

Using the Media libraries application, users can store various files, for example, photos, pictures, sound files, videos, package files, presentations or document files. This means that media libraries are able to store non-media types of files as well. You can set the site up to allow visitors to view and add content to media libraries. Media libraries can also be associated with certain [groups](#).

This page covers some of the more general Media library settings:

- [Adjusting general media library settings](#)
- [Supported file types in Media libraries](#)
- [Allowing custom file types to be added into media libraries](#)
- [Configuring maximum file upload size in media libraries](#)
- [Format of media library files on the disk](#)
- [Configuring custom storage for media libraries](#)
- [Media library limitations when storing files in an external storage](#)

Adjusting general media library settings

1. Navigate to **Settings -> Content -> Media**.
2. Adjust the [relevant settings](#).
3. **Save** the changes.

Supported file types in Media libraries

Media libraries support the following types of images: **bmp, gif, png, jpg, jpeg, tiff, tif**. These types of image files can be displayed on your site by default.

The system detects audio and video files based on their [MIME type](#) (audio types starting with the *audio/* prefix and video types starting with the *video/* prefix). For successfully detected audio and video file types, the system renders appropriate HTML5 markup. The types of files that can be previewed or played on your site therefore depend on the web browser (modern HTML5 web browsers typically support one or more of the **ogg, mp4**, or **webm** video codecs, and **mp3, wav**, or **ogg** audio codecs).



All other file types are recognized as **documents**, which means that you cannot play them as videos or display them as images. You can still store them in the library and users can download them on the live site.

Allowing custom file types to be added into media libraries

You can allow custom file types by modifying the site's **web.config** file (or properties of the **MediaControl.ascx** control). The system can recognize and process any media types, if you have the right player for the given file type. By specifying the file types in the web.config keys, the automatic detection and support of media files is disabled and only the specified file types are recognized as image/audio/video files by the system.

Example of configuring custom detection of media file types

To allow only media files of selected types to be added into media libraries:

1. Add the following keys into the **appSettings** section of your **web.config** file.

```
<add key="CMSImageExtensions" value="bmp;gif;ico;png;wmf;jpg;jpeg;tiff;tif" />
<add key="CMSAudioExtensions" value="wav;mp3" />
<add key="CMSVideoExtensions" value="mp4;ogg" />
```

2. Save the **web.config** file.

Allowing upload of selected files into media libraries

1. Open the **Settings** application.

2. Navigate to the **Content -> Media** category.
3. Add the selected file extensions into the **Media file allowed extensions** setting (separated by semicolons).
4. **Save** the changes.

Configuring maximum file upload size in media libraries

The default maximum size for files uploaded into media libraries is approximately 28.6MB (limited by the IIS [Request Limits](#) configuration). To enable uploads of larger files:

1. Edit your project's **web.config** file.
2. If necessary, modify the value of the **<httpRuntime>** element's **maxRequestLength** attribute. Enter the value in **kiloBytes**.



Important: Modify the **<httpRuntime>** element directly under the web.config's main **<system.web>** section, NOT within a specific **<location>** element.

Do not edit the *maxRequestLength* value inside the **<location>** element marked by the **WebDav location** comments, which the web.config contains by default.

```
<httpRuntime maxRequestLength="2000000" ... />
```

3. Add the following **<security>** element into the web.config's main **<system.WebServer>** section. Enter the value of the **<requestLimits>** element's **maxAllowedContentLength** attribute in **Bytes**.



Important: Do NOT modify the **<security>** element inside the **<location>** element marked by the **WebDav location** comments, which the web.config contains by default.

```
...
<system.webServer>

  <security>
    <requestFiltering>
      <requestLimits maxAllowedContentLength="2147483648" />
    </requestFiltering>
  </security>

  <modules>
    <remove name="WebDAVModule" />
    <remove name="XHtmlModule" />
    <remove name="CMSApplicationModule" />
    ...
  </modules>
</system.webServer>
```

4. Save the **web.config** file.

Users can now upload files in the size that you specified in the **maxRequestLength** and **maxAllowedContentLength** attributes into Media libraries.

Format of media library files on the disk

When the thumbnail image or the image itself is larger than the required thumbnail image size, the system creates a copy in the required size. Both thumbnails and resized copies are stored in a **hidden folder** in the root of the media library folder. You can set the name of the folder in **Settings -> Content -> Media -> Media file hidden folder**.

- **File** - <file name>.<file extension>.

- **Resized file** - <hidden folder>/<file name><file extension><width>_<height>.<extension>.
- **Thumbnail** - <hidden folder>/<file name>_<file extension><thumbnail suffix>.<thumbnail extension>.
- **Resized thumbnail** - <hidden folder>/<file name><file extension><thumbnail suffix><width>_<height>.<thumbnail extension>.

Example

If you upload the following files:

- **Uploaded file** - *MyImage.jpg*.
- **Uploaded thumbnail** - *MyPhoto.bmp*.

with the following settings defined in **Settings -> Content -> Media**:

- **Media file hidden folder** - *__thumbnails*.
- **Media file thumbnail suffix** - *_thumbnail*.

the system stores the uploaded files in the media library folder with the following names:

- **File** - *MyImage.jpg*.
- **Resized file** - *__thumbnails/MyImage_jpg_100_200.jpg*.
- **Thumbnail** - *__thumbnails/MyImage_jpg_thumbnail.bmp*.
- **Resized thumbnail** - *__thumbnails/MyImage_jpg_thumbnail_20_30.bmp*.

Configuring custom storage for media libraries

The default location of all libraries of a particular site is *~/<sitename>/media*. To customize the location:

1. Navigate to **Settings -> Content -> Media -> Media libraries folder**.
2. Enter the folder location you want the system to store the files in. The location needs to be in one of the following formats:
 - a. **physical path** - for example, *c:\Libraries*.
 - b. **root-relative path** - for example, *~/Libraries*.
 - c. **UNC path** - for example, *\\<server name>\Libraries*.
 - **Note** that if you want to use a UNC path while hosting your media library on a different server, you need to use the *CMSAllowCheckIOPermissions* [web.config](#) key set to false.
3. (Optional) Enable the **Use site-specific subfolders for custom media libraries folder** option. The system then stores media library files in a sub-folder named as the site code name under the custom files folder. That is, *<custom media libraries folder>/<site code name>*.
4. **Save** the changes.



Notes:

- When you change the location of the media library folders, the system does NOT automatically move existing media files and sub-folders to the new location. To ensure that your media content is accessible and works correctly, you need to manually move the files from the old location to the new one in the file system.
- The system always generates only permanent URLs for media files stored in custom media library folders located outside your project.

i If you are running the system on a **web farm** and have **the same UNC root defined** on all servers, it is necessary to add the following key into your *web.config* file. The files stored in the libraries are then not transferred when synchronizing the web farm content:

```
<add key="CMSWebFarmSynchronizeMediaFiles" value="false" />
```

Media library limitations when storing files in an external storage

If your project is configured to store files in an external storage, for example [Azure Blob Storage](#) or [Amazon S3](#), make sure that you structure the media files into a hierarchy of folders. Storing a large number of media files in a single folder can significantly affect the performance of your project when editing the files in the Media library application.

We recommend storing at most 100 files in a single media library folder. Once you reach the limit, make sure you create a structure that allows you to divide the files into multiple folders. The number of files in one folder directly affects the performance of the Media library user interface.

Additionally, the system's automatic clearing of files from the server-side cache does not work for files stored in an external storage. If you modify a media file, the website may still display the old version until the cache expires (unless you manually clear the application's cache). See also: [Caching files and resources](#)