European Commission Seventh Framework Programme
Call: FP7-ICT-2007-1, Activity: ICT-1-4.1
Contract No: 216267

Application Streaming V2

Deliverable No: D7.3

Version 1.0

Editor:	NISV
Work Package:	WP6
Status:	Plan
Date:	M36
Dissemination Level:	RE

Project Overview

Project Name: LiWA – Living Web Archives

Call Identifier: FP7-ICT-2007-1

Activity Code: ICT-1-4.1

Contract No: 216267

Partners:

1. Coordinator: Universität Hannover, L3S Research Center, Germany
2. European Archive Foundation (EA), Netherlands
3. Max-Planck-Institut für Informatik (MPG), Germany
4. Computer and Automation Research Institute, Hungarian Academy of Sciences (MTA SZTAKI), Hungary
5. Stichting Nederlands Instituut voor Beeld en Geluid (NISV), Netherlands
6. Hanzo Archives Limited (HANZO), United Kingdom
7. National Library of the Czech Republic (NLP), Czech Republic
8. Moravian Library (MZK), Czech Republic

Document Control

Title: Application Streaming V2

Author/Editor: Jaap Blom (NISV)

Document History

Version	Date	Author/Editor	Description/Comments
0.1	Jan 14, 2011	Jaap Blom	-
1.0	Jan 25, 2011	Jaap Blom	Final version

Legal Notices

The information in this document is subject to change without notice.

The LiWA partners make no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The LiWA Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Table of Contents

Introduction

This deliverable describes the software architecture of the final version of the WP7 Application: „Streaming“, developed at The Netherlands Institute for Sound and Vision.

Relation with other deliverables: 6.10.

In the annex of this document a global user manual is provided.

Architecture

This chapter provides an overview of the Application: "Streaming" architecture. It describes the involved software components separately. Figure 1 overviews the complete architecture and illustrates the relations between components.

