Chapter 2 "Installation"
In this package, you will find:

- A biography of the authors of the book
- A preview chapter from the book, Chapter 2 "Installation"
- A synopsis of the book’s content
- Information on where to buy this book

About the Authors

Werner Altmann is the author of the original german handbook for editors and a member of the TYPO3 documentation team. As technical project manager he is in charge of planning and implementation of major TYPO3-based projects.

René Fritz has created parts of the TYPO3 core and has programmed various extensions to the system, including the digital asset management system. He was also among the first TYPO3 users. He works as a freelance technical consultant and developer and continues to be part of the core developer team of TYPO3. He is also an active member of the TYPO3 Association.

Daniel Hinderink is a managing partner of a consulting company specializing in web-technology based information systems for medium to large companies. He is also a long-time associate of the TYPO3 project and provided strategic advice and practical work in research and planning as well as marketing. He is also an active member and currently the vice-chairman of the TYPO3 Association.
In this chapter we will introduce variations and the recommended fields of application for various installation types, and describe the necessary steps to implement them. With the many thousands of installations of TYPO3 already made, you would think that the installation hurdles are not actually set too high. However, the authors of this book can also tell a tale about “the first time” and of some of the problems that might throw the beginner off course.

Finally you will find an overview of some resources available to you that are rich in content, to find answers to any problems you may have.

2.1 Criteria for System and Package Selection

TYPO3 requires just a database and a web server that is PHP-capable; on this basis, details such as the hardware used, the operating system, the database system, and the web server can be selected using various criteria, the most important of which we will explain here.

A basic decision you need to make concerns the operating system. TYPO3 can run on most UNIX-based variants, and on Windows systems. There is no difference in the scope of the two versions in terms of core functionality, although a number of extensions do require UNIX programs. Please consult the documentation for the relevant extension, so that such restrictions can be considered.  

One technical advantage of using UNIX systems is that updates are

---

1 Indexed Search, for example, a powerful search engine in the TYPO3 framework, requires UNIX software to index documents. Various services of the DAM also need this, as do the PDF converter extensions.
2 Installation

considerably faster, which is made possible by so-called “symbolic links”.²
The dissemination of knowledge through the community is also important.
The trend, certainly in terms of numbers, is dominated by the use of Linux
systems, and to this extent support and operating system-dependent new
development for extensions are much more widespread for this system. So
if there are no compelling counter-arguments—such as a predefined
Windows-based infrastructure, or your own (lack of) knowledge—the Linux
variation makes the better choice, from a TYPO3 point of view.

In the following sections you will find further information on hardware and
software selection in terms of the web server, the database system, other
useful software, and finally the package selection of TYPO3 itself.

2.1.1 Hardware

As PHP-based software, TYPO3 at the minimum requires hardware equip-
ment capable of running a web server. Even though this may still be possi-
ble with old 286 and 32 MB RAM machines, these cannot provide a useful
platform for operating a TYPO3 system. The system should have 512 MB
RAM or more in order to provide a solution with adequate performance.

If you want to use your own server, consider the following factors when
sizing your hardware:

Type of Usage

The decisive factor is how the information provided on your website is to
be used: will the website be mainly, or completely, static? Do you want to
operate a portal and do you need dynamic content generation on the web
server? Are you planning an application to be made available to normal
visitors to your site, such as a shop, eCards, or forums? The following basic
rule applies: More functions operated on the web server translate to
increased hardware requirements and expense in separating the live
system from the production system.

Expected Capacity

A number of parameters can be expressed in numbers:

- How many users will use the system at the same time?

² This can now be remedied by the use of Junction, an additional software package,
see soft link 394 945.
How fast (in seconds) should the server deliver a page?
How many pages per hour do you intend delivering?
How many page impressions do you expect per month?
How much data traffic, in MB or GB, do you expect per month?

If you do not know the appropriate values, you will find it difficult to make a well-based decision. If you have any doubts, try to make contact with operators of similar services to collect relevant information. We shall look at three very typical scenarios:

1. **Small to Medium Performance Requirements**
   These include sites that do not economically justify the costs of having their own server, expect no more than five to 10 users in parallel in the back end and 50 in the front end, a page delivery of up to 1.5 seconds, no more than 100 pages per hour, and less than 100,000 page impressions per month. In all, you expect data traffic of less than 5 GB per month. If your profile matches this in certain critical points, you should look around for a good and cheap hosting provider, who might even provide preinstalled TYPO3.

   If you decide on a solution without preinstalled TYPO3, clarify beforehand whether the necessary requirements for installation are fulfilled:
   - At least 100 MB free memory space
   - MySQL database
   - PHP from version 4.3.x with GDLib and Freetype
   - ImageMagick
   - Access via SSH

   A few notes of warning at this point: for a long time now, miraculous abilities are no longer needed to hire out web space. Through cheap offers for dedicated servers, in principle anybody who has a few Euros and a month to spare can set up store as a reseller and hosting provider. Knowledge and ability are not mandatory requirements, and this can sometimes have dramatic consequences.

   The hosting business is subject to the same economic constraints as any other business, and low-level prices are almost certainly linked to one or more of the following:
   1. Poor knowledge of the material, since someone with experience in this business knows that server downtime can cancel out the earnings of months
2 Installation

2. Low personal costs, since the commodity itself is purchased from one of the mass providers (with correspondingly little room for maneuver for your contract partner)

3. Size of provider—due to its large size, it is worth employing experts

Of course, it is perfectly legitimate to search for as cheap an offer as possible. Just take into account the fact that at such low prices you cannot expect an optimal service and protection of values, with respect to installation and content in your CMS.

To proceed carefully, and at least have a basic level of security (and a supplier who is in liquidity and from whom you can seek compensation), you should pay a fair price, probably not below 25 Euro per month. It is here that you may be able to benefit from smaller providers, who can offer an individual service and be a reliable partner with whom you can trust your work.

Irrespective of the price class in which your project is located, a situation without an automatic backup must never occur—make this one of the fundamental criteria on which to base your decision.

2. Medium to Large-Sized Requirements and Intranet

It is a good idea to have your own web server if your CMS application justifies the expense for operating and setting it up. In doing this, the price for renting such a server and/or for co-location or housing will constitute the least of the costs to be expected.

Most studies yielded a ratio of 35% to 65% in the distribution of costs between purchase or rent on the one hand, and maintenance and other running costs on the other.

In the intranet, you have no choice but to run your own server. The key question here is that of sizing. A sound calculation, which also takes into account growth through scaling (two servers are not just 10 percent more work than one server), is very advisable so that you can arrive at a balanced decision.

The generally accepted formula of calculating overall costs during the lifecycle of an investment commodity, the so-called total cost of ownership, is (for this example) calculated as follows:
2.1 Criteria for System and Package Selection

The TCO "Iceberg"
visible costs
+ capital expenditure
+ HW/SW purchase price
+ installation
+ implementation

invisible costs
+ system adoption
+ support
+ administration
+ training
+ maintenance
+ degree of non-usage

= total cost of ownership

This calculation model has become very refined indeed \(^3\), but for demonstration purposes and a rough assessment of the actual costs of an operation, the above list is normally sufficient.

