

EZcast

technical documentation

Document written by

- › Michel JANSENS
- › Arnaud WIJNS

from ULB PODCAST team
<http://podcast.ulb.ac.be>
<http://ezcast.ulb.ac.be>
podcast@ulb.ac.be



SOMMAIRE

SOMMAIRE	2
1. INTRODUCTION	3
2. INTERFACES INTERNALS	3
2.1. COMMONS	3
2.2. EZMANAGER	5
2.2.1. REPOSITORY:	6
2.2.2. WEBSPACE:	6
2.2.3. CODE:	6
2.3. EZADMIN	8
2.3.1. WEBSPACE	8
2.3.2. CODE:	8
2.4. EZPLAYER	9
2.4.1. WEBSPACE	9
2.4.2. CODE:	10
2.5. EZRENDERER	10
2.5.1. INTROS	11
2.5.2. WORKING DIRECTORIES	11
2.5.3. CODE	11

1.

INTRODUCTION

EZcast consists of 3 web interfaces:

- EZadmin for users, courses and classrooms management
- EZmanager for recordings management
- EZplayer for watching videos in a rich media player

It also includes EZrenderer, which is the application that processes the video files.

These products are described in this document.

2.

INTERFACES INTERNALS

EZcast is installed in one directory (usually /usr/local/ezcast) and contains the following directories:

- commons
- ezadmin
- ezplayer
- ezrenderer

2.1. COMMONS

The ‘commons’ directory contains libraries, command line (CLI programs) and configuration files that are shared between EZcast products.

LIBRARIES:

- “**<ezcast_root>/commons/lib_auth.php**”: Library that relies on other libraries to authenticate a user and retrieve information about him.
- “**<ezcast_root>/commons/lib_auth_file.php**”: Library that uses files to authenticate and get information about a user (included in lib_auth.php).
- “**<ezcast_root>/commons/lib_auth_ldap.php**”: Library that uses ldap to authenticate and get information about a user (included in lib_auth.php).
- “**<ezcast_root>/commons/lib_courses.php**”: Consultation of courses in EZadmin’s database
- “**<ezcast_root>/commons/lib_mysql.php**”: Library for mySql operations (open DB, close DB, select, insert, ...)
- “**<ezcast_root>/commons/lib_template.php**”: Related to template parsing

- “**<ezcast_root>/commons/lib_various.php**”: Contains various useful functions

NOTE: You can easily create your own authentication library. You just have to:

1. Implement the following functions:

- **x_checkauth(\$login, \$password)**

Checks if user credentials are ok and returns an assoc array containing information about the authenticated user.

@param \$login user’s login

@param \$password user’s password

@return assoc_array | false: user’s information if authentication has succeeded; false otherwise

with **\$assoc_array[‘full_name’]**: user’s first and last names

\$assoc_array[‘email’]: user’s email address (optional)

\$assoc_array[‘login’]: user’s login

\$assoc_array[‘real_login’]: runas (if an admin connects as the user)

- **x_getinfo(\$login)**

Retrieves information about the user without requiring any password (used for runas)

@param \$login user’s login we want information about

@return assoc_array | false: user’s information if the user has been found; false otherwise

see ‘checkauth’ for information on assoc_array

2. Replace ‘x’ in the previous functions name by the name of your authentication
 - → For instance: **sso_checkauth((\$login, \$password)**

3. Write the functions in a file ‘lib_auth_x.php’ located in “**<ezcast_root>/commons**”, with ‘x’ the name of your authentication (i.e lib_auth_sso.php)

4. Add the name of your authentication in **\$auth_methods** array of the **config.inc** file in “**<ezcast_root>/commons**” (i.e \$auth_methods = array(“file”, “ldap”, “sso”);)

COMMAND LINE (CLI PROGRAMS):

- “**<ezcast_root>/commons/cli_template_generate.php**”: Generates the web interfaces using lib_template.php

OTHER FILES:

- “**<ezcast_root>/commons/config.inc**”: Contains the global configuration of EZcast