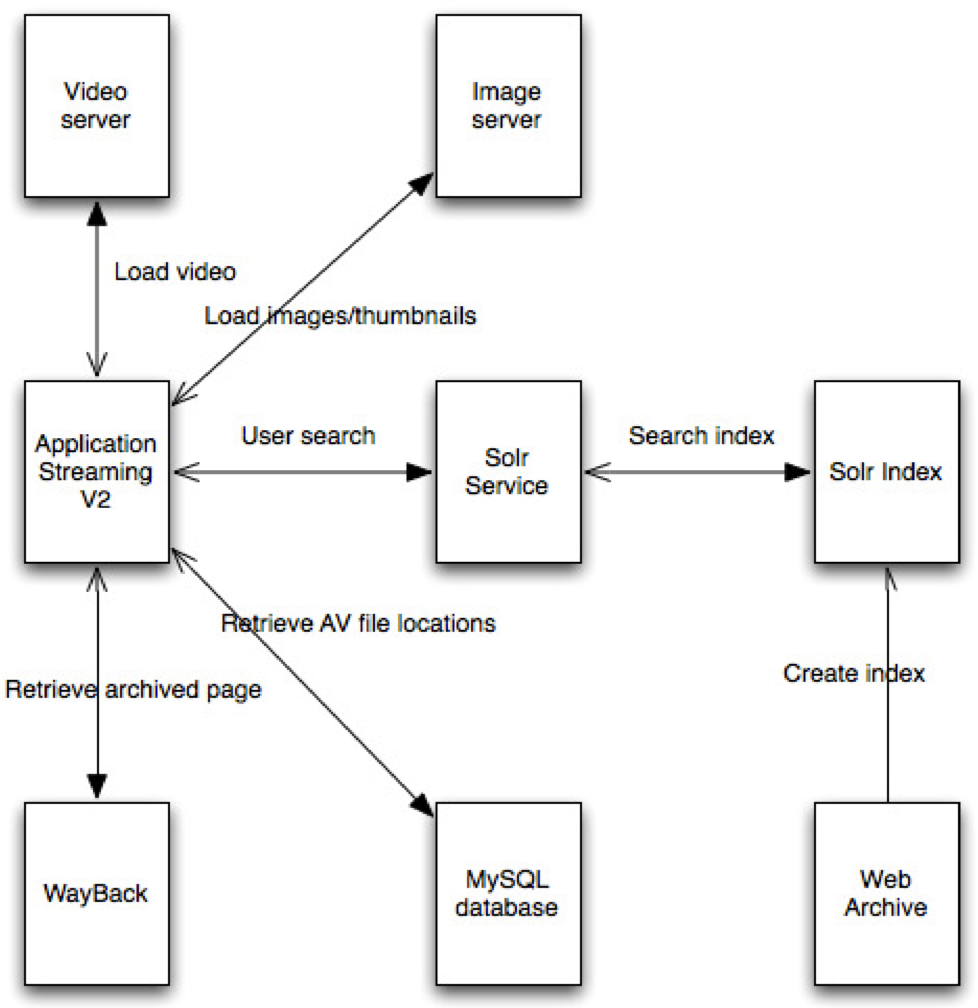


Figure 1: Software architecture of Application: "Streaming"

Main application

The main application is hosted on an Apache Tomcat [1] web server and was implemented using the following technologies:

Technology	Functionality
Java Server Pages [2]	Server side scripting language
Java [3]	Server side programming language

HTML[4]/CSS[5]	Front-end design / styling
JavaScript[6]	Client side scripting language
JQuery[7]	Specialized JavaScript library
AJAX[8]	Client side sending of HTTP requests to server
JSON[9]	Data format sent by the server
JW player[10]	Player used for playing archived video

Solr service

The Solr [11] service is the interface the main application uses for querying the Solr index that is generated from the web archive.

For generating the index from the web archive, the package org.archive.io.arc from the HERITRIX [12] Java library was used to read the contents of the ARC [13] files. To create the actual index documents, the Solr Java library was used.

Video server

All videos that were archived are hosted on the same Apache Tomcat server running the application. This means that all videos that are played in the application are served via HTTP download, rather than a streaming protocol.

Wayback

The Wayback [14] application is deployed as an Apache Tomcat web application. Wayback does not use the Solr index that is generated for the search engine of the application, but generates its own index in order to function.

In order to be able to play archived video files, the Wayback code has been extended. The added code works in such a way that whenever a web page with an archived video is requested by a user, the web page is transformed in the following way:

- the orginal video player is removed from the HTML
- in place of this video player, a custom JW player is inserted

The reason for replacing the original player is, that the original player will always try to load the original video URL, meaning that a live version of the video is loaded from the internet, rather than the archived version. Moreover as each player has different parameters for playing out videos, using a custom player makes it straightforward to implement the right settings for playing out archived video URLs.

Image server

All the thumbnails of web pages and video stills that are shown in the application are stored on an Apache [15] server.

Processing application data

In order for the application to function properly several types of data need to be extracted from the web archive:

- context information
- thumbnails of archived webpages
- stills of archived videos
- application data for the temporal site map

All of the above data is generated with separate pre-preprocessing algorithms using specific tools. In the following chapter each of the items above will be described in more detail.

Context extraction

For the Application Streaming, context extraction means, obtaining the most relevant information from each of the archived web pages. More specifically for web pages containing video players, this entails obtaining the (textual) information that most likely is related to the video that can be played.

In the application user interface, the context information is displayed as descriptive text with every items n the search result list.

The main technology used for obtaining the context information is the Java library called BoilerPipe [16], which offers several functions for analyzing HTML pages and identifying the most relevant pieces of textual information.

Generating thumbnails from web pages

In order to generate the thumbnails for the archived web pages the MozRepl [17] FireFox plug-in is used. MozRepl creates the possibility of communicating with the FireFox browser via the telnet protocol. Also MozRepl has the possibility of adding scripts to FireFox that are executed whenever a web page is requested via the MozRepl telnet interface.
In this case a script is added that enables the creation of a screenshot from any requested web page.