If your calculation indicates making a decision in favor of your own server, the following technical points should be clarified: performance capability of the hardware, its ability to be updated, availability of replacement parts, backup strategy, and security (access restrictions, etc.).

3. Heavy-Duty Operation

In heavy-duty operation, above one million page impressions per month, the standard solution is to combine a number of servers into a cluster. The reason for this lies not only in the spread of the load, but also in its fault tolerance. Depending on the purpose it is used for, the maximum load the server has to cope with can be exceeded even with a small amount of traffic. Also, the task conditions of the application may not allow—for example in a management information system—any downtime. And yet web servers and database servers can also be operated separately and separately scaled, depending on the load to which they are subjected. Scenarios exist both with 10 Apache web servers, each with its own hardware and one MySQL server, as well as ones with 3 Apache servers and 5 MySQL servers.

Whatever the case, special solutions are required here which require

precise planning and implementation, as well as continual maintenance.

2.1.2 Web Server

In theory, TYPO3 can be operated with Apache, IIS, or any other web server that can run PHP. The most common combination, according to numbers of installations, is Apache and PHP. Each particular version and installation type of PHP has, to some extent, an influence on TYPO3, which will be described in the “Server Compatibility” section on TYPO3.org.

2.1.3 Database

The standard database for TYPO3 is MySQL, for a long time the only option. Because the database abstraction that has now been introduced uses MySQL-compatible SQL instead of a meta language, this combination still provides the highest performance, since database queries do not have to be transformed. In addition, many existing extensions use their own database queries, so checks need to be made in individual cases to see whether an extension is fully functional without having to make modifications to other databases.

The decision to choose another RDBM system instead of MySQL should be considered very carefully; ultimately the costs of installation and updates will for the time being be greater than for the standard setup.

An interesting variation on the use of other databases is to combine different systems, so that specific data is kept on your own database system, and this store is made available for use in a specific application in the TYPO3 framework. In this way data can be integrated directly from other database systems, and the disadvantages of replication or synchronization of data stores avoided. The database abstraction is not in the least restricted to RDBM systems—flat files such as XML data can also be addressed via SQL queries.

Below we shall only cover the standard situation with MySQL, as more information for other scenarios is not yet available now. However, the documentation of the Database Abstraction Layer system (DBAL) does provide a number of hints (see soft link).
2.1.4 Other Software

Two software packages are necessary to be able to use the image-processing functions of TYPO3. These functions are entirely optional; TYPO3 still works without graphical processing and without these packages. The first package is GDLibrary, a PHP extension, which itself can be extended by Freetype with functions for representing typefaces. GDLibrary is already included in the standard PHP installation; it will not be discussed separately here.

The second software package, used predominantly in the scaling and generation of image file previews, is ImageMagick. With TYPO3, using an older version of ImageMagick (Version 4.2.9) is recommended, as it has a number of advantages over later versions. 4

TYPO3 can be used with current versions of ImageMagick if the disadvantages in functions for masking, sharpening, and softening of images do not affect your website.

The recommended version 4.2.9 is available for download on the TYPO3.org website under the soft link shown here. Current ImageMagick versions can either be installed with the packet manager of your Linux distribution or downloaded from the ImageMagick homepage. 5

Alternatively you could use GraphicsMagick, an offshoot of the ImageMagick project, which pledges to maintain a more strict line to the API development. 6

4 A corresponding question sent to the development team of ImageMagick, asking why objective disimprovements had been introduced in some areas, yielded no satisfactory response. The e-mail communication, explaining individual disadvantages, can be found under the soft link shown above.

5 http://www.imagemagick.com/

6 http://www.graphicsmagick.org/ : In particular the changing story of the ImageMagick API has stopped many developers from basing their development on this software. TYPO3 went along with this capricious behavior of recent years, and the configuration options reflect the changes made in the list of exceptions for individual ImageMagick versions.
2.1.5 TYPO3 Package Selection

TYPO3 exists in various different packages, depending on the use to which it will be put. The main distinction is initially between the two different operating systems:

All UNIX packages can be found as so-called “tarballs”\(^7\) and have the file ending `.tar.gz`. Windows packages can be found as zip archives, and have the `.zip` file extension.

There is no difference between the actual files contained in each version. The difference between the two versions lies in the fact that the tar.gz-distribution may be smaller, because it does not have double folders, since these are represented by symbolic links instead. In the zip-distribution these folders exist twice, as shown in the screen shot.

The double use of directories (using symbolic links) is essentially due to historical factors. You must bear in mind that symbolic links cannot be set up via an FTP access. If your access to the web server is limited to FTP, choose a zip-distribution.

For some time now it has also been possible, through utility software such as Junction, to set up links in the file system (similar to symbolic links) in NTFS-formatted partitions of Windows.

\(^7\) Tarball is the name given to archives generated with the tar program (and usually compressed with gzip as well). Symbolic links are supported with this file format.
2.1 Criteria for System and Package Selection

The TYPO3 version is always the most up-to-date one for all packages; they differ only in the examples they contain. Generally all tar.gz-packages do not contain the actual core system, which needs to be downloaded additionally as typo3_src followed by the version number. All zip-distributions include the system core.

In essence, the TYPO3 project offers the following different packages for download:

QuickStart
This is the installation for beginners and includes the beginner's tutorial. You should choose this package if you want to make use of the tutorial, since it includes material and defaults for the examples. You can also find the tutorial in the German version under the soft link shown here.

Test Site
The Test Site contains a number of examples of TypoScript templates, menu variations, and extensions such as Shop, address lists, and news. For tests, and to get to know the system on your own by means of examples, this rather old package can be used—as far as the examples are concerned.

Dummy
The Dummy package differs from the Quickstart and Test Site only in its empty database and the absence of example material. It is the standard package for experienced developers who are starting a new project with a clean slate.

The included database dump contains an administrator account, which can be called after installation with the user name admin and the password password.

TYPO3 Source
The TYPO3 source is a tarball that contains all the directories needed for running the base system. Zip-distributions already contain this source package by default.

With the help of symbolic links, a number of TYPO3 installations (websites) can be operated using a single source. Corresponding extensions can be stored in the global extension directory typo3/ext/ or in the corresponding typo3conf/ext/ directory valid for that instance.
2.2 Installing a Test and Training Installation

TYPO3.org provides a whole range of complete packages for various systems. We shall now take a close look at the two classical distributions, the WAMP (Windows Apache MySQL PHP) Installer for Windows by Ingmar Schlecht and the Quickstart package for Linux. There are other packages and installation guides for BigApache, Mac OS X, Debian, Gentoo, Mandrake, and others. A look at the download page and at the documentation matrix at TYPO3.org may also help you to find new ideas for the purpose of your installation.

