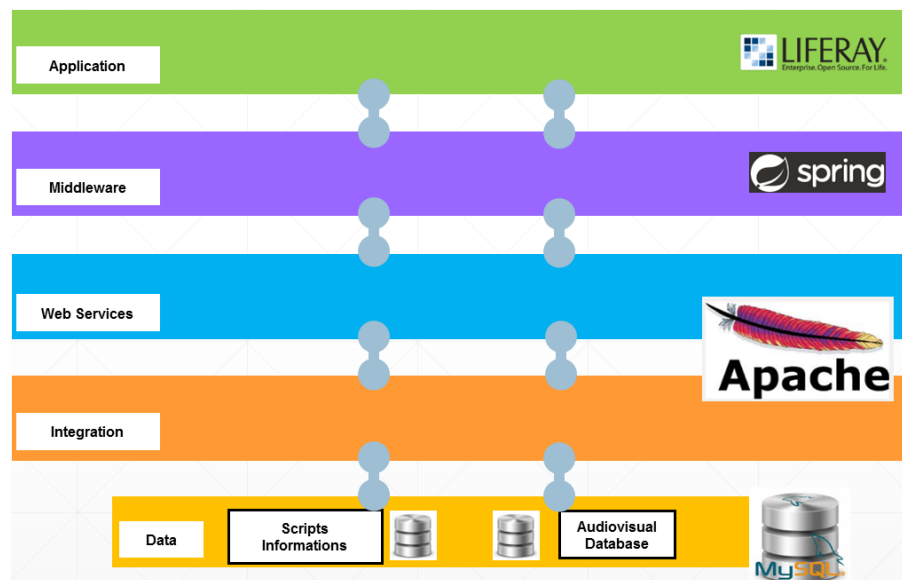


Figure C.1.1: Mex-Culture platform based on SOA.

- **Data** : This part is composed of cross-media information and information based on metadata.
- **Integration** : Base component, this part facilitates the integration and interaction of components to do the base processing on unstructured all types data (image, video, speech,

audio), the indexing, the summarization and the retrieval of multimedia information of the database of Mex-Culture project. Also it allows the interaction with Web Services.

- **Web Services:** They are services of processing and human-machine interface. Each web service performs a particular processing that can be inserted into a chain and associate it with other web services to meet a specific need.
- **Middleware :** Communications bus, provides communication between different applications, which are not necessarily made to operate together. It is the lightly vertebral column and pervasive of integration through which software services and application components circulating.
- **Application :** This enable the user to access services in an hidden way through a simple web browser: the activation of processing chain is then called. Here, it ensures displaying the GUI. The client that will be used is based on a thin client web portal.

Based on the criteria in [B.1, B.2], the following tools have been selected to cover the main parts of the open SOA platform.

C.1.1 JAVA ENTERPRISE EDITION

Java Platform, Enterprise Edition, or Java EE [C.3], is the de facto standard for developing server-side Java applications.

C.1.2 APACHE SERVER

Apache is a web server [C.4], produced by Apache Software Foundation [C.5]. A web server is capable of interpreting the HTTP requests arriving at the port associated with the HTTP protocol, and provide a response with the same protocol.

C.1.3 LIFERAY PORTAL

The central idea of SOA is the concept of reusable services that can easily recombine to create business processes. A user interface with these characteristics is what is called a portal. The benefits are greater satisfaction of users of the systems and more efficient access to information.

A portal is a Web-based application that acts as a gateway between users and a range of different high-level services. A portlet is a web component that processes requests and generates dynamic content. The content generated by a portlet is also called a fragment (e.g. HTML, XHTML, WML) and can be aggregated with other fragments to form a complete document.

Liferay Portal [C.6] is a free and open-source enterprise portal solution using the latest in Java and Web 2.0 technologies and distributed under the GNU LGPL. Liferay is Portlet technology which follows the JSR 168 and JSR 286 Compliant [C.7, C.8].

D DESIGN OF THE PORTAL

The design of the portal is guided by the goal of providing advanced solutions for indexing, summarization, searching and accessing large scale digital audiovisual content in the context

of Mexican Culture. So the portal must be relevant to user needs. In the following the portal sections are described.

D.1 GALLERY SECTION

The initial section of the portal Mex-Culture project is shown in the figure D.1.1.

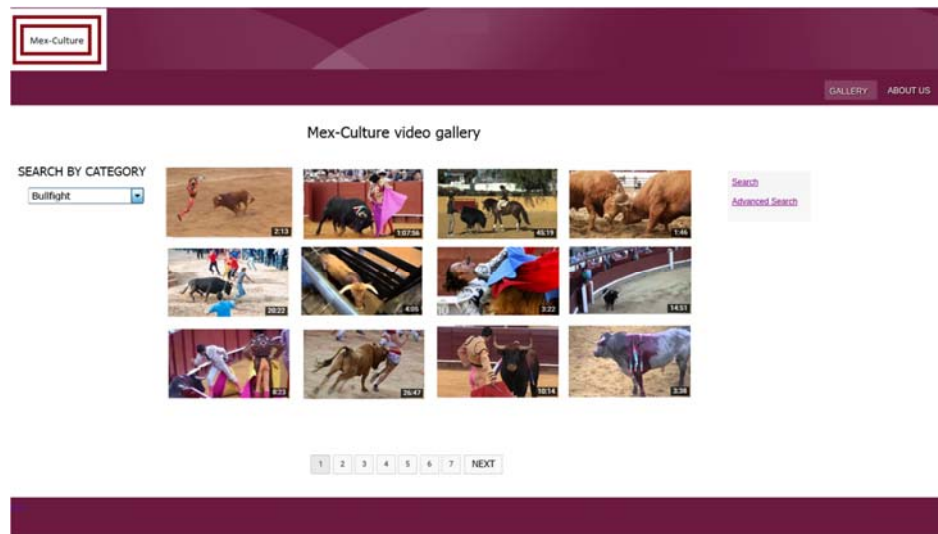


Figure D.1.1: Gallery section.