For generating screenshots for the application the following steps are taken:

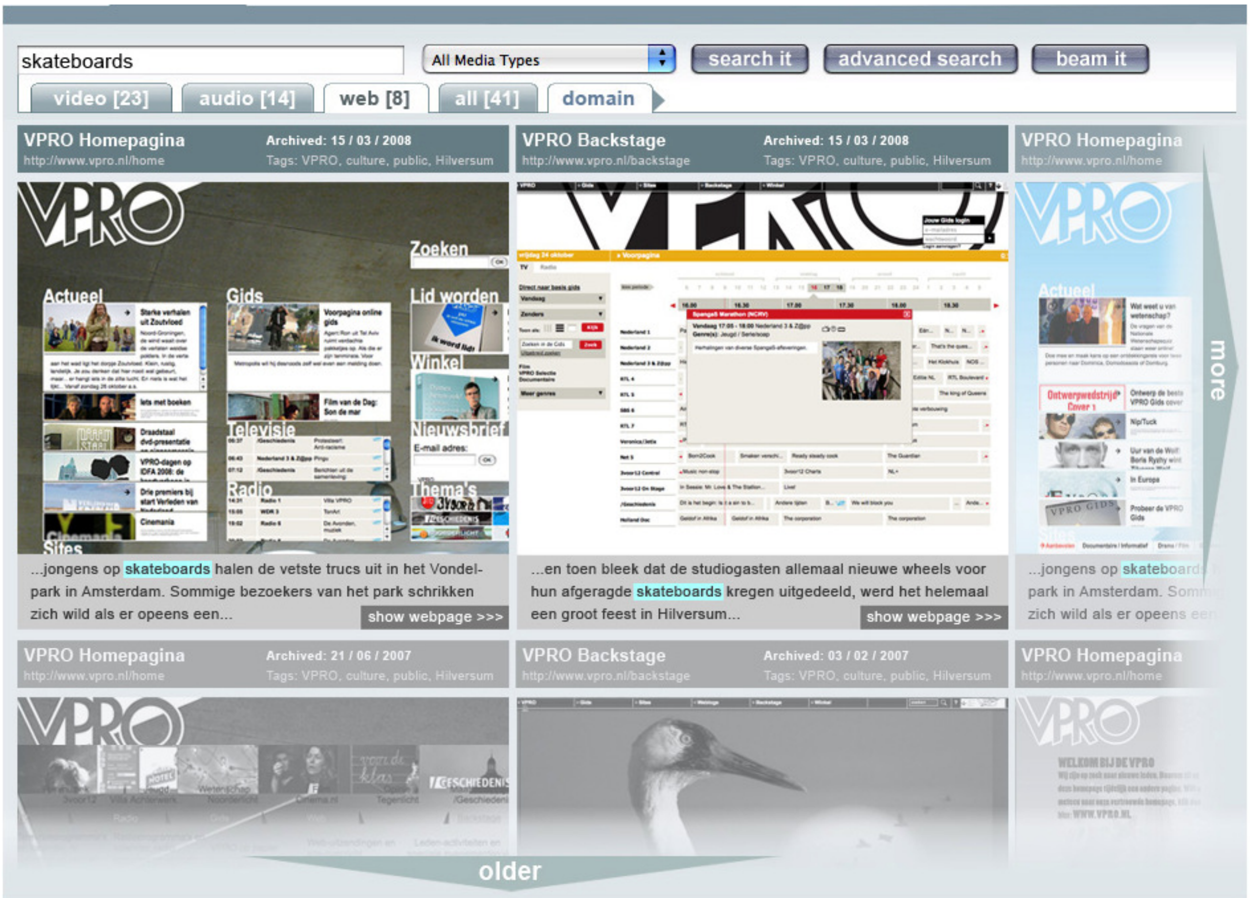
- a Java program runs through all the indexed web pages and for every page:
  - calls the MozRepl script through the telnet interface to make a screenshot
  - stores for each screenshot two versions: a large one for the full view and a thumbnail version
- all screenshots were transferred to the image server

Video stills

The video stills are generated calling Mplayer [18] in a UNIX bash [19] script. The script is configured so that it captures the 5th second of each video.

Temporal site map

In order to implement a time view page that resembles the one presented in the mock-up (see Fig. 2) the page was revised to represent the so called: “temporal site map”, meaning to show the different archived versions of the most notable pages of a domain. This approach also makes it possible to provide of a nostalgic view on an Internet archive related to one of NISV's presented use cases. This use case was mostly intended for the general public.