2.2.1 The WAMP Windows Installer

You will find the WAMP installer under the installer packages on the download page of TYPO3.org. It is very simple to use:

1. Load the installer onto your computer.
2. Open the file by double-clicking on it.
3. After a short time you will see a dialog asking you to accept the GPL. The GPL is a comparatively short license, which you should now read if you want to know about your rights. If you do not agree, this is the temporary end of your TYPO3 career: there is no other license agreement for TYPO3 apart from this one, and you must adhere to its conditions if you do not want to lose the right to use it.
4. Then a message appears that the installer will write all the files to C:\apache. Accept this only if you do not have such a directory, or if you want to delete the data saved there, since all data stored at this location will be irretrievably lost.
5. The installation is now finished.

You will now find a new TYPO3 entry in your start menu with these options:

- Start Apache: Starts the Apache web server
- Start MySQL: Starts the MySQL database server
- Stop MySQL: Ends the MySQL database server
- TYPO3 start servers before: Calls up the TYPO3 front end in a window of your default browser under http://localhost
2.2 Installing a Test and Training Installation

- TYPO3 (Alternative URL): Calls up TYPO3 under http://127.0.0.1

If you now start Apache and MySQL and call up the TYPO3 start page, you will find all the information you need on the front end and back end. The WAMP installer has already carried out the other steps of the installation, and TYPO3 is now ready for tests, and for your own programming attempts.

2.2.2 Linux (et al.) Quick Install

If you already have a web server available that supports MySQL and PHP, you will reach your destination very quickly under Linux and other comparable *NIX-flavours, like BSD, OS X, etc.

If you have SSH access to the web server, you should log in there and download the Quickinstall archive from TYPO3.org, using curl or wget. The addresses are mentioned on the download page of TYPO3.org.

If you only have FTP access on your web server, use the zip-distribution. Transfer all the files onto the web server in the directory from which your website will be provided (typical directory names are .../htdocs/, .../html/, or .../www/).

You must now change some permissions so that certain parts of TYPO3 can be edited. If you do not have any access via SSH, you need another way of changing permissions. Many web servers have graphical tools to edit files, such as Cpanel, Confixx, or Webmin. If in doubt, ask your administrator or provider.

The permissions of the following files must be changed as shown below:

```
chmod 777 typo3/temp/
chmod 777 typo3/ext/
chmod 777 typo3temp/
chmod 777 typo3conf/
chmod 777 typo3conf/ext/
chmod 777 uploads/
chmod 777 fileadmin/
```

Beware: by making all the files world-writeable, security of your system could be compromised by other users with access to your server. You can consult section 2.3.1 for more information on how to set up a safe environment.

You can call up TYPO3 from http://www.yourdomain.de/index.php on your web server.

If you do not have a file called index.html in the directory of your web
server, you can omit index.php. Otherwise we recommend that you rename the file index.html, perhaps to index_alt.html. You can delete it of course, providing you no longer need it.

Before the Installation Tool is called up in the so-called "1-2-3 mode", you will see a warning, which asks you to change the Installation Tool password immediately.

Now enter a user name and password for the MySQL database and send off the form. Then you either create a new database or select an existing one. Please remember that all data contained in it will be deleted!

Import the file quickinstall.sql into the database. You can now log in to the back end with the user name admin and the password password, under http://www.yourdomain.de/typo3. You can log in to the Installation Tool with the password joh316.8 Please make sure you change the Installation Tool password on your first log in!

The basic installation for test purposes is now finished.

2.3 Installation for Productive Use

The installation for productive use is distinguished mainly by selecting the package that only contains the source, without the example data, and takes up a minimum amount of space. In addition, the Installation Tool contains considerably more options than are available in the simplified mode. Not only is it essential to know this to be able to configure the system optimally, the options also provide a number of insights into the structure of TYPO3.

2.3.1 LAMP Installation

As the most widely used operating system for web servers, UNIX/Linux is also a very good choice for using TYPO3. Especially when performing updates, the Linux platform has a clear advantage; there are a number of supporting software packages that are only available under Linux, and that are required for certain additional modules of TYPO3. So it really is worth taking a look at the online documentation for an extension before downloading and installing it. Below, we will assume a standard installation with MySQL.

8 For the curious: Kasper Skårhøj refers here to the following verse of John the Evangelist: “For God so loved the world that he gave his only begotten Son, that whosoever believeth in him should not perish but have everlasting life”.

TYPO3, by Packt Publishing
Depending on the Linux distribution, there may be differences in the installation of Apache/PHP and MySQL. For image scaling and processing, ImageMagick is required. Alternatively you could use GraphicsMagick, an offshoot of ImageMagick. Please consult the documentation of your system or the relevant package manager. Whatever the case, you should ensure that PHP has at least 16 MB of memory available (this is set in the php.ini file) and also allows uploads of sufficient size (configured in php.ini and Apache).

For the installation of TYPO3 you first require an empty MySQL database with user name and password, as well as one of the above described packages from TYPO3.org.

There you can download the source and the so-called dummy distribution, which contains symbolic links and a configuration directory, and which will save you a bit of typing work on the command line.

The best way is to open the download page for the desired package in the browser. At the same time, open a command line application (bash, term, putty) and make a connection via SSH to the web server:

```
user@linux:~> ssh user@domain.de
```

Change to the directory of your web server, which lies one level above the website directory:

```
user@domain:~> cd /srv/www
```

Download the current version of the dummy and the source package. The correct address and the name of the file can be found on the packages page at TYPO3.org (see soft link):

```
user@domain:/srv/www> wget > http://typo3.sunsite.dk/unix-archives/3.6.2/dummy/dummy-3.7.0.tar.gz
```

and

```
user@domain:/srv/www> wget > http://typo3.sunsite.dk/unix-archives/3.7.0/typo3_src/typo3_src-3.7.0.tar.gz
```

Here, `srv/www/` stands for the directory in which the directory of the website (in our example, `htdocs/`) will be located. If necessary, consult the configuration file for your web server to find out the correct path.

After downloading to the `srv/www/` directory, unpack the archive with the commands:

```
user@domain/srv/www> tar xzf typo3_src-3.7.0.tar.gz
```
2 Installation

and

user@domain{/srv/www> tar xzf dummy-3.7.0.tar.gz

Now move the files in the dummy-3.7.0 directory, with the command

user@domain/srv/www> mv dummy-3.7.0/* htdocs/

to the htdocs directory, or the directory from which your website should be issued.