In the Gallery section the user can see the results of the queries or do other searches. The figure D.1.1 shows the element displayed:

- Search by category,
- Search and advanced search,
- Video grid,
- Next or back buttons.

D.2 SEARCH BY CATEGORY

To realize a *search by category*, the user must choose one of the options in the Category's combo box. There are seven categories in the Mex-Culture portal. Once the user selects the option, the grid that contains the videos will change, showing the new images that represent the videos that belong to the category that was chosen by the user, as shown in the figure D.2.1. If the user wants to see more results for the same category he only needs to click Next. When the user selects a video by clicking an image as shown in the figure D.2.1, the user will be redirected to the *display video selection* (section D.6).

Technically speaking, the flow of all functions begins when a user selects one of the options in the category's combo box, see figure D.2.1. There is an event detector that activates when the page is loaded or refreshed. The search by category module (portlet) is in the application layer of the SOA architecture (see figure D.2.2). This module sends the information to a class object in the middleware layer. This class is in communication with the web service object. The web services call a component in the integration layer. This component is the tool used to do queries in the data layer. All the addresses of the videos

and images that belong to a category are saved in component arrays. The information in this component is read by a middleware object. Thanks to this object the videos for the category can be displayed on the application module, data grid module, see figure D.2.1.

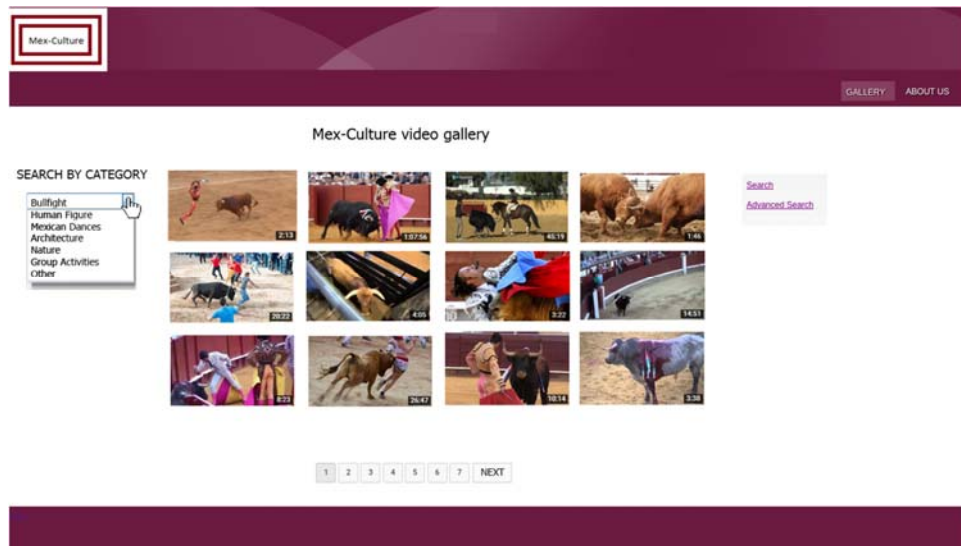


Figure D.2.1: Gallery section when selected search by category.

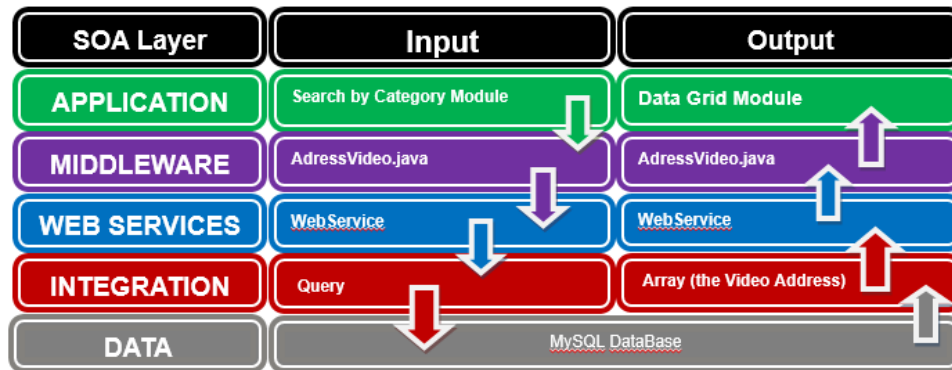


Figure D.2.2: SOA architecture for search by category.

D.3 VIDEO BASED SEARCH

This section describes the *video based search* on the Mex-Culture portal and explain how works with SOA architecture from the application to the databases.

Video based search is one of the function on the Mex-Culture portal. This function begins in the application layer of the SOA architecture interacting with the users Mex-Culture portal (see figure D.3.1). It uses blocks of html controllers called portlets. These portlets on SOA are the application modules. In this project the application module is the portlet of search.