...jongens op skateboards halen de vetste trucs uit in het Vondel-park in Amsterdam. Sommige bezoekers van het park schrikken zich wild als er opeens een...

...en toen bleek dat de studiogasten allemaal nieuwe wheels voor hun afgeragde skateboards kregen uitgedeeld, werd het helemaal een groot feest in Hilversum...

...jongens op skateboards park in Amsterdam. Somm zich wild als er opeens een

Figure 2: time view page from the Application: “Streaming” mock-up

All the images in the temporal site map page were generated manually. Also the URLs of the archived web pages were chosen manually.

References

[1] Apache Tomcat , http://tomcat.apache.org/

[2] Java Server Pages, JSP, http://java.sun.com/products/jsp/

[3] Java, http://www.oracle.com/technetwork/java/index.html

[4] Hypertext Markup Language, HTML, http://www.w3schools.com/html/

[5] Cascading Style Sheets, CSS, http://www.w3schools.com/Css/default.asp

[6] JavaScript, http://www.w3schools.com/js/default.asp

[7] jQuery, http://jquery.org/

[8] Asynchronous JavaScript and XML, http://www.w3schools.com/ajax/default.asp

[9] JavaScript Object Annotation, JSON, http://www.json.org/

[10] Jeroen Weijering Player, JW Player, http://www.longtailvideo.com/

[11] Apache Solr, http://lucene.apache.org/solr/

[12] HERITRIX, http://crawler.archive.org/

[13] Archive File Format, http://www.archive.org/web/researcher/ArcFileFormat.php

[14] WayBack, http://archive-access.sourceforge.net/projects/wayback/

[15] Apache HTTPD, http://httpd.apache.org/

[16] BoilerPipe, http://code.google.com/p/boilerpipe/

[17] MozRepl, https://github.com/bard/mozrepl/wiki

[18] Mplayer, http://www.mplayerhq.hu/design7/news.html

[19] Bash (UNIX Shell), http://en.wikipedia.org/wiki/Bash_%28Unix_shell%29

Annex A: Application Guide

This section will provide a rough guide on how to use the Application: „Streaming“

Getting started

The Application: “Streaming“ can be found at:

http://rdbg.tuxic.nl:8080/liwa

Arriving on the welcome page, you can either start directly searching the archived content in the simple search bar or do an advanced search by pressing the advanced search button. Another possibility is to navigate to the temporal sitemap by clicking the corresponding link in the top menu.

Searching

In the simple search bar you can simply enter one or more search terms to search the archive. The advanced search offers the additional functionality of being able to specify the archived domain and the date period to narrow down the search results.

When you have performed a search the search results will be displayed in the way that it will split up the three types of archived content, web pages, audio and video, using separate tabs.

Web content

In the tab titled „web“ you can browse all the web archived web pages that were found. By clicking on one of the items, detailed information about that web page is shown in the right hand panel. By clicking on the thumbnail of the web page (if present), you can see the web pages in an enlarged view.

In order to navigate to the actually archived web page, either click on the image in the right panel or on the URL displayed below that image (if the image is present).

Audio content

Because there is no audio material archived, the tab „audio“ will never contain any results

Video content

In the tab titled „video“ it is possible to view all the archived video content that was found. In this view there are two thumbnails for each item of which the first contains an image of the archived web page and the second contains a still of the archived video that was found on that web page.

When clicking the result item, detailed information is displayed into the panel on the right side. In this panel you can either play the archived video in the player that is shown or navigate to the archived web page by clicking the URL. When navigating to the archived page, the archived video is being played from the archive as described in the Wayback section of this document.

Temporal sitemap

The temporal site map was created to have some idea of what a nostalgic viewer of an internet archive could look like.

In order to use it you can simply choose an archived domain from the drop down list to load the domain you are interesting.

When the temporal sitemap viewer is loaded you can use the arrows to navigate through the archived domain. The left and right arrows move you through the list of key pages of the selected domain and the up and down arrows move you through the different versions that were archived fort his domain.

By clicking on one of the images (of archived web pages) you will be directed to the archived web page.