You can now delete the archive and the empty folder:

user@domain/srv/www> rm -r dummy-3.7.0
user@domain/srv/www> rm dummy-3.7.0-3.tar.gz

After this, if you call your htdocs directory with

user@domain/srv/www> ls -al htdocs/

it should appear as follows:

total 648
-drwxr-xr-x 6 user group 204 11 Mar 22:00 ..
-rw------- 1 user group 621 24 Jul 2003 Changelog
-rw------- 1 user group 4815 24 Jul 2003 INSTALL.txt
-rw------- 1 user group 608 24 Jul 2003 PACKAGE.txt
-rw------- 1 user group 8087 24 Jul 2003 README.txt
-rw-r----- 1 user group 134 27 Sep 2003 _.htaccess
-rw-r----- 1 user group 46 7 Sep 1999 clear.gif
drwxr-x--- 4 user group 136 12 Dec 2003 fileadmin
lrwxr-xr-x 1 user group 18 11 Mar 22:05 index.php
   -> tslib/index_ts.php
lrwxr-xr-x 1 user group 12 11 Mar 22:05 media ->
   tslib/media/
lrwxr-xr-x 1 user group 17 11 Mar 22:05 showpic.php
   -> tslib/showpic.php
lrwxr-xr-x 1 user group 16 11 Mar 22:05 t3lib ->
   typo3_src/t3lib/
lrwxr-xr-x 1 user group 16 11 Mar 22:05 tslib ->
   typo3_src/tslib/
lrwxr-xr-x 1 user group 16 11 Mar 22:05 typo3 ->
   typo3_src/typo3/
lrwxr-xr-x 1 user group 18 11 Mar 22:05 typo3_src
   -> ../typo3_src-3.7.0
drwxr-x--- 6 user group 204 16 Feb 2003 typo3config
drwxr-x--- 2 user group 68 12 Dec 2002 typo3temp
drwxr-x--- 7 user group 238 12 Dec 2002 uploads
Run the following commands one after another to make the directories writable for the web server:

```bash
chmod 777 typo3/temp
chmod 777 typo3/ext
chmod 777 typo3temp
chmod 777 typo3conf
chmod 777 typo3conf/ext
chmod 777 uploads
chmod 777 fileadmin
```

Assigning 777 permissions is not without risk, however, since this gives access to all users of the system. It would be better to set the permissions to 770, if the user can be set to the name of the webmaster, and if the group can be set to the group name under which the web server is running. But this depends on the administrative options and the permissions on the web server available to you; 777 functions in all cases.

Now release the security lock in the Installation Tool by opening the following file in an editor (in our example, `vi`, available on most Linux platforms):

```bash
user@domain/srv/www> vi typo3/install/index.php
```

At the beginning, change the following line:

```perl
die("In the main source distribution of TYPO3, the install script is disabled by a die() function call.<BR>Open the file typo3/install/index.php and remove/out-comment the line that outputs this message!";)
```

If you are using `vi` (or `vim`), enter `dd` followed by `ZZ`. You have now deleted the line, saved the file and exited `vi`. Alternatively you can comment out the line by entering `//`, if you want to reactivate the lock later for reasons of security. To do this, enter `i` to reach insert mode. After entering `//` at the beginning of the line, exit the insert mode by pressing `Esc`, and then `:wq!`, to save your changes and close the editor.

In the browser you can now call the Installation tool, by entering

```
```

### 2.3.2 WAMP Installation


After installing Apache, MySQL and PHP, you should install ImageMagick.
2 Installation

This is also Open Source software; under the soft link shown here you will find a corresponding version of ImageMagick, optimized for use with TYPO3. You need Windows binaries (also called executables) for the installation if you do not want to compile the software yourself.

Install ImageMagick on your computer and then you can continue with the TYPO3 installation.

For this, download the dummy package in the zip distribution from the download page of TYPO3.org to your computer. Unpack the zip archive in the directory from which Apache will publish the website. Normally this is the following directory:

C:\Programs\apache\htdocs\n
In the next step the user permissions for the web server must be set, so that the user under which Apache is running can read and edit all the necessary areas. You must be able to edit the following directories:

- typo3\temp\n- typo3\ext\n- typo3temp\n- typo3conf\n- typo3conf\ext\n- uploads\n- fileadmin\n
In the file typo3\install\index.php the line

```
< concentrated ?>In the main source distribution of TYPO3, the install script is disabled by a die() function call.<br>Open the file typo3/install/index.php and remove/out-comment the line that outputs this message!">}
```

must be deleted or comment out by placing // at the beginning of the line.

Let's move on to the Installation Tool.

2.3.3 WIIS Installation

The Windows Internet Information Server installation consists of seven steps:

1. System preparation: To prepare the system, you should set up a separate partition for the web server, so that the permissions you have to set do not influence the permissions on your system partition, which could otherwise potentially jeopardize your system.

2. MySQL installation: Load a current MySQL version from
2.3 Installation for Productive Use

http://www.mysql.com/ to your Windows server, unpack the installation file, and follow the instructions. When the installation is complete you will see a graphical interface with the program WinMySQLadmin, with which you can set up users and databases. Create an empty database and a user for this database.

3. ImageMagick installation (optional): The ImageMagick software is also Open Source; and under the soft link shown here you will find a version optimized for use with TYPO3. You need Windows binaries, also called executables, if you don't want to compile the software yourself.

4. PHP installation: After the PHP installation you should check a number of settings in the file php.ini and adjust them where necessary. The php.ini file contains the PHP configuration parameters. The most important setting for the operation of TYPO3 is:

   memory_limit=8M

   This should be increased to at least 16M (16 MB). The Installation Tool checks a number of other parameters, but in general these have been correctly set in the default configuration.

5. IIS configuration: The IIS configuration contains no special features in terms of TYPO3; the settings can either be taken over automatically by the PHP installer or carried out manually, as described in the installation guide for PHP. For reasons of performance, it is generally highly recommended that you operate PHP in ISAPI mode.

6. Unpack TYPO3: The selected zip package is unpacked in the target directory, usually F:\inetpub\wwwroot\, although the drive name may vary, of course.

7. Assign NTFS permissions: Finally, permissions must be given to both users under whom IIS is operated. These users are called IUSR and IWAM, to which their server name is added. The user IUSR_MACHINENAME needs read permissions for the entire file system, in order to carry out the PHP functions file_exists(), is_file(), etc., and not just for the directories in which TYPO3 is installed. This user must also be assigned read and execute permission for the program cmd.exe in order to run ImageMagick, where appropriate, and must also have read permission for the file php.ini. In the web server directory the following permissions must be assigned to the user IUSR_MACHINENAME:
The Installation Tool

The Installation Tool is essentially a graphical interface for editing TYPO3 system settings saved in the file `localconf.php`, in the directory `typo3conf/`. This file, as well as the entire `localconf` directory must therefore also be writable for the web server.

Let's try to understand the functioning of the configuration system. At run time, TYPO3 generates cache files in the `typo3conf` directory, on whose configuration parameters TYPO3 operates. So if configuration data is modified, these cache files must be deleted for the changes to take effect. Normally this is done automatically, but if the sources are replaced, especially when downgrading the version, TYPO3 may not become active by itself, so the cache files have to be deleted manually. These files have names such as `temp_CACHED_ps2268_ext_localconf.php`.

By adding `/typo3/install/` on to the name of your domain, you call up the Installation Tool: `http://www.yourdomain.de/typo3/install/`.

To be able to use the Installation Tool, a locking function must be removed from the script `typo3/install/index.php`. This has already been described in the section on the LAMP and WAMP installations.