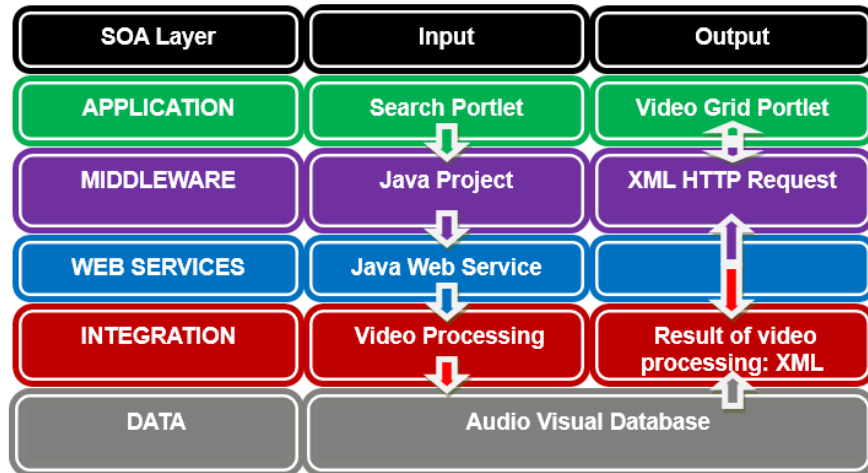


Figure D.3.1: SOA architecture for *video based search*.

The flow begins when a user selects the option *video based search* on the *search* or *advanced search* menu (figure D.2.1 and see figure D.3.2.).

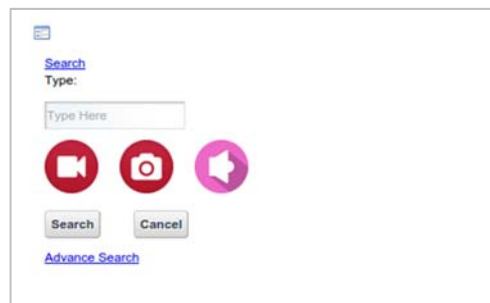


Figure D.3.2: *Search* or *advanced search* option.

The portlet open a file input to select a video file to search based on that video, figure D.3.3.

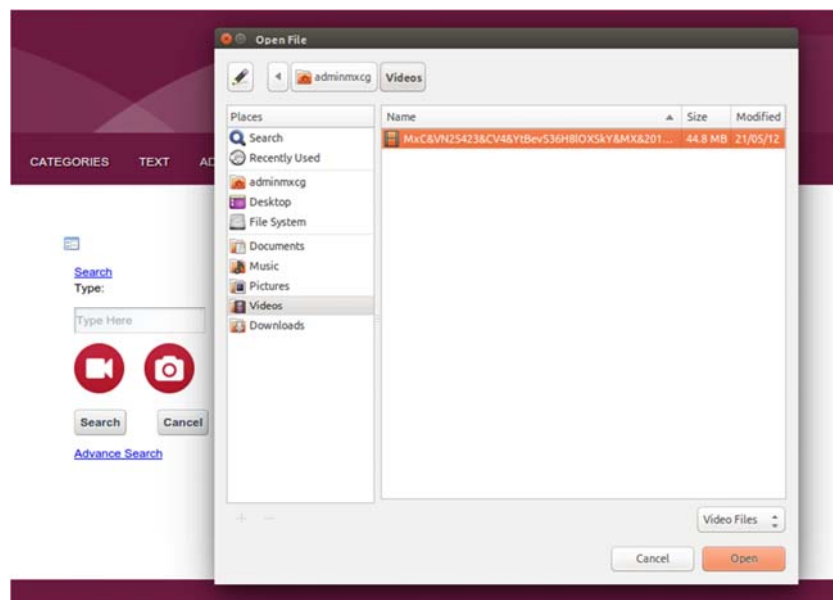


Figure D.3.3: File Input to select a video to search.

If the selected file is not a video, the portlet display an alert with the message “allowed extensions are: mp4”, figure D.3.4.



Figure D.3.4: Alert for video file input exception.

If the selected file is a video, the application plays that video and display a button (Search button) to begin the *video based search*, figure D.3.5, D.3.6.

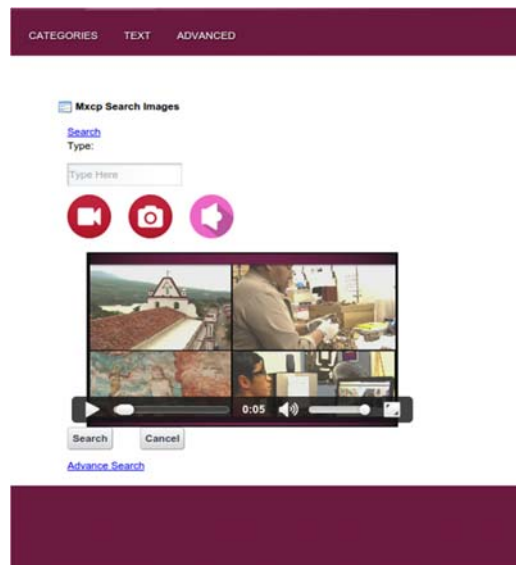


Figure D.3.5: Graphical display of the selected video file.