\$application_url	URL to your EZcast server (http://...)
\$application_safe_url	Safe URL to your EZcast server (https://...)
\$auth_methods	Array containing all available authentication methods
\$basedir	Absolute path to the EZcast main directory
\$copyright	Copyright note (to be included in video titling)
\$db_host	IP address of your database (localhost)
\$db_login	Username for the database
\$db_name	Name of your database
\$db_password	Password for the database

\$db_prefix	Prefix to be added to the tables of your database
\$db_type	Type of your database (mysql, ...)
\$distribute_url	URL to web distribute
\$ezadmin_url	URL to EZadmin
\$ezmanager_basedir	Absolute path to the EZmanager directory
\$ezmanager_url	URL to EZmanager
\$ezplayer_basedir	Absolute path to the EZplayer directory
\$ezplayer_url	URL to EZplayer
\$https_ready	Whether it uses https for authentication or not (your server requires an ssl certificate)
\$ldap_server_auth_json_file	File containing the list of ldap servers for authentication
\$ldap_server_cred_json_file	File containing the list of ldap servers for credentials
\$mailto_alert	Email address aimed to receive alerts from EZcast
\$organization_name	Name of your organization (to be included in video titling)
\$php_cli_cmd	Php binary
\$recorder_basedir	Absolute path to the root directory of EZrecorder
\$recorder_subdir	Path from EZrecorder root directory to the directory aimed to contain the admin.inc file
\$recorder_user	Username for EZrecorder
\$repository_basedir	Absolute path to the root of the repository
\$repository_path	Absolute path to the repository (containing albums)
\$rsync_pgm	Rsync binary
\$ssh_pgm	Ssh binary

- “**<ezcast_root>/commons/ldap_servers_auth.json**”: Contains the list of ldap servers used to authenticate a user (checkauth function in lib_auth_ldap.php)
- “**<ezcast_root>/commons/ldap_servers_cred.json**”: Contains the list of ldap servers used to get information about a user (getinfo function in lib_auth_ldap.php)
- “**<ezcast_root>/commons/install_templates.sh**”: Bash script for templates generation
- “**<ezcast_root>/commons/pwfile.inc**”: File, generated by EZadmin, containing information about the users (encrypted password, full name and email address)

2.2. EZMANAGER

EZmanager is deployed on EZcast server. It is designed following the MVC pattern. The model consists in several libraries (lib_ files) and command lines calls (cli_ files). The views are the web interfaces (templates), gathered in ‘tmpl_source’ folder. The controllers are ‘web_index.php’ and ‘web_distribute.php’, which make the link between the views and the libraries.

EZmanager has parts in three different locations of EZcast server. The first is the repository (see ‘ezcast_infrastructure’ for further information), located by default under ‘/var/lib/ezcast’. The second is the content for users (htdocs), placed in the webspace. The last is the rest of the code, located by default under ‘/usr/local/ezcast/ezmanager’.

2.2.1. REPOSITORY:

The repository can be found in EZmanager "directory (as 'ezcast_tree') and contains working directories, log files and actual repository. Here is its structure:

- **".../ezcast_tree/ezplayer"**: contains user's preferences for EZplayer and message of the day (to display in EZplayer main page)
- **".../ezcast_tree/log"**: contains errors logs and changes logs
- **".../ezcast_tree/queues"**: contains upload directories (from EZrecorder and submits), working directories and scheduling queues for rendering.
- **".../ezcast_tree/repository"**: contains all albums and rendered assets

2.2.2. WEBSITE:

'Htdocs' must be reproduced in the website:

- **".../htdocs/css"**: contains stylesheets for EZmanager
- **".../htdocs/images"**: contains images for EZmanager
- **".../htdocs/js"**: contains javascript functions for EZmanager
- **".../htdocs/swf"**: contains flash files for EZmanager
- **".../distribute.php"**: redirects to web_distribute.php. Allows consultation and download of video files
- **".../htdocs/index.php"**: redirects to web_index.php

2.2.3. CODE:

- Code placed under '**<ezcast_root>/ezmanager'**:
 - **".../tmpl_source"**: contains the templates for the web interface of EZmanager. Templates use labels defined under 'translations.xml' to generate web pages in various languages. Templates are divided in multiple parts to be refreshed independently via AJAX.

