

EXT: Video Basic (Imagine virtual services)

Extension Key: dam_ivs_videobasic

Language: en, de

Keywords: keywords comma-separated

Copyright 2009, Sebastian Siebert, <ss@imagine-vs.de>

This document is published under the Open Content License
available from <http://www.opencontent.org/opl.shtml>

The content of this document is related to TYPO3
- a GNU/GPL CMS/Framework available from www.typo3.org

Table of Contents

EXT: Video Basic (Imagine virtual services).	1	Administration.....	8
Introduction.....	3	FAQ.....	8
What does it do?.....	3	Configuration.....	9
Requirements.....	3	FAQ.....	9
Screenshots.....	3	Reference.....	10
Users manual.....	4	Known problems.....	12
Add a new video to dam.....	4	To-Do list.....	13
Add a new video to content.....	6	ChangeLog.....	14
FAQ.....	7		

Introduction

What does it do?

This extension handles the videos with encoding server und shows those in a flash player NonverBlaster. Required extension is Media (DAM).

Requirements

- Server with access via SSH
- PHP >= 5.2.0
- ffmpeg >= 0.5
- qt-faststart (Tool are included at ffmpeg)
- gcc are using for the build the tool qt-faststart

Screenshots

See next chapter "Users manual"

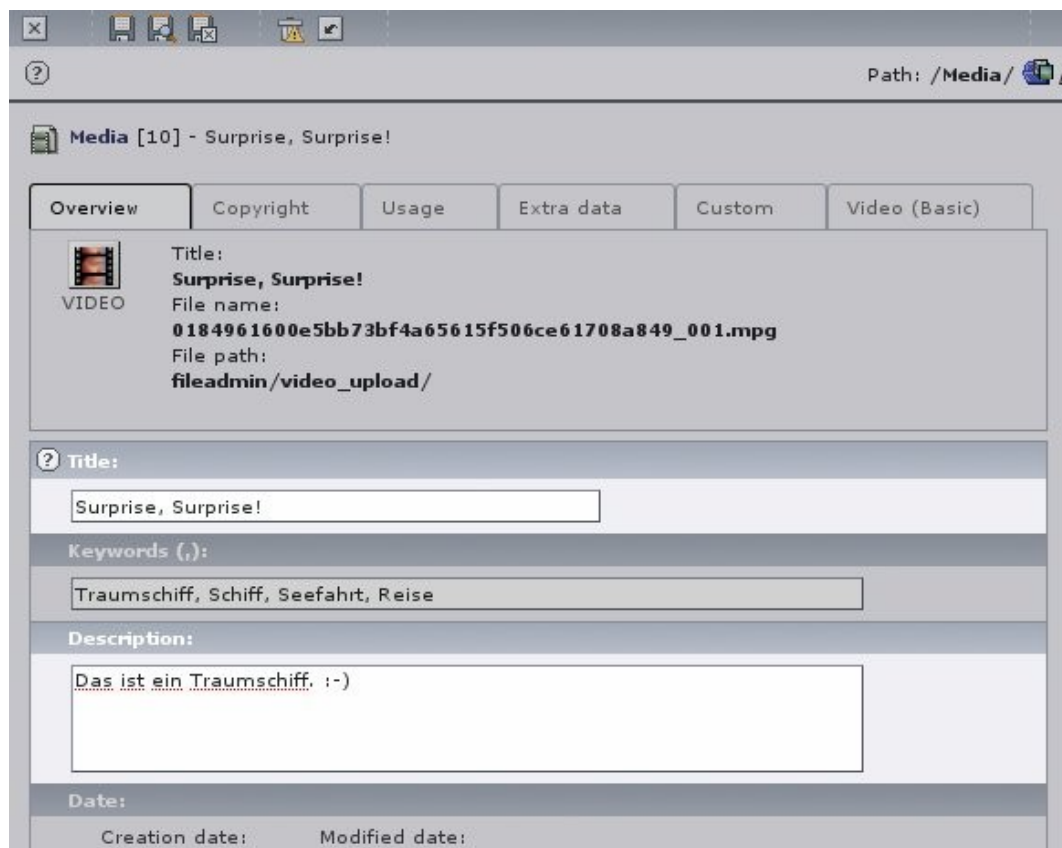
Users manual

Add a new video to dam

1. Module File → go to video directory (if doesn't exist, create a folder called "video")
2. Select in the drop down field "Upload"
3. Select a video to upload on your webserver
4. Module "File" → select your video to edit.