The default password for the Installation Tool is `joh316` and should be changed immediately after the first login. After this, the installation can be started, by entering various information. You need to enter the following:

1. User name and password as well as host name and, if appropriate, the name of an already created database that

<table>
<thead>
<tr>
<th>TYPO3</th>
<th>Permission</th>
</tr>
</thead>
<tbody>
<tr>
<td>web server directory (normally: Drive:\inetpub\wwwroot)</td>
<td>Read</td>
</tr>
<tr>
<td>fileadmin\</td>
<td>Edit (with subdirectories)</td>
</tr>
<tr>
<td>typo3temp\</td>
<td>Edit (with subdirectories)</td>
</tr>
<tr>
<td>uploads\</td>
<td>Edit (with subdirectories)</td>
</tr>
<tr>
<td>typo3ext\</td>
<td>Edit (with subdirectories)</td>
</tr>
<tr>
<td>typo3conf\</td>
<td>Edit (with subdirectories)</td>
</tr>
<tr>
<td>C:\PHP\uploadtemp\</td>
<td>Edit</td>
</tr>
<tr>
<td>ImageMagick</td>
<td>Read and Execute</td>
</tr>
<tr>
<td>C:\Windows\system32\cmd.exe</td>
<td>Read and Execute</td>
</tr>
</tbody>
</table>
2.4 The Installation Tool

TYPO3 should use.

2. The path to the directory where ImageMagick is installed on your system. With the command

```
user@linux:~> locate identify
```

you can quickly find this out on most Linux distributions.

The Installation Tool is divided into the following areas, where the first three have to be edited to some extent when it is installed. The other views are for system maintenance.

### 2.4.1 Basic Configuration

The tab for basic configuration checks the permissions of the directories that need to be writable for TYPO3, and also checks the PHP configuration in the `php.ini` file. Any problems that might prevent the installation from being performed are shown here, with corresponding warnings.

![Directory Configuration](image1.png)

Here you must also specify the access data for your database. After entering the user name, password, and host name (usually "localhost", if MySQL is installed on your server), the form must be submitted once with the `update configuration` command in order to select a database or, if your user has the appropriate permissions (`create`), to create a new database.

![Database Access Data](image2.png)
2 Installation

Next we will discuss the settings for generating images. If you have installed ImageMagick, you should specify the path to your installation here. TYPO3 searches automatically in the default directory for ImageMagick and also determines independently whether you have compiled the GDLib with Freetype into PHP. With current Freetype versions, the text in the test picture might extend beyond the edges. If this problem persists, we will fix it at a later stage (see section 2.4.4).

If you submit the form, the Installation Tool will write the corresponding values to the `localconf.php` file.

2.4.2 Database Analyzer

The script for database analysis enables you to both edit and update an existing database as well as to fill it for the first time with a database definition and standard contents. A basic database definition is included in the dummy package. The corresponding SQL file is in the "localconf" directory, `typo3conf/`, and is displayed automatically in the Installation Tool.

With the **Import** option, the database is imported. After clicking on this option, you will receive a security query, where you should select the option to read in all data, and submit the form with **Write to database**.
Depending on the performance provided by your web server, this processing may take a while. In the next step you will see a list of all database tables, with the warning that they already exist. If this is not the case, and a number of tables are displayed as not yet existing, then the import process has failed, and you should try it again. Initialization of the database is now finished. An administrator account is then set up with the user name admin and with the password password.

<table>
<thead>
<tr>
<th>Import SQL dump:</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Import SQL dump" /></td>
</tr>
</tbody>
</table>

Figure 2.6: Importing the database

Figure 2.7: Message of the Database Analyzer after successful import of the SQL file (attention: your display may contain different tables!)
2 Installation

An alternative method that does not need the database.sql file from the dummy package can be used to initialize the database—select the COMPARE option. A list of all required database tables is shown. These have been pre-selected. Press the confirm button in the form to create the database tables. Then select the Create "admin" user function and set up an administrator account.

2.4.3 Image Processing

This script is a collection of test outputs to test the function of ImageMagick and the GDLib and Freetype. It has no influence on the configuration of your system, and is therefore not discussed further here.

2.4.4 All Configuration

The All Configuration script contains entry fields with short explanations on all configuration options of TYPO3, which are named in the TYPO3_CONF_VARS array. In general, only a few entries are important. If the text in the previously introduced Freetype test extends beyond the edges, adjust the resolution for Freetype and increase it from 72 to 96 dpi.

Enter this value in the field \[GFX\][TTFdpi]; you will find it as the final part of section \[GFX\], before the \[SYS\] section.

Further entries are not normally necessary here.

2.4.5 typo3temp/

Here you are shown a statistical overview of the files stored in the typo3temp directory, and you can also delete them from here. This is an important function for maintaining the installation. The directory is used by TYPO3 to store image files shown in the front end. The images are freshly generated each time the page cache is deleted, while the old ones remain in this folder. For large websites with large amounts of image material, many MBs can quickly accumulate here. If the images are deleted from this directory, although the files are still referenced from the pages, the image is newly generated, as long as you emptied the page cache after deleting the files. You can also take a look in the back end of your TYPO3 installation, in the Tools | Check DB module under Missing Relations, to see how many files without a database relation are located in this directory.
2.4.6 phpinfo()

This area calls up the PHP default function `phpinfo()` and displays all the essential parameters of the PHP installation. At the top you will also find a summary of the most important system parameters of TYPO3, which you should include in your e-mail in case of questions to the Install list, so that all essential information to search for the error is provided.

2.4.7 Edit Files in typo3conf/

This option shows a list of the files in the `typo3conf/` directory. You can edit each of the files; just click on the name of the file to reach the editing dialog. This can be a very useful option for experienced TYPO3 users, to set configuration values by hand, for example, if one of the installation scripts has failed to work. You have the option of converting all line breaks from the Windows format into Linux line breaks, and automatically creating a backup copy of the file you are editing. This last function is highly recommended for configuration attempts in the file `localconf.php`, so that you can easily restore the former configuration.

2.4.8 About

You already know this page: it is the first page you are shown when the program starts. It provides you with the option of changing the Installation Tool password. If you have not already done this, do it now.

2.5 Configuration Options in TYPO3_CONF_VARS

Apart from the basic configuration, a whole range of settings can be influenced by the All Configuration script. Here is a list of all the parameters, together with descriptions, some of which are contained in the script itself in abbreviated form.

2.5.1 [GFX]:$TYPO3_CONF_VARS

The GFX area (for “Graphics”) contains all the configuration options for image processing in TYPO3. IM is an abbreviation for ImageMagick, GD stands for the GDLibrary.
2 Installation

[image_processing]
Boolean value (0,1). Switches image processing in TYPO3 on or off.

Example: [image_processing] = 1

[thumbnails]
Boolean value (0,1). Switches display of thumbnails in the back end on or off.

Example: [thumbnails] = 1

[thumbnails_png]
Bits. Bit 0: the entry 0 creates thumbnails in GIF format, 1 in PNG format. Bit 1: the entry 2 specifies that all files displayed in JPG format are also displayed as GIF, or with 3, as PNG.

Example: [thumbnails_png] = 0

[noIconProc]
Boolean value (0,1). If switched on (1), icons are not processed dynamically in the back end with overlays (e.g. start time and end time, hidden), but must be available on the server. This can be useful to obtain a functioning back end, even if overlays are not correctly performed by the installed ImageMagick software. This option should only be switched off if the server provides all image-processing options.