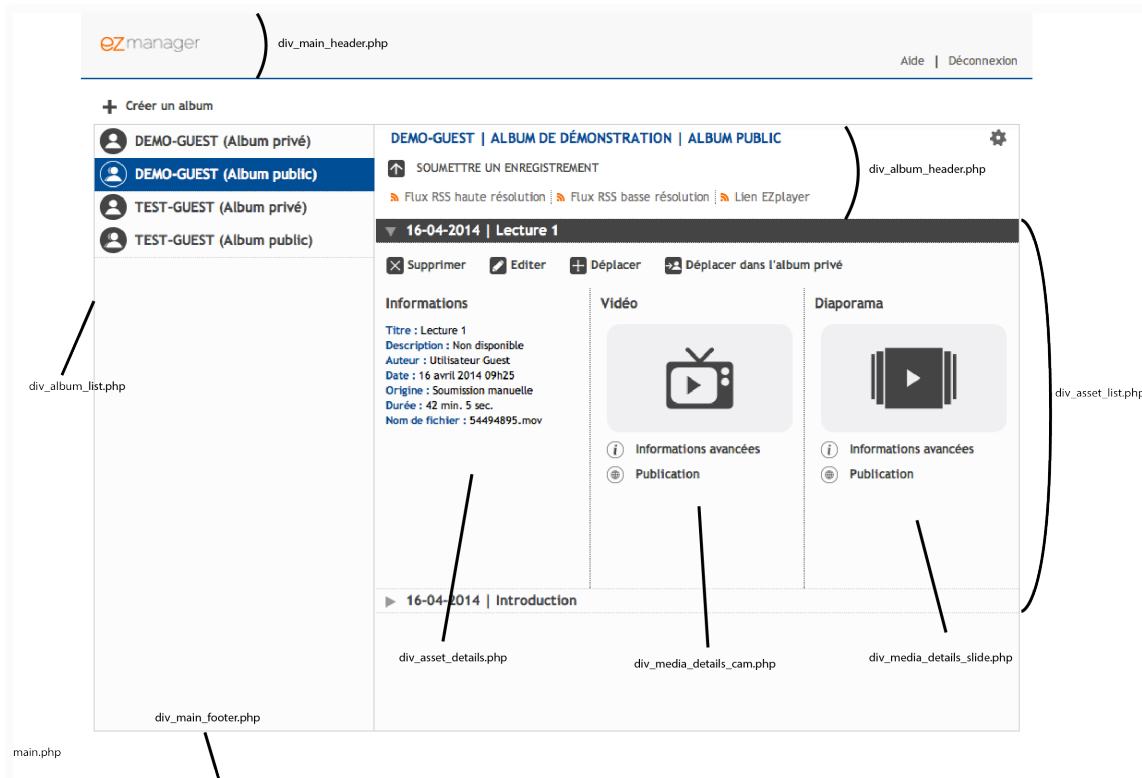


Figure 1 – structure of EZmanager web interface

CONFIGURATION FILES:

- **“.../admin.inc”**: Contains the list of administrators for EZmanager
- **“.../classroom_recorder_ip.inc”**: Contains the list of IP address authorized by EZmanager (included EZrecorder)
- **“.../config.inc”**: Specific variables for EZmanager.
- **“.../renderers.inc”**: Contains the list of all available renderers.