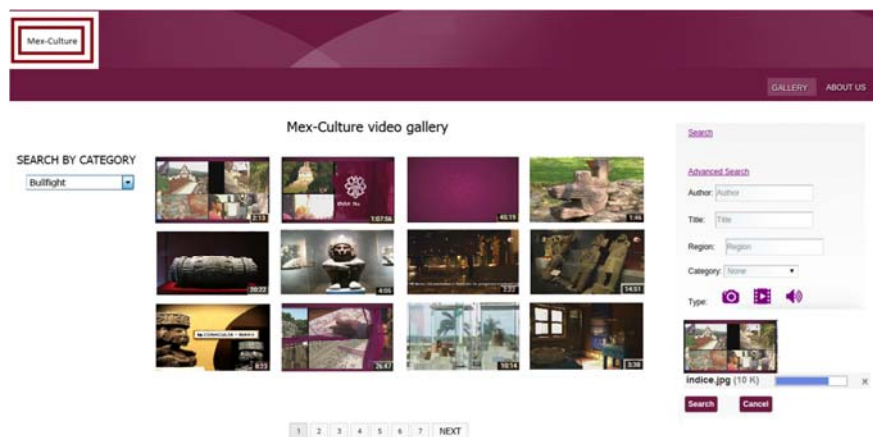


Figure D.3.6: Gallery section. User uploading a file to use as a parameter in the video search.

If the user clicks on the *search* button, the search portlet with the input file make a post operation on the middleware object. To process the video, the middleware object calls the web service. The web service send the information to the component video processing in the SOA integration layer. The component video processing uses the data layer to compare the video with all videos on the audiovisual database.

After the video was processed, the component video processing send the next information, see figure D.3.1:

- file information with the most similar videos in the Mex-Culture video database,
- file information with the main information of most similar sections in each video with respect to video query.

The file information with the similar videos is read by a middleware object. Thanks to this middleware object, similar videos can be displayed on the application module (Video Grid Portlet), figure D.3.7.

Mex-Culture video gallery

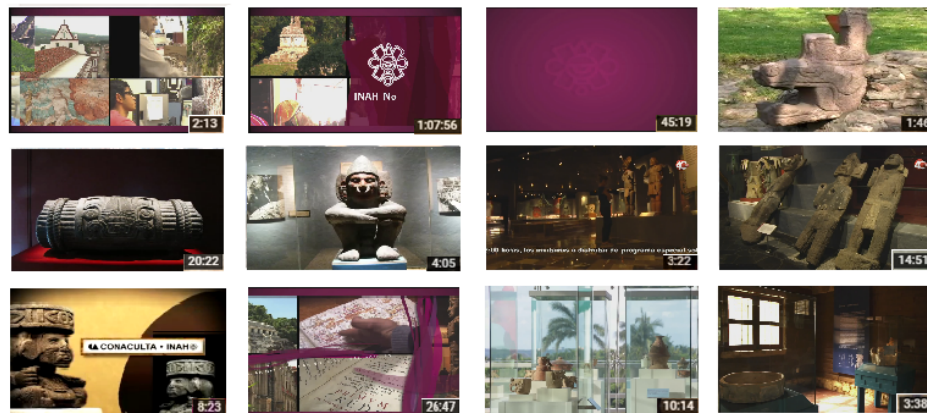


Figure D.3.7: Video Grid Portlet (display of all the similar videos).

When the user selects a video by clicking an image in the figure D.3.7, the user will be redirect to the *display video selection* (section D.6).

D.4 IMAGE BASED SEARCH

This section describes the *image based search* on the Mex-Culture portal and explain how works with SOA architecture, from the application to the databases.

Based in the figure D.4.1, the search image-portlet as application object is used. In this portlet, the user can upload the image file that is needed to process, figure D.4.2. Then, with this application object, the image is sent to the middleware layer that functions like a bridge between the application object (image search portlet) and the web service object that manipulates the information file.

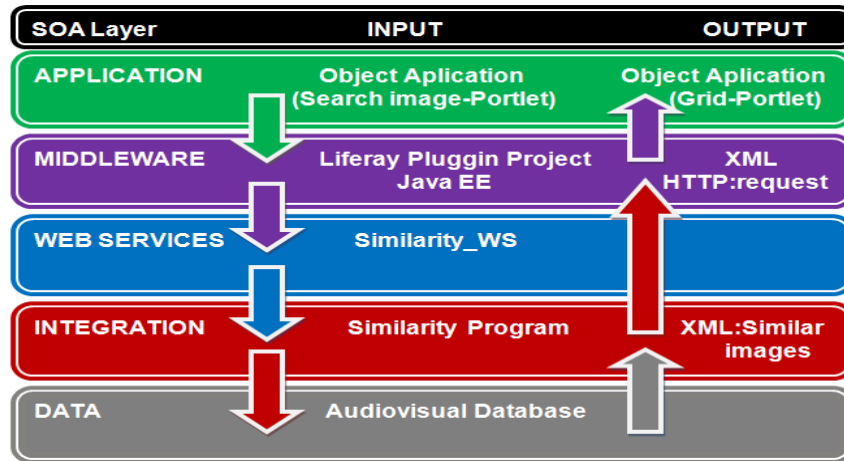


Figure D.4.1: SOA architecture for image based search.

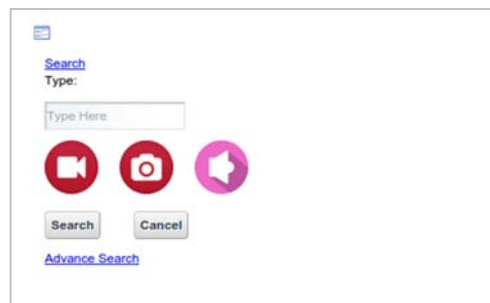


Figure D.4.2: Search or advanced search option.

The user clicks in the open button to add a image file to search. The image file is selected from the user's computer and then uploaded to the application object, figure D.4.3.