5. Fill the video information in title and description. This information will use in front end content:



6. Go to the tab "Video (Basic)"

The screenshot shows the 'Video (Basic)' tab of the dam_ivs_videobasic extension. The interface includes a top bar with a path '/Media/' and a PID of 23. Below the path is a title bar 'Media [10] - Surprise, Surprise!'. The main area has several tabs: Overview, Copyright, Usage, Extra data, Custom, and Video (Basic). The 'Video (Basic)' tab is active, showing the following fields:

- Encode:** A checkbox that is currently unchecked.
- Location:** A dropdown menu set to 'local'.
- Screenshot:** A section containing a text input field with a long hexadecimal string, a file icon, and a file preview. The file preview shows a thumbnail of a building and the following details:
 - File name: 7bdf268c02b3305269748d0609597fb70ac49ade_6.jpg
 - File size: 15.1 KB
 - Metrics: 352x288 px
- Directory (Stream):** An empty text input field.
- Filename (Stream):** An empty text input field.
- Width:** A text input field with the value '0'.
- Height:** A text input field with the value '0'.
- Status:** A dropdown menu set to 'not submitted'.

At the bottom of the interface, there is a checkbox labeled 'Show secondary options (palettes)' which is checked.

Description:

Encode → Set this video to encoding into MP4-Format. If you save the informations of video, it will be send the informations to video encoding server (See chapter "Configuration of the dam_ivs_videobasic extension")

Location → In this basic variant is only local available. If you wish to upgrade to professional variant, you can select a content delivery network called "amazon web services" too.

Screenshot → After encoding process you can select a screenshot under the folder called encoded_basic_(videoname)

Directory (Stream) → Show the path to the video (not editable)

Filename (Stream) → Show the filename of the video (not editable)

Width → Show the width of the video (not editable).

Height → Show the height of the video (not editable).

Status → This is the encoding status of the video. Possible status information: submitted, wait (encoding), encoding error, ready

7. Save the additional information

Add a new video to content

1. Module Page → create a new element as “Regular text element”
2. Switch the type to “Video Basic (Flash)”

Path: /Tiefkühl-Commun.../plugin/ [pid: 24]

Pagecontent [60] - Test

General | Video (Basic) | Access

Type: Video Basic (Flash)

Language: Default Columns: Normal Before: After: Frame: Index: ☒

Hide: ☐

Header: Test

Align: Type: Normal Link: Date:

To top: ☐

☒ Show secondary options (palettes)

3. Switch to the tab “Video (Basic)”

General | **Video (Basic)** | Access

Video

Surprise, Surprise!

Media

MPG MPEG MP4 AVI WMV

File name: 0184961600e5bb73bf4a65615f50

File size: 3.7 MB

Position:

Above, center

Columns: 1 No rows: ☒ Border: ☒

Video → Select the uploaded video and added to media field

Position → Set the position of the flash player. Possible are center, right, left

FAQ

No FAQ for User available

Administration

Please install the Media (DAM) extension (dam) with the Extension Manager:

1. Go to Extension Manager (Ext Manager)
2. Select in the dropdown field "import extensions"
3. Put into the textfield: "dam" (without quotes)
4. Install the dam extension called as "Media (DAM)" in the category "Backend Modules"

Some extension for dam are recommend:

- MediaCategories (dam_catedit)
- DAM for Content Elements (dam_ttcontent)

Installation of the dam_ivs_videobasic extension:

1. Install the dam_ivs_videobasic extension with the Extension Manager
2. Include static (Video Basic (dam_ivs_videobasic)) from this extension to your template configuration

Build qt-faststart from ffmpeg-tools (Shell)

1. Download the source code from ffmpeg <http://ffmpeg.org/download.html>
2. Unpack the archive: tar xvf ffmpeg-0.5.tar.bz2
3. Change to directory: cd ffmpeg-0.5/tools
4. Compile the tool: gcc qt-faststart.c -o qt-faststart
5. Install the tool with root: install -m 755 qt-faststart /usr/bin/qt-faststart

FAQ

No FAQ for Administration available.

Configuration

Configuration the encoding script:

- You need a webserver via SSH to create a cronjob. The Server have to access the part of the encoding script from network.
- Unpack the archiv on the webserver `./dam_ivs_videobasic/3rdparty/typo3video.tar.bz2`
- Set in the configuration the database access, the path to the applications and the URL of encoding server: `/path/to/your/script/typo3video/config/config.php`
- Type into the shell: `crontab -e`
- Insert follow line into the editor: `* /5 * * * * /usr/bin/php -f /path/to/your/script/typo3videobasic_cronjob.php >> /var/log/typo3video.log`
- Create a new database for the encoding script: `typo3video`
- Import the create.sql into the database “typo3video” (included in the tar.bz2-Archive)
- It's expressly recommended to create a new user for the created database “typo3video”.