Example: [noIconProc] = 1

[gif_compress]
Boolean value (0,1). This activates the t3lib_div::gif_compress() function, which recompresses generated GIF files if they do not use any compression or only use RLE compression (Run Length Encoding). See: [im_path_lzw].

Example: [gif_compress] = 1

[imagefile_ext]
You can specify here, in a comma-separated list, which file extensions should be interpreted by TYPO3 as images. The list should be restricted to gif,png,jpeg,jpg if IM is not available. The file extensions are written in lower case, without spaces.

Example: [imagefile_ext] = gif,jpg,jpeg,tif,bmp,pcx,tga,png,pdf,ai
2.5 Configuration Options in TYPO3_CONF_VARS

[gdlib]
Boolean value (0,1). This setting enables the use of the GDLibrary by TYPO3.
Example: [gdlib] = 1

[gdlib_png]
Boolean value (0,1). Entering 1 specifies that GD will generate only PNG files instead of GIF files. However, older browsers cannot display PNG files. Even the current versions of the IE do not support all PNG features, such as transparencies.
Example: [gdlib_png] = 0

[gdlib_2]
Boolean value (0,1). This value should be set to 1 if your web server uses GDLib 2.0.1 or higher, otherwise problems may occur.
Example: [gdlib_2] = 0

[im]
Boolean value (0,1). Enables the use of ImageMagick (IM) by TYPO3.
Example: [im] = 1

[im_path]
Here the path in which the IM programs convert, combine, and identify are located in the file system of the web server is specified.
Example: [im_path] = /usr/local/bin/

[im_path_lzw]
Path details to an IM version whose convert command can implement LZW compression. This compression belongs to a Unisys patent and was supported only temporarily by ImageMagick. The recommended version 4.2.9 of IM can be equipped with LZW compression; in this case the field is left blank and [gif_compress] is switched off. It can be expected that future versions of various software will support LZW compression, since the patent on it has expired.
Example: [im_path_lzw] = /usr/local/typo3sh/bin/
2 Installation

[im_version_5]

Boolean value (0,1). This value must be set to 1 if you are using an IM version upwards of 5.x.

Example: [im_version_5] = 0

[im_negate_mask]

Boolean value (0,1). From version 5.1, images must be inserted before they are overlaid with a mask.

Example: [im_negate_mask] = 0

[im_invMaskState]

Boolean value (0,1). Since version 5.4.3+, the situation described in the last item has been reversed, and TYPO3 can be configured so that the global [im_version_5] setting does not require any conversion of the image files to be masked, as was necessary until this version. 9

Example: [im_invMaskState] = 0

[im_no_effects]

Boolean value (0,1). Effects with ImageMagick work well in the recommended version 4.2.9. In newer versions they have become much slower, and can be switched off with this setting. In the most recent IM versions, effects perform much better again, but the syntax of the API has again been changed—no comment.

Example: [im_no_effects] = 0

[im_v5effects]

Integer (-1,0,1). 0 = switched off. -1 = do not focus sharpen images by default. 1 = All; blurring and sharpening are enabled and the setting [im_no_effects] is overridden.

Example: [im_v5effects] = 0

[im_mask_temp_ext_gif]

Boolean value (0,1). This should be enabled when using IM

---

9 This and many other developments led to the splitting of the GraphicsMagic project, aimed at creating a more reliable interface. Kasper Skårhøj’s original comment at this point is obviously a heartfelt one: “Halleluja for ImageMagick - have I ever regretted using that package...”
2.5 Configuration Options in TYPO3_CONF_VARS

versions 5+. The tslib_cObj class by default uses the PNG format, because generating is faster and puts lesser load on the CPU. But since some IM versions above 5 do not correctly support the PNG format, this can be suppressed here.

Example: [im_mask_temp_ext_gif] = 0

[im_mask_temp_ext_noloss]
String. While images are being masked, the temporary files should be stored in a loss-free format, of course. ImageMagick’s own format, named miff, is ideal for this. Unfortunately version 5.4.9 is not in a position to generate its own file format, so that this function became necessary. If problems occur with masked files, TIF, PNG, or JPG can be used instead.

Example: [im_mask_temp_ext_noloss] = miff

[im_noScaleUp]
Boolean value (0,1). If enabled, images are not enlarged.

Example: [im_noScaleUp] = 0

[im_combine_filename]
String. More recent IM versions have renamed the combine command to composite; the proper name can be given here.

Example: [im_combine_filename] = combine

[im_noFramePrepended]
Boolean value (0,1). Some image formats such as GIF or TIF allow several images to be saved in one file. ImageMagick provides an option to work only with the first image, which in general increases working speed. Unfortunately, some IM versions contain an error that causes the IM to simply ignore these images. In such a case, this option must be enabled.

Example: [im_noFramePrepended] = 0

[enable_typo3temp_db_tracking]
Boolean value (0,1). Here it is specified whether all files in typo3temp/ are recorded in a database table. This prevents files from being doubly rendered, since the relation between the temporarily rendered output file and the source file is noted here. In addition, in the Tools | Database check module in the back end and in the Installation Tool, you can find out how many old files there are in the temp/ directory.

Example: [enable_typo3temp_db_tracking] = 0
2 Installation

[TTFLocaleConv]
String. Up to version 3.6.0 you could specify here, using the recode designations, in what encoding TrueType functions were generated. Since version 3.6.0, the output has always been in UTF-8.

Example: [TTFLocaleConv] =

[TTFdpi]
Integer. This important option is for setting the resolution in dpi (dots per inch) in which the existing Freetype system on your server works. Since version 2 this has been 96 instead of 72 dpi; this causes fonts to be displayed too large if the value is not changed to 96 dpi for a more recent version.

Example: [TTFdpi] = 96

2.5.2 [SYS]:$TYPO3_CONF_VARS["SYS"]
This section describes configuration options involving system behavior in the back end and front end.

[textfile_ext]
You can specify here, via file extensions, which file types may be edited in the back end.

Example: [textfile_ext] = txt,html,htm,css,inc,php,php3,tmpl,js,sql

[contentTable]
With this option you can specify what the page content table should be called. The default value is tt_content.

Example: [contentTable] = tt_content

[sitename]
This is the name of the installation and is shown at the top of the page tree next to the globe. The name can also be set directly in the Basic Configuration view of the Installation Tool.

Example: [sitename] = BT3 Enterprise

[ddmmyy]
Format of the date display—corresponds to the notation of the PHP function, date().

