Webcampak User Guide

Eurotechnia Ltd.

Contents

Chapter 1

Overview

Overview

Webcampak or Webcampak Cloud

You probably have seen on our website mentions of **Webcampak** and **Webcampak Cloud**, those two names refer to the same software but installed in a different context:

- **Webcampak** is typically installed on low-power embedded computers and are directly attached to D-SLR cameras
- **Webcampak Cloud** is typically installed in a datacentre on high performance servers and are used as a central place to store and process pitures coming from different **Webcampak** systems.

But as mentioned above, we took early on the decision to only have one single software. It greatly facilitates development (one codebase to maintain), but it also means a need for greater education and understanding from users, on the system's capabilties. For example, nothing in the software will prevent users from starting compute intensive manipulations on a **Webcampak** since it might make sense in some situations.

For example, we don't have mechanisms in place to prevent a user to configure a capture rate of 1 picture every 10s, requesting at 27 degree rotation of the picture on the fly. We know it's likely going to create difficulties in most situations, but maybe in this use case, webcampak is running on an extremely powerful computer with a high-speed data transfer camera.

But webcampak try to be as verbose as possible on the impact of various settings, those are usually timed, logged and available to users to assess performance. One of our recommendation, when discovering the system, is to start small and progressively add more and more complexity into the setup to evaluate the impact of configuration settings.

So education and understanding of the implication of using available settings is key for a good operation of the system.

Capture devices (D-SLR or others)

Webcampak has been built to be as flexible as possible and adding new type of capture devices into the software should be relatively straight-forward for developers (us).

The most common use-case though is the use of D-SLR cameras, Webcampak capture acquisition process relies on gPhoto2, you can find a list of known supported D-SLR cameras on this page of their documentation

Open Source

Webcampak is open source, unless specified otherwise, most of its codebase is licensed under GNU GPL v3.

We are strong believers of the benefits of using and developing open source software, and welcome external contributions.

Chapter 2

Sources

Files

Your Webcampak will probably have to store a (very) large number of pictures. It has been designed for this task and multiple features are available to manage and secure those files.

Redundancy

Always keep in mind that hardware failure or human error (i.e. delete pictures by mistake) can happen and it's always better to take precautions before such events happen.

Webcampak is able to send pictures to remote FTP server, you can also store pictures automatically on another source of the same Webcampak to prevent human errors (just keep in mind it will require twice the amount of disk space).

Automated deletion

Webcampak is equipped with automated deletion mechanisms to ensure its internal hard disk never runs out of space. Those mechanisms can be configured to fit various requirements.

When Webcampak store its pictures to a remote server, it can use its internal disk space as a buffer to ensure no pictures are lost in case of network issue. Depending of the capture frequency and picture size, this buffer size can span days, even months.

File structure

In most cases, access to webcampak files is done through FTP, using some of the automatically created account.

A gloabl FTP account (wpresources) is available for admins to access shared webcampak elements, such as:

- cache files (temporary files created by Webcampak)
- webcampak database (users, permissions, sources details)
- emails (queued, sent, failed)
- configuration files
- logs
- statistics
- common watermark files (shared between all sources)
- xfer (queued, completed, failed)

Each source then get its own FTP account giving access to its own directory tree, containing:

- **live**: directory containing hotlink pictures and videos

- **pictures**: directory containing pictures archives, with one sub-directory per capture day
- **resources**: direcotry containing various source specific elements necessary for processing and source operation
- **tmp**: temporary directory used in processing and/or picture acquisition
- **videos**: directory containing generated videos

Chapter 3

Capture

Capture

Interval

Webcampak capture pictures at an interval defined in minutes or second in a source configuration, with captured triggered based on clock time. For example, if a source is configured to capture every:

- **20s**: The capture process will be triggered at 5:00“00, 5:00”20, 5:00“40, 6:00”00, etc...
- **10mn**: The capture process will be triggered at 5:00“00, 5:10”00, 5:20“00, 5:30”00, etc...

To get a “clean” and consistent output we recommend using one of the following settings:

- **Seconds**: 10, 20, 30
- **Minutes**: 1, 5, 10, 15, 20, 30, 60

Note: a 10s capture frequency will highly depend of your hardware, and might not be feasible in most situations.

Precise Capture Time

When capturing from a D-SLR camera, and depending of multiple factors, the actual capture process might be triggered a few miliseconds or seconds after the defined interval. Depending if you care about time accuracy or not, you can specify if the picture should take the time it the capture request was sent (interval time) or the time the picture was actually captured.

Filename

Webcampak uses the capture time in file naming, using the following convention: YYYYMMDDHHmmSS (YearMonthDayHourMinuteSecond). For example 20170224124007.jpg is a picture captured on February (02) 24th, 2017 at 12:40 and 07 seconds.

To prevent too many files from being stored in the same directory (which creates issue for very long projects) pictures are stored in a directory corresponding to the picture's day, for example, filepath for the above picture will be: pictures/20170224/20170224124007.jpg

Calendar

A 24h format weekly calendar is available in each source to identify when to capture pictures (day of the week, time of the day).

If you want to enable variable capture rate (for example capture business hours at a different rate than outside business hours), you would configure two sources with a different non-overlapping capture calendar.