COMMAND LINE (CLI PROGRAMS):

- **“.../cli_mam_insert.php”**: handles videos and slides, from EZrecorder or manual submit. Insert original recordings in the repository (call lib_ezmam.php) and execute the processing on EZrenderer.
- **“.../cli_recorder_download.php”**: download the recordings from the remote recorders (in classrooms)
- **“.../cli_rendered_maminsert.php”**: Archives the video processing logs and transfers the processed video files from the working directory to the repository.
- **“.../cli_scheduler_job_perform.php”**: Copies the video file in a working directory of EZrenderer. Executes a script that handles pending jobs for EZrenderer.
- **“.../cli_scheduler_launch.php”**: Executes a script that handles pending jobs for EZrenderer
- **“.../cli_submit_intro_title_movie.php”**: Creates an XML file containing information for the video processing. Creates a job for EZrenderer and launches the scheduler.
- **“.../install.sh”**: Script for templates creation.

LIBRARIES:

- **“.../lib_acl.php”**: Controls access to albums and assets for a user.
- **“.../lib_error.php”**: Handles and displays errors.
- **“.../lib_ezmam.php”**: Handles the repository; album creation, assets, metadata handling, tokens generation, RSS feeds, ...
- **“.../lib_scheduling.php”**: Creates and handles jobs for EZrenderer.
- **“.../lib_toc.php”**: Handles official bookmarks related to a specific asset (deletes or moves bookmarks according to the actions on the video).
- **“.../lib_upload.php”**: Handles manual video submission from the web interface of EZmanager.
- **“.../lib_various.php”**: Useful functions.
- **“.../web_index.php”**: Controller for EZmanager. All actions executed by an authenticated user in the web interface are intercepted by web_index.php and treated using the libraries.
- **“.../web_distribute.php”**: Handles actions related to the video, such as visualization and downloading.
- **“.../web_download_from_recorder.php”**: Handles HTTP requests from the recorders (in classrooms).

2.3. EZADMIN

EZAdmin allows administrators to manage users, courses, renderers and classrooms. It is deployed on EZcast server and uses a database.

2.3.1. WEBSPACE

‘Htdocs’ must be reproduced in the webspace:

- **“.../htdocs/bootstrap”**: framework for web interfaces
- **“.../htdocs/css”**: contains stylesheets for EZadmin
- **“.../htdocs/js”**: contains javascript functions for EZadmin
- **“.../htdocs/index.php”**: redirects to web_index.php
- **“.../htdocs/install.php”**: web interface for EZcast first install

2.3.2. CODE:

Code placed under **‘<ezcast_root>/ezadmin’**:

- **“.../tpl_source”**: contains the templates for the web interface of EZadmin. Templates use labels defined under ‘translations.xml’ to generate web pages in various languages. Templates are divided in multiple parts to be refreshed independently via AJAX.

CONFIGURATION FILES:

- **“.../admin.inc”**: Contains the list of administrators for EZadmin.
- **“.../config.inc”**: Specific variables for EZadmin.
- **“.../var”**: contains temporary files and generated files
-

COMMAND LINE (CLI PROGRAMS):

- **“.../cli_album_count.php”**: counts the number of albums in EZmanager and update the DB accordingly.
- **“.../install.sh”**: bash script for templates generation.

LIBRARIES:

- **“.../lib_database.php”**: Functions for interactions with the database of EZadmin. It creates a connection with the database and allows the admin to create, edit or delete users, courses, ...
- **“.../lib_error.php”**: Error managing and logging library
- **“.../lib_scheduling.php”**: Functions for handling jobs for EZrenderer.
- **“.../lib_various.php”**: Useful functions for EZadmin
- **“.../web_index.php”**: Controller for EZadmin. All actions executed by an authenticated admin in the web interface are intercepted by web_index.php and treated using libraries.
- **“.../web_install.php”**: Handles the installation of EZcast, through the web interface.
-

2.4. EZPLAYER

EZplayer is a rich media player. It uses video files from the repository, handled by EZmanager. It allows authenticated users to add temporal bookmarks on the videos. It provides anonymous access for users who don't have a login.