Example: [ddmmyy] = d.m.y
2.5 Configuration Options in TYPO3_CONF_VARS

[hhmm]
Format of the hour display—corresponds to the notation of the PHP function, date().
Example: [hhmm] = H:i

[encryptionKey]
Here you can specify a random string to be used in the creation of hash values for encryption in the context menu, the Direct Mail Module, and at other points in the system; this helps to increase security.
Example: [encryptionKey] = Haaken Flip

[doNotCheckReferer]
Boolean value (0,1). With this option, active checking in the back end to see if the currently accessing host is identical to the Referring Host can be switched off. If this value is set to 1, checking is switched off. This can be useful when accessing via proxy servers, which prevent the correct output of the HTTP_REFERER variable.
Example: [doNotCheckReferer] = 0

[recursiveDomainSearch]
Boolean value (0,1). If enabled (1), TYPO3 searches for a suitable entry, if access is attempted to a non-existent domain by recursively deleting parts of the domain name to find a suitable match.
Example: [recursiveDomainSearch] = 0

[T3InstID]
This option has not yet been used. The intention was to create a unique identity designation with which each installation could identify itself when accessing the Extension Repository. This was intended for statistical purposes, but until now has not been put into practice.
Example: [T3InstID] = N/A

[devIPmask]
Comma-separated IP addresses. This important option defines a list of IP addresses from which error output is displayed in the front end. The Debug() function uses these entries as a filter. Without an entry, no access is allowed, * allows all hosts. * as a wild card can also be used for any part of the IP; thus
2 Installation

192.168.*.* allows all IP addresses which begin with 192.168.*.

Example: [devIPmask] = 192.168.*,127.0.0.1

[curlUse]

Boolean value (0,1). The getUrl function used by the system can be used by entering 1 curl instead of fopen(), so that you can work with proxies (which is not possible with fopen()). Curl must be available in your PHP installation.

Example: [SYS][curlUse] = 0

[curlProxyServer]

URL. Curl must be given the address of the proxy server in the form http://proxy:port/.


[curlProxyTunnel]

Boolean value (0,1). For some security systems it is necessary to choose a "tunneled" connection through the proxy. 1 instructs Curl to do this.

Example: [curlProxyTunnel] = 0

[curlProxyUserPass]

String. User name and password for access to the proxy server should be entered here, if appropriate, in the notation User name:Password.

Example: [curlProxyUserPass] = Leeloo:Multipass

[form_enctype]

String. The encryption type of most forms in TYPO3 can be globally adjusted here. multipart/form-data is set by default, to allow file uploads. If file uploads are not permitted in your PHP installation, data created with this form time will not be transferred. The encryption type can be changed accordingly to application/x-www-form-urlencoded.

Example: [form_enctype] = multipart/form-data

[loginCopyrightWarrantyProvider]

String. The GPL excludes all claims of warranty from the author of the software. If you want or need to assume the warranty for the function towards a customer, you can enter your name here, so
2.5 Configuration Options in TYPO3_CONF_VARS

that it can be displayed in the login page. An Internet address (URL) must also be given in the next step.
Example: [loginCopyrightWarrantyProvider] = VEB Optimism

[loginCopyrightWarrantyURL]
String. A URL is specified, in the form http://www.yourdomain.de/; this is automatically placed as a link to the names given in the previous option.
Example: [loginCopyrightWarrantyURL] = http://www.veb-optimism.de/

[loginCopyrightShowVersion]
Boolean value (0,1). If 1 is entered, the login page shows the TYPO3 version.
Example: [loginCopyrightShowVersion] = 0

[binPath]
String. Comma-separated list: here you can enter a list of absolute system paths where a search is made for external programs. This is used by the DAM for services, for example.
Example: [binPath] = /usr/bin/

[t3lib_cs_convMethod]
String. In the t3lib_cs class, entering one of the following values specifies with which tool character sets are converted: iconv or recode are external programs; the default is TYPO3’s own PHP code. The external programs are significantly faster, but the SGML fallback HTML coding of special characters is not supported.
Example: [t3lib_cs_convMethod] = recode

[t3lib_cs_utils]
String. With this entry you can specify that instead of TYPO3’s own code for processing character sets, the PHP module mbstring should be used, which is much quicker. The only possible value, in contrast to what is specified in the English description in the Installation Tool, is mbstring; if nothing is entered, TYPO3’s own code is used.
Example: [t3lib_cs_utils] = mbstring
2.5.3 [EXT]: $TYPO3_CONF_VARS["EXT"]

This section describes various options for configuring the Extension Manager and related system parts.

[noEdit]

Boolean value (0,1). 1 prevents files in the Extension Manager from being edited. For developers, the value should be set to 0 so that changes can be implemented directly here.

Example: [noEdit] = 1

[allowGlobalInstall]

Boolean value (0,1). With 1 global extensions can be installed in the typo3/ext/ directory via the Extension Manager, updates can be implemented, and extensions deleted from that directory. This option is important if several instances are to be operated on one installation, and if administrators are not to edit global extensions of individual instances.

Example: [allowGlobalInstall] = 0

[allowLocalInstall]

Boolean value (0,1). 1 enables the administration of local extensions in the typo3conf/ext/ directory of the instance in question, via the Extension Manager.

Example: [allowLocalInstall] = 1

[em_devVerUpdate]

Boolean value (0,1). If 1 is entered, developer versions of extensions are marked in red in the Extension Manager if updates are available.

Example: [em_devVerUpdate] = 0

[em_alwaysGetOOManual]

Boolean value (0,1). With the value 1 you can specify that the documentation included in extensions, available in OpenOffice format, is always downloaded to the server.

Example: [em_alwaysGetOOManual] = 0

[em_systemInstall]

Boolean value (0,1). With the value 1 the Extension Manager is allowed to install extensions to the sysExt/ directory. This is needed to be able to perform updates to the cms and lang system
2.5 Configuration Options in \texttt{TYPO3\_CONF\_VARS}

extensions with the EM.

Example: \texttt{[em\_systemInstall] = 0}

\textbf{\texttt{[requiredExt]}}

String. Comma-separated list: here you can specify extensions that cannot be disabled by the Extension Manager. Normally these are at least \texttt{cms} and \texttt{lang}, which form the core of the system.

Example: \texttt{[requiredExt] = cms,lang}

\textbf{\texttt{[extCache]}}

Integer (0,1,2,3). 0 causes the extension scripts \texttt{ext\_localconf.php} and \texttt{ext\_tables.php} not to be stored in the cache, but to be read through each time the page is requested—interesting when developing extensions, as it saves you from deleting the cache in the back end.

The default is 1: the scripts are saved in files of the form \texttt{typo3conf/temp\_CACHED\_[sitePathHash]*}, reducing the load on the server and increasing performance—a useful setting for live operation.

The value 2 has the effect that the file name of the cache file contains a hash value based on the \texttt{[extList]} string. 3 means that the names of the cache files do not contain any hash value.

Example: \texttt{[extCache] = 1}

\textbf{\texttt{[extList]}}

String. Comma-separated list: extensions installed by the EM are entered in this list. If you install a defective extension that makes working with the system impossible, you must edit the file \texttt{typo3conf/localconf.php} to remove this extension and delete the key from this list. In addition the \texttt{typo3conf/temp\_CACHED\_*} files must be deleted for the system to be operational again.

Example: \texttt{[extList] = tsconfig\_help,context\_help, extra\_page\_cm\_options,...}
2.5.4 [BE]: $TYPO3_CONF_VARS['BE']

This section involves the parameters for back end configuration.