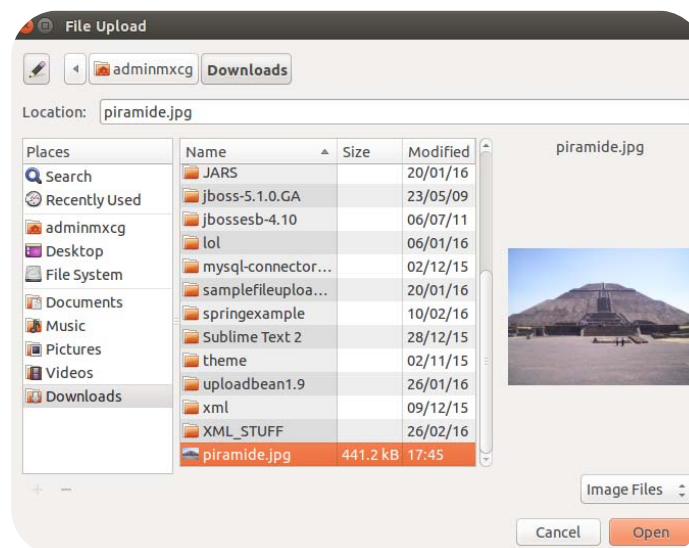


Figure D.4.3: File Input to select an image to search.

Once the image is properly selected, the user can click on the “search” button. When the user clicks the *search* button, it pass to the middleware level, in this case to he “liferay plugin project”. In order to process the image, the middleware sends the information to the web service that is in charge of the extraction and delivery the information to the component image processing called similarity program in the SOA integration layer.

The component similarity program uses the data layer to compare the image with all images on the audiovisual database.

After the image was processed, the component similarity program send the next information, see figure D.4.1:

- file information with the most similar images in the Mex-Culture audiovisual database.
- file information with the main information of most similar images in each video with respect to image query.

The file information that was obtained previously, is readed by the middleware object. This object assigns the resources and shows them in the grid of the application object. In this part, the result is visible for the user (figure D.4.4). Unless the user wants to make a new search, in that case the process is repeated.

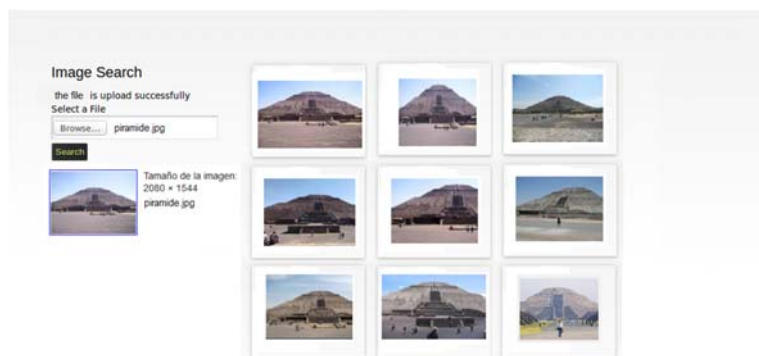


Figure D.4.4: Result from the image search.

If the user clicks in one image of the grid, the image is show in a new page/tab, figure D.4.5.



Figure D.4.5: View of individual image.

D.5 SEARCH BY TEXTE

This section describe the *search by texte* on the Mex-Culture portal and explain how works with SOA architecture, from the application to the databases on the Mex-Culture project.

The function *search by texte* begins in the application layer of the SOA architecture interacting with the users Mex-Culture portal using blocks called portlets. These portlets on SOA are the application objects. In this project, the application object is the portlet of search, see figure D.5.1.

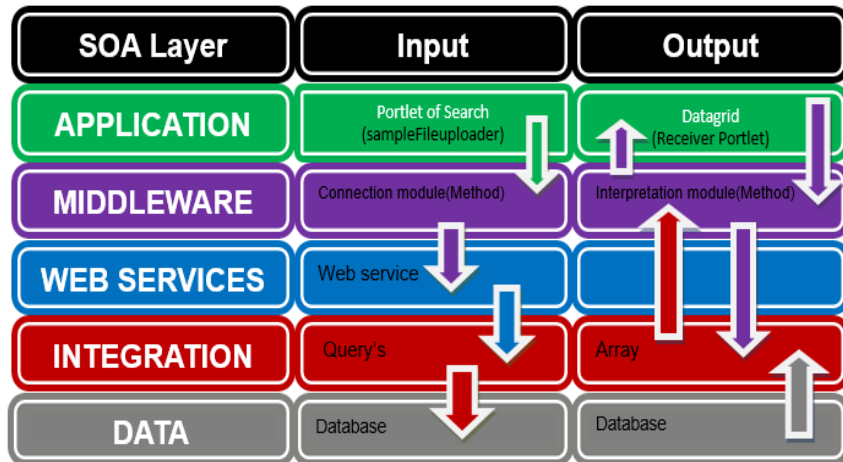


Figure D.5.1: SOA architecture for video based search.

To relize video queries through text, it is necessary to go to application layer in the portlet search and select the field "search"(see figure D.5.2). When this option is select, it can show an input text in application portlet to make a search via text. Once writing something into input text, it is necessary to click button *search*. Then this portlet send the information between the middleware object and the web service. This web service calls the component queries in the integration layer. This component generates a set of instructions to search information requested. Once integrated the information, the component generates a file information. The file information with the all videos found is read by a middleware object. Thanks to this object the videos found can be displayed on the application object (datagrid portlet). It can select see more videos, using "back" and "next" button in the portlet, this process is the same for each search.