Add a project to your encoding server (Shell):

- `/path/to /your/script/typo3video_command.php add (projectname)`

Create a new hash (Shell):

- `/path/to /your/script/typo3video_command.php renewhash (projectname)`

Remove a project with all their files (Shell):

- `/path/to /your/script/typo3video_command.php remove (projectname)`

Configuration of the dam_ivs_videobasic extension:

- URL to the encoding server (without typo3video.php): p.e. <http://www.sitename.com/>
- Project hash (see “Configuration the encoding script”). p.e. `2227C83e7a9856590c3076cf0be5af3f`
- Number of screenshots from video (Integer): p.e. `10`
- Set the video width for embed flash (It's affected the encoded video too): p.e. `352`

CONFIGURATION:
(Notice: You may need to clear the cache after configuration of the extension. This is requir these settings.)

URL to the encoding server [urlServer]

Project hash [projectHash]

Number of screenshots from video [screenshots]
 (Integer)

Set the video width for embed flash... [fixwidth]
 Set the video width for embed flash (px)
 (Integer)

FAQ

No FAQ for Configuration available

Reference

TypoScript Configuration for plugin.tx_damivsvideobasic_pi1

Property:	Data type:	Description:	Default:
ampersand	string	Sets the ampersands in documents. Possible parameters: html, xhtml	xhtml
flashParams.quality	string	Sets the quality of the video. Possible parameters: low, high, autolow, autohigh, best	high
flashParams.allowscriptaccess	string	Sets the security restriction to Flash. Possible parameters: always, sameDomain, never	always
flashParams.wmode	string	Sets the window mode property of the Flash movie in the browser: Possible parameters: window, opaque, transparent	transparent
flashParams.swliveconnect	boolean	Sets this parameter for running Flash and Javascript on same time. Possible parameters: true, false	true
flashParams.allowfullscreen	boolean	Sets to allowed to display Flash in fullscreen mode. Possible parameters: true, false	true
flashParams.scale	string	Sets for resize the video. Possible parameters: showall, noborder, exactfit	default
flashVars.allowsmoothing	boolean	Sets for allowed smoothing: Possible values: true, false	true
flashVars.autoplay	boolean	Sets to play the video immediately after the DOM-Object are ready. Possible values: true, false	false
flashVars.buffer	integer	Sets the buffer as time in second	3
flashVars.showtimecode	boolean	Sets to show Timecode. Possible values: true, false	true
flashVars.loop	boolean	Sets to loop the video. Possible values: true, false	true
flashVars.controlColor	hexdezimal	Sets the color of control panel in hexdezimal.	0xf4f1ec
flashVars.scaleiffullscreen	boolean	Sets to allowed to scaling the video on fullscreen	true
flashVars.showscalingbutton	boolean	Sets to allowed to scaling the buttons on fullscreen	true
videoTemplate	string	<p>You can set your own htmlcode for flashvideo:</p> <pre><div> ###FLASHVIDEO### <div class="video_title">###TITLE###: ###VIDEO_TITLE###</div> <div class="video_description">###DESCRIPTION###: ###VIDEO_DESCRIPTION###</div> </div></pre> <p>Follow Placeholder are possible:</p> <p>###FLASHVIDEO### - Insert the video flash object ###TITLE### - Insert the label title in your language ###VIDEO_TITLE### - Insert the title from video via dam ###DESCRIPTION### - Insert the label description in your language ###VIDEO_DESCRIPTION### - Insert the description from video via dam</p> <p>That's all</p>	

Example

You can configure the flash player or the html-output as TypeScript, here an example:

```
plugin.tx_damivsvideobasic_pi1 {
    ampersand = xhtml
    flashParams {
        quality = high
        allowscriptaccess = always
        wmode = transparent
        swliveconnect = true
        allowfullscreen = true
        scale = default
    }
    flashVars {
        allowsmoothing = true
        autoplay = false
        buffer = 3
        showtimecode = true
        loop = true
        controlcolour = 0xf4f1ec
        scaleiffullscreen = true
        showscalingbutton = true
    }
    videoTemplate (
        <div>
            ###FLASHVIDEO###
            <div class="video_title"><strong>###TITLE##:</strong> ###VIDEO_TITLE###</div>
            <div class="video_description"><strong>###DESCRIPTION##:</strong>
            ###VIDEO_DESCRIPTION###</div>
            </div>
        )
    }
}
```

You can change the CSS in your CSS layout, if you wish:

```
.tx-damivsvideobasic-pi1 .video_left {
    width: 100%;
    text-align: left;
}

.tx-damivsvideobasic-pi1 .video_center {
    width: 100%;
    text-align: center;
}

.tx-damivsvideobasic-pi1 .video_right {
    width: 100%;
    text-align: right;
}
```

Known problems

- no bugs, problems or performance issues.

To-Do list

- not yet available

ChangeLog

22.06.2009 – Sebastian Siebert

- initial development