

# File storage

In many features, like [File Gallery](#), [Trackers](#), [Wiki](#) , etc., it's possible to upload binary files (like .png, .odt, .xls, .doc, .pdf, etc). By default, Tiki stores these files in your database with the rest of the data. If you will have a lot files, or large files, you should change this setting to store files on the file system.

But for other features, storage is not configurable. Please check `tiki-admin_system.php` for your Tiki set up.

## Storing Files in Database or File Directory

There's an important setting for the file galleries: the storage strategy for files. You can choose one of two options to store file gallery files:

- Store files on the database
- Store files on a directory

The differences and settings needed for each option are explained below:

### Storing files on the database

If you decide to store files in the database, then each file data will be stored in a database column (field) as a blob. This has the following advantages and disadvantages:

- You don't need to configure anything else.
- You may have problems if you plan to store large files
- The database can grow a lot making more difficult to make backups/dumps

### Storing files in a directory

If you decide to store files in a directory, you must indicate the path to a directory where you will store files. The directory can be absolute or relative to the Tiki directory for example:

`files/`

`/www/tiki/files/`

`/home/tiki/files/`

`files/uploads/`

can be configured as directories.

Storing files in a directory has the following advantages and disadvantages

- You must configure properly a directory to store files
- You can generally upload larger files. See [Server Check](#) to see the limits.

You must ensure that:

- The configured directory exists and PHP (Apache) can read/write to it
- For security, only Tiki can read or access the contents of that directory. Do that with one of the two following options.
  - Use a directory outside the webserver DocumentRoot (so the users won't be able to list the

directory)

- Use a directory inside the DocumentRoot (relative to the Tiki dir for example) and prevent listing the directory using a .htaccess file or other configuration depending on your webserver.

You may need to ask your ISP/hosting service for an upload strategy and instructions if the ISP/hosting allows uploads (some PHP's have file uploads disabled for unknown reasons, maybe to prevent disk space consuming). Since Tiki can be configured to store files on the database or a directory if there's a way to handle uploads then Tiki can use it 😊 choose the strategy that is best for you.

## Ideal scenario

The best scenario is to work like in the sandbox mode of [ClearOS](#).

To illustrate: Your Tiki instance should be here: `/var/www/virtual/example.org/html/` and for your data-files:

Path	What goes here	Your setting in Tiki admin panel
<code>/var/www/virtual/example.org/tiki-data-files/file-gallery/</code>	<a href="#">File gallery</a>	<code>../tiki-data-files/file-gallery/</code> files
<code>/var/www/virtual/example.org/tiki-data-files/wiki-attachments/</code>	<a href="#">Wiki Attachments</a>	<code>../tiki-data-files/wiki-attachments/</code>
<code>/var/www/virtual/example.org/tiki-data-files/tracker-attachments/</code>	<a href="#">Tracker Attachments</a>	<code>../tiki-data-files/tracker-attachments/</code>
<code>/var/www/virtual/example.org/tiki-data-files/user-files/</code>	<a href="#">User Files</a>	<code>../tiki-data-files/user-files/</code>
<code>/var/www/virtual/example.org/tiki-data-files/image-gallery/</code>	<a href="#">Image gallery</a>	<code>../tiki-data-files/image-gallery/</code> files
<code>/var/www/virtual/example.org/tiki-batch-upload/</code>	<a href="#">Batch Upload</a>	<code>../tiki-batch-upload/</code>

Batch Upload is purposely put in a different place than tiki-data-files because one is long term storage, and the other is just a temporary space.

### Benefits

- Easy to understand
- Everthing related to your site is in one subdirectory
- Data files are outside the web root
- Works with TRIM's `make restore` and `make clone`

## Related links

- [Search within files](#)