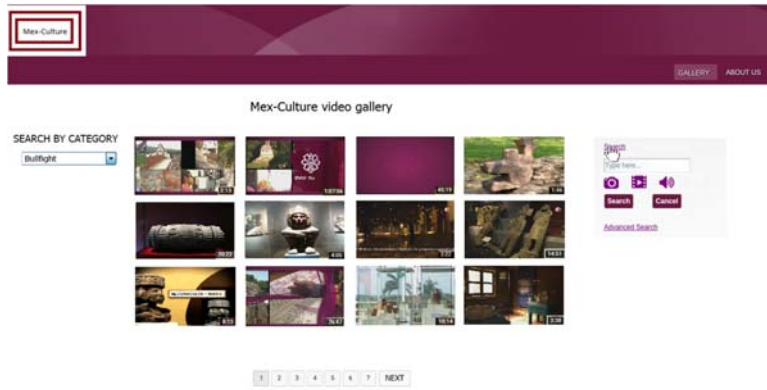


Figure D.5.2: Gallery section when the user using basic search.

The figure D.5.3 shows the advance search option. In this case, there are some specific fields to be completed by text. The search procedure is similar to that discussed above.

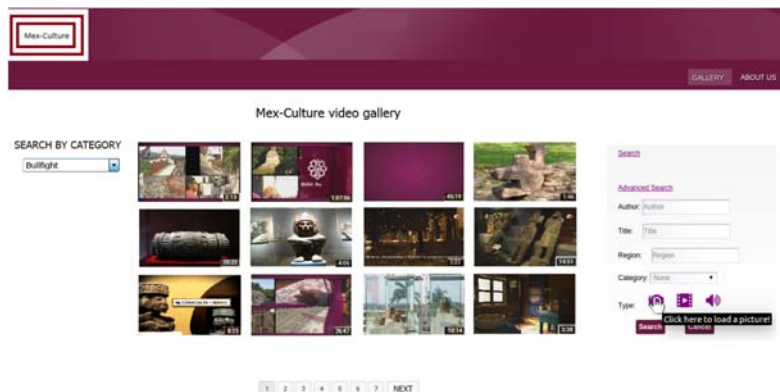


Figure D.5.3: Gallery section when the user uses advance search.

When the user selects a video by clicking an image in the figure D.5.3, the user will be redirect to the display video selection section D.6.

D.6 DISPLAY VIDEO SELECTION

This section describe the *display video selection* on the Mex-Culture portal its description and its relation with SOA architecture, figure D.6.1.

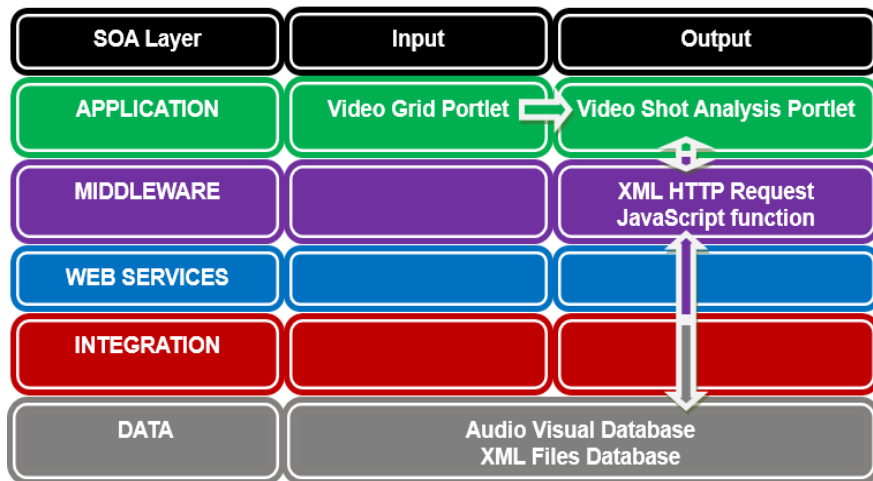


Figure D.6.1: SOA architecture for display video selection.

To interact with the users Mex-Culture portal uses blocks of html controllers called portlets. These portlets on SOA are the application modules. In this function the application module is the video grid portlet (see figure D.2.1) and the video shot analysis portlet. When a user clicks a video the application module (video grid portlet) send the information to the video shots analysis portlet. On the video shots analysis, a page will be displayed, see figure D.6.2:

The video:

- To play the video in the application, a middleware object calls a video from the data layer (audiovisual database Mex-Culture) and display that video on the application layer (video shot analysis portlet).

The main information of the video:

- To display the information of the video, a middleware object read the information from the data layer. In the data layer exists an information file for each video description.

A table of times for each shot:

- To generate the table of shots, a middleware object read the information from the data layer. This information contains the time of begin and end for each shots and the information of the keyframes of each shot.

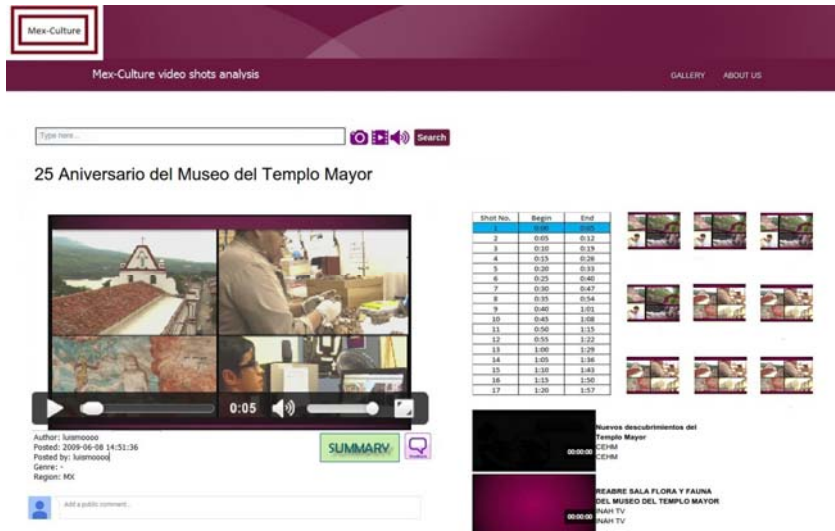


Figure D.6.2: Description and information shots of the video selected.