Non-overlapping sources

When configuring sources to capture from the same physical source (i.e. same D-SLR camera), be cautious not to have overlapping capture requests. Cameras can only handle one capture at a time and which source will get the picture will be "first-come-first-served", with the other source failing and the risk of causing the camera to crash.

RAW Files

Raw pictures are stored in a specific sub-directory, for the above picture it will be: pictures/raw/20170224/20170224124007.jpg

It might be useful to understand that, in webcampak, raw pictures are mostly considered as a supplement to jpg pictures, but picture manipulation happens on jpg pictures only. Raw pictures are stored in a different directory tree and at various stages, the system will check for the presence of a raw pictures (and eventually configuration settings) to determine if such files should be sent.

Chapter 4

Logging

Logging

This is something that has been dramatically extended from webcampak 2.x and is worth its own section in the documentation to emphasis on why reviewing logs can be relevant.

Logs usually do not take a vast amount of physical space, so we decided to built it into most of the processing done by webcampak and log almost everything. Some logs would likely never be used when some can be especially useful in the configuration stages to understand what is happening under the hood.

Configuration changes

All configuration changes to sources or general configuration are logged, this lets you understand who changed what when. This is especially useful if you want to rollback to a previous configuration setting but do not remember what this was.

Capture logs

Most actions taken automatically by webcampak when it captures and manipulate pictures are logged, and those logs also provide timed events,not only the timestamp but the time it took to run some actions.

The most relevant value in capture logs is the overall capture time (20.7s in the below capture), which gives you an idea on the maximum capture rate to be supported by the system.

```
1 2017-02-24 14:50:33,864 (INFO) webcampak : capture.run(): Capture: Overall capture time: 20752 ms
```

Then, digging into the logs, you can identify the time taken by various actions, for example (below), creating a 1920x1080 hotlink file took 5.6 seconds.

```
1 2017-02-24 14:50:21,508 (INFO) webcampak : captureUtils.generateHotlinks(): Hotlink File:
    /home/webcampak/webcampak/sources/source1/live/webcam-1920x1280.jpg
2 2017-02-24 14:50:27,157 (INFO) webcampak : pictureTransformations.resize(): Resized picture to 1920x1280
    in 5646 ms
```

It is safe to assume that in this case, disabling this particular action in the configuration would save 5s and bring the capture time closer to 15s, potentially allowing to increase capture rate.

Chapter 5

File Management

File Management

Over a project's lifetime, a webcampak system will capture, manipulate, transfer and store a very large number of files. Different mechanisms have been implemented to ensure storage and transmission is performed in an efficient and secured manner.

Local storage

Webcampak is usually shipped with a 256 or 512 GB SSD providing ample space for local storage. In a vast majority of situations, captured pictures are transferred to a NAS or another Webcampak (cloud in particular).

In this situation, local storage can be considered as a buffer, the system will be configured to keep X days (X depending of multiple factors, but ofter a month or two) of pictures locally, automatically clearing older pictures.

Using webcampak local storage as a buffer is the ideal solution to accomodate network connectivity issues, when such an event happen, you will have X days to identify and address the issue before starting to loose pictures.

At the end of the incident, a webcampak feature (Xfer Reports) to transfer back missing pictures to remote servers (more below).

File Synchronization

Webcampak "Xfer Reports" features provides means to ensure all files on a local webcampak are also available in remote locations by comparing file names and file sizes. By looking at a report, users can decide to initiate transfer of any missing files.

This feature cover multiple use cases:

- Re-synchronize files after a network incident
- Transfer all files at once to a new destination (a new NAS for example)
- Ensure that no pictures are missing

Bandwidth Constrains

Bandwidht constraints sometimes prevents pictures form being transferred as they are captured, for example if there is a higher capture rate during a specific timeslot. The "Xfer" feature will de-serialize picture transmission from the capture process.

New pictures are placed in a queue, which is then processed as per configured parameters (in particular number of parallel transfers). This prevent risks of overloading the system with too many parallel jobs.

Quota

Each Webcampak source can have a filedisk quota, to define how much space can be used by the source on this system. This configuration is a "**Soft Quota**", going over the limit will not trigger actions others than reporting over-usage. Pictures are critical to our system, and we didn't want to take any action that might result on improperly deleting pictures. If a source goes over quota, the system will report so, but will not take further actions.

Chapter 6

Users

Users

Webcampak has an authentication and an authorization module providing feature and source isolation.

Groups

Webcampak handles feature authorization on a group level. When configuring a group, users can select which features (view pictures, configure sources, run reports, etc...) are available for users in this group.

By default, Webcampak is shipped with three groups:

- **View**: Access pictures and videos, no configuration
- **Configure**: Access pictures and videos, only configure sources (no access to system-level configuration)
- **Admin**: Access to all Webcampak features

Sources Access

Source access is managed on a per user basis and is independent from feature access (managed through user groups). A user member of a "configuration group" and given permission to access sources A, B and C will be able to configure those three sources.

Webcampak does not support further granularity, for example it is not possible for user Joe to have config permission on source A & C and view-only permission on source B. This use case has not been identified relevant in our use cases. Nothing prevents an admin from creating multiple usernames to cover this use-case.

Customers

Users can be attached to a "Customer", this configuration has very limited impact on Webcampak and is only used to identify user provenance and customize background color and logo.