2.4.1. WEBSPACE

‘Htdocs’ must be reproduced in the web space:

- **“.../htdocs/css”**: contains stylesheets for EZplayer
- **“.../htdocs/images”**: contains images for EZplayer
- **“.../htdocs/js”**: contains javascript functions for EZplayer
- **“.../htdocs/swf”**: contains flash files for EZplayer
- **“.../htdocs/index.php”**: redirects to web_index.php

2.4.2. CODE:

Code placed under ‘**<ezcast_root>/ezplayer**’:

- **“.../tmpl_source”**: contains the templates for the web interface of EZadmin. Templates use labels defined under ‘translations.xml’ to generate web pages in various languages. Templates are divided in multiple parts to be refreshed independently via AJAX.

CONFIGURATION FILES:

- **“.../admin.inc”**: Contains the list of administrators for EZplayer.
- **“.../bookmarks.xsd”**: structural schema for xml files that contain bookmarks for importation.
- **“.../config.inc”**: Specific variables for EZplayer.

COMMAND LINE (CLI PROGRAMS):

- **“.../install.sh”**: bash script for templates generation.

LIBRARIES:

- **“.../lib_acl.php”**: Functions related to access control. Determines whether or not a user can access to a course, is admin, ...
- **“.../lib_error.php”**: Error managing and logging library
- **“.../lib_ezdam.php”**: Handles the repository; album creation, assets, metadata handling, tokens generation, RSS feeds, ...
- **“.../lib_toc.php”**: Handles official bookmarks related to a specific asset.
- **“.../lib_user_prefs.php”**: Handles user’s preferences and his personal bookmarks related to a specific asset.
- **“.../lib_various.php”**: Useful functions for EZplayer
- **“.../Browser”**: Contains classes for getting information about user’s browser and operating system.

2.5. EZRENDERER

This directory contains the rendering engine. It will be installed by EZadmin on rendering machines. This could be on the same server (same or different account) or on remote servers (dedicated or shared computing platform)

The application can be installed on EZcast server or on a remote server, but in both cases, it will require an SSH link.

EZrenderer is composed of three directories:

- **“<ezrenderer_root>/bin”**: Contains the code of the application
- **“<ezrenderer_root>/intros”**: Contains video files that will be integrated as a jingle in processed movies.
- **“<ezrenderer_root>/queues”**: Contains working directories.

2.5.1. INTROS

For each introduction that you add in the “**config.inc**” file of EZmanager (in variable **\$intros** and **\$default_intro**), you must have an entry in the “**intros**” directory of EZrenderer.

- “**<ezrenderer_root>/intros/<intro_name>**”: contains a list of movies, in various resolutions, that will be used as introduction of the processed video. Every available resolution should be mentioned in “**config.inc**” file in **\$intro_movies** variable.

2.5.2. WORKING DIRECTORIES

- “**.../queues/downloaded**”: where the video file is placed after being downloaded from the repository (EZcast server)
- “**.../queues/downloading**”: where the video file is downloaded
- “**.../queues/failed**”: Video file is moved here if the processing has failed
- “**.../queues/processed**”: Video file is moved here when the processing is finished
- “**.../queues/processing**”: Video file is placed here while being processed

2.5.3. CODE

CONFIGURATION FILES:

- “**.../bin/encoders**”: contains preset files used by ffmpeg for video processing.
- “**.../bin/config.inc**”: Specific variables for EZrenderer

COMMAND LINE (CLI PROGRAMS):

- “**.../bin/cli_job_kill.php**”: kills a job currently processed
- “**.../bin/cli_renderer_test.php**”: Tests that all pre-requisites are installed / enabled.
- “**.../bin/cli_statistics_get.php**”: returns statistics relative to the renderer.

LIBRARIES:

- “**.../bin/intro_title_movie.php**”: Processes the video files: moves movies from downloaded directory to processing directory. Adds a jingle and a title to the movies, encodes the movies in various resolutions.
- “**.../bin/lib_ffmpeg.php**”: Functions for video processing based on ffmpeg. Concatenation of videos, annotation, ...
- “**.../bin/lib_gd.php**”: Functions for generating a title for the video.
- “**.../bin/lib_metadata.php**”: Converts associative arrays in xml files and vice versa.
- “**.../bin/lib_statistics.php**”: calculates statistics relative to EZrenderer.