If the user selects a shot from the table, a middleware object read an information from the data layer to know the informations of the keyframes and return the informations to the middleware object to display the keyframes on the application module. the keyframes are from the in and out of the video segment, see figures D.6.2, D.6.3.



Figure D.6.3: Display of the keyframes for each video segment.

D.7 SUMMARY NAVIGATION OF A SELECTED VIDEO

This section describes the summary navigation of a selected video on the Mex-Culture portal and explain how works with SOA architecture, from the application to the databases on the Mex-Culture project.

Video summary offers a possibility to the user to navigate in an audio-visual material by refining its navigation. Different levels of granularity of summary are possible and this is in subspaces description. The partitions of the same audio-visual materials are used, but they are performed with different descriptors. This is where we distinguish this portal from others. In this portal we introduce some applications that have yet not been seen in other web pages.

The summary section is responsible of displaying a video and its corresponding depth levels. What is a depth level? To acquire depth levels, a video is divided into several parts. This has the objective of displaying the different sections that a video could have, for example: a video may be 30 minutes' long, let us say that this video has an interview somewhere in it, but we do not exactly know its position in the video. We have one choice; we could watch the entire video, or skip forward in it until we find the specific scene that we are looking for. Any of those choices are not optimal, to save time; one of the main features of the Mex-Culture project is to present to the user the option of a section that has a slider with all the video shots of a specific video. The goal of this, is to avoid what we described on the paragraph above. The user can then, quickly browse the shots of the video to look for a specific scene in any video.

To navigate along the video summary, it must choose the video with the button: summary. See figure D.7.1.

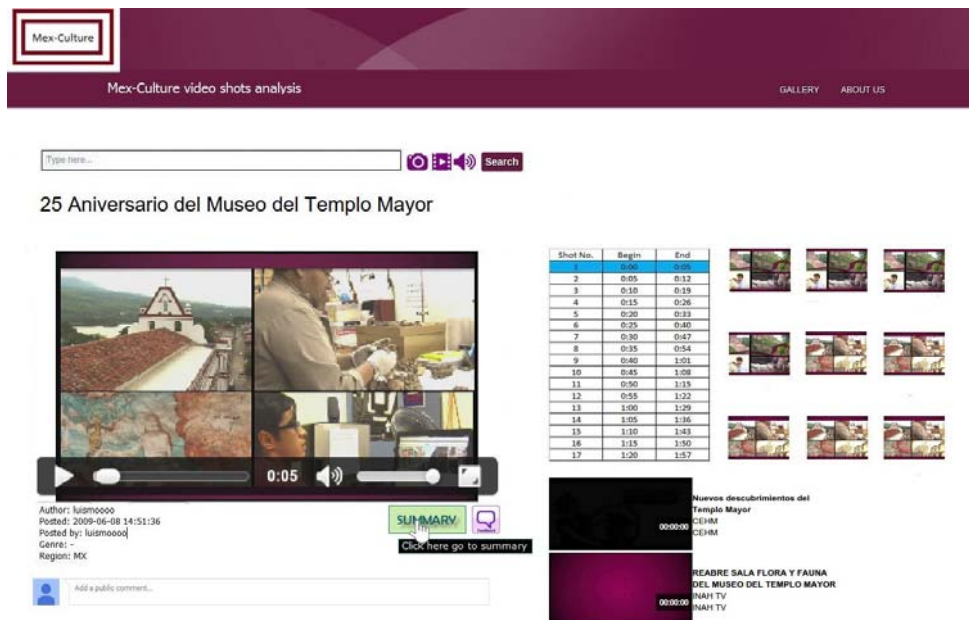


Figure D.7.1: Button summary in the page “Display video selection”.

To make the summary page work, we need to know first know which video we will be going to put in the player. When we have this, we are going to search in the database the shots of the corresponding video. We have four levels of depth available for each video. Each video segment can have up to four depth levels, so we will deploy four sliders each one of them will have the corresponding shots of the corresponding level, if there are no shots to show, the slider will be disabled. The first slider corresponds to the level one, the next one to the level two and so on.

The summary page is divided in two sections, the section of the video player and the section of the sliders, figure D.7.2. This interface allows the user to navigate between the various level of details and dimensions of the scalable summary.



Figure D.7.2: Summary: video player, video slider.

The representative images (keyframes) of video summary of first level are presented in the slider panel. Each slider panel has the keyframe and the time where the frame is located in the video. Clicking on the representative keyframe, allows to navigate the video summary.

The function summary navigation begin on the application layer of the SOA architecture, see figure D.7.3. To interact with the users, Mex-Culture portal use blocks called portlets. This portlets on SOA are the application objects. In this project the application object is the portlet of summary. For the first level of summary, the summary portlet call the middleware object. This object send the information to the component video and database processing in the layer integration and data SOA. After the information was processed, the data processing object and component summary send the information of video section. Keyframes and video shots corresponding to first level of summary. This information is read by a middleware object. Thanks to middleware object the first level of summary video and their keyframes corresponding can be displayed on the application object (summary portlet). To continue with the following levels of the summary, the same procedure is done.

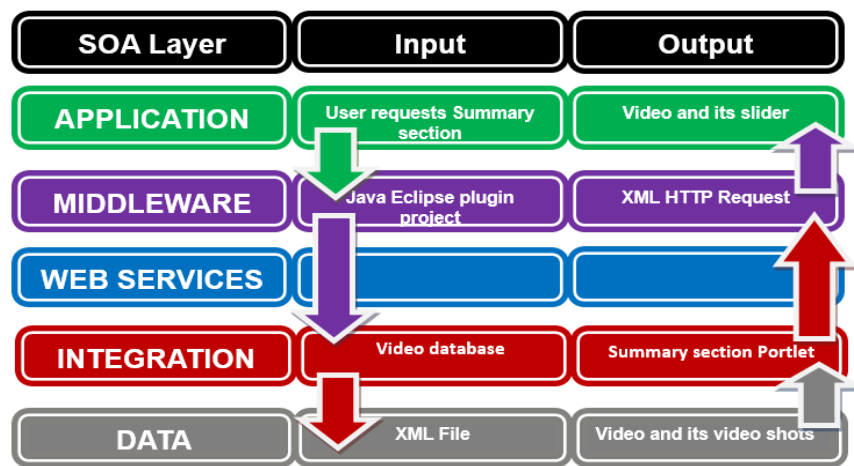


Figure D.7.3: Summary navigation SOA layer diagram.

D.8 ABOUT US

The *about us* section contains the general information of the project and the logos of the partner of the project.

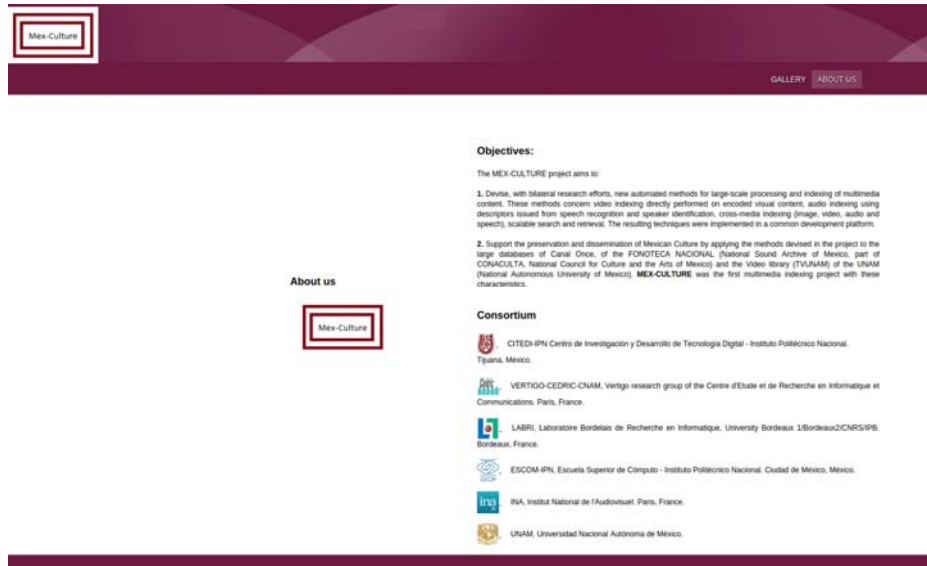


Figure D.3.1: *About us* section.

E CONCLUSIONS

The Mex-Culture platform is designed as a Service Oriented Architecture: the different functionalities are distributed in a set of independent and loosely coupled services.

The platform performs information queries based on the audiovisual content of the database of the Mex-Culture project. The techniques developed allow for example, to retrieve information based on images and videos from their inferred visual characteristics of their pixels. These are conventionally described as giving an account of their texture, color, shape and movement. For audio and speech, it will be similar context. In this document is also described the operational design of the portal showing interesting multimedia functionalities.

F REFERENCES

- [B.1] Thomas Erl., Service-Oriented Architecture: concepts, technology and design. Pearson Education, 2005.
- [B.2] Jeff Davis., Open Source SOA, Publisher: Manning Publications, 2009, ISBN-10 1933988541.
- [C.1] C. Emig, K. Langer, K. Krutz, S. Link, C. Momm und S. Abeck. The SOA's Layers, C+M Research Report, University of Karlsruhe. 2006.

- [C.2] C. Mayerl, F. Troscher, and S. Abeck. Process-oriented integration of applications for a service-oriented it management, in 1st IEEE/IFIP International Workshop on Business-Driven IT Management, Apr. 2006.
- [C.3] Java Platform Enterprise Edition (Java EE). Oracle Technology Network. <http://www.oracle.com/technetwork/java/javaee/overview/index.html>.
- [C.4] Apache, http Server Project. <http://httpd.apache.org/>
- [C.5] The Apache Software Foundation, <http://www.apache.org/>
- [C.6] Liferay portal. <http://www.liferay.com/>
- [C.7] R.J. Allan, C. Awre, M. Baker and A. Fish. Portals and Portlets 2003. Proc. NeSC Workshop 14-17/7/2003 (CCLRC and NeSC, 2004), http://www.nesc.ac.uk/technical_papers/UKeS-2004-06.pdf.
- [C.8] Portal specification JS 168 and API, <http://www.jcp.org/en/jsr/detail?id=168>.