[unzip_path]
Path details on the decompression program, unzip.
Example: [unzip_path] = /usr/bin/

[diff_path]
Path to the command line application diff, used to compare files. A Windows version of diff can be downloaded from http://unxutils.sourceforge.net/.
Example: [diff_path] = diff

[fileadminDir]
Path details of the Fileadmin directory; this must be relative to the path of the website from the constant, PATH_site.
Example: [fileadminDir] = fileadmin/

[RTE_imageStorageDir]
Path to where files of the Rich Text Editor are saved.
Example: [RTE_imageStorageDir] = uploads/

[staticFileEditPath]
Path to the directory where so-called static files can be stored and edited. Database fields can be configured in such a way in the $TCA that in reality they are stored in a file. The extension sys_staticfile_edit is one application of this.
Example: [staticFileEditPath] = fileadmin/static/

[lockRootPath]
String. Specifies the first part of the path to [userHomePath] and to [groupHomePath]. Please note that the first parts of [userHomePath] and [groupHomePath] must match [lockRootPath]. This is also used to check if a path is also allowed outside the PATH_site constants. This can be changed if, for example, data processing is to be allowed in a level above the web directory.
Example: [lockRootPath] = /srv/www/archiv/

[userHomePath]
String. Path to a directory where the back end user should have
his or her own home directory. If the path is entered here, TYPO3 automatically creates its own directory for each user, for example if the value /home/typo3/users/ is entered, then a directory with the path /home/typo3/users/43_cameronfrye/ is created for the user cameronfrye with the uid 43.

Example: [userHomePath] = /srv/www/htdocs/typo3/users/

[groupHomePath]

String. In the same way as for users, a separate directory can be created automatically for each group.

Example: [groupHomePath] = /srv/www/htdocs/typo3/group/

[userUploadDir]

String. A pre-configurable suffix that is added to user directories. If the user directory matches the user name and the uid, that is, 43_cameronfrye/, and the value set here is /250GT, a directory with the name 43_cameronfrye/250GT/ is made available to the user.

Example: [userUploadDir] = /250GT

[fileCreateMask]

In accordance with the UNIX umask syntax, you can define here what access permissions should be given to files created by TYPO3 in the file system.

Example: [fileCreateMask] = 0644

[folderCreateMask]

Is the same as the above setting, but refers to folders.

Example: [folderCreateMask] = 0755

[warning_email_addr]

E-mail address to which a warning message is sent if four failed login attempts to the back end have taken place within one hour.

Example: [warning_email_addr] = ronald@evilempire.com

[warning_mode]

Integer (1,2). With 1, the address specified above always receives a message if a user has logged in to the back end. If 2 is set, the address only receives a warning message if an administrator logs in.

Example: [warning_mode] = 2
2 Installation

[ipmaskList]

String. IP addresses can be specified here for which access to the back end is granted exclusively, so that users from other addresses basically have no access at all. It is possible to use * as a wild card.

Example: [IPmaskList] = 192.168.1.*

[adminOnly]

Boolean value (0,1). If 1 is specified, only administrators may log into the back end. 0 allows access to all users. This can be implemented to exclude users from the system during maintenance work and updates.

Example: [adminOnly] = 0

[lockBeUserToDBmounts]

Boolean value (0,1). 1 is entered here by default so that users only have access to their own Pagemount. This can be disabled with 0. Such an application scenario is very improbable, however.

Example: [lockBeUserToDBmounts] = 1

[lockSSL]

Integer (0,1,2). With both entries, TYPO3 is instructed to make the back end available only via an SSL connection. 2 causes access to http://your-domain.de/typo3 to be automatically redirected to https://your-domain.de/typo3.

Example: [lockSSL] = 0

[disable_exec_function]

Boolean value (0,1). On Windows systems, it is possible that the use of the PHP function exec() needs to be suppressed. This is done by entering 1. For ImageMagick this can be achieved by switching off all graphic functions: [GFX] [im]=0.

Example: [disable_exec_function] = 0

[usePHPFileFunctions]

Boolean value (0,1). If PHP is operated in safe_mode, it is possible, even under UNIX, that all file functions with standard PHP functions have to be used together with exec() instead of with external commands. This is changed accordingly by entering 1.

Example: [usePHPFileFunctions] = 1
2.5 Configuration Options in TYPO3_CONF_VARS

[compressionLevel]
Integer (1-9). Requires zlib in PHP. Compression using gzip is
set with a value on a scale from 1-9. Compressed pages use less
bandwidth, but system load increases with higher compression
rates. 0 stands for no compression, 9 for maximum compression.
Alternatively you can specify TRUE, whereby compression is
adjusted dynamically, depending on the system load (only under
Linux and FreeBSD). Compression can alternatively be config-
ured in Apache.

Example: [compressionLevel] = 0

[MaxFileSize]
Integer. The maximum file size which can be edited by TYPO3 can
be set here. The setting is only relevant in the context of the file
size defined for PHP in the file PHP.ini.

Example: [maxFileSize] = 10000

[RTEenabled]
Boolean value (0,1). With this option the Rich Text Editor can be
switched on (1) and off (0) globally, independent of the settings
in the back end.

Example: [RTEenabled] = 1

[forceCharset]
String. Character set is normally used in accordance with the
language set in the back end for the respective user. The charac-
ter set can be defined here for all users. The options can be seen
in the character set tables and the t3lib/csconvtbl/ directory.
utf-8 can be used, for example, for Unicode coding. The character
sets must be specified in lower case.

Example: [forceCharset] = iso{-8859-8}

[InstallToolPassword]
String. This is the md5 hash value of the password for the
Installation Tool. To lock the access, do not enter a value. It is
advisable to additionally protect the directory for the Installation
Tool, typo3/install/, with a password query through a
.htaccess file.

Example: [installToolPassword] =
e1c102cf0300bf73e47018f5bd7766e5
2 Installation

[trackBeUser]
Boolean value (0,1). With the value 1, TYPO3 is instructed to log every call of a script in the back end, in the table sys_track-beuser. To evaluate this, the extension beuser_tracking is needed.
Example: [trackBeUser] = 0

[defaultUserTSconfig]
String. TypoScript: It is possible to define default TypoScript settings for all back end users here (see Chapter 4.8).
Example: [defaultUserTSconfig] = admPanel.enable= 1

[defaultPageTSconfig]
String. TypoScript: It is possible to define default TypoScript settings for all back end users here (see Chapter 5.8).
Example: [defaultPageTSconfig] = mod.web_layout.
         tt_content.colPos_list= 0,3

[enabledBeUserIPLock]
Boolean value (0,1). If 1 is set, the User/Group TSconfig option option.lockToIP is enabled. Further settings can be found in the TSConfig field of the user or group to be configured.
Example: [enabledBeUserIPLock] = 1

[fileDenyPattern]
String. In accordance with the eregi() function, components of names can be specified here. Files containing these components cannot be loaded to the server or renamed.
Example: [fileDenyPattern] = \.php\.|\.php3\n
[interfaces]
You can specify here which interfaces are presented to users for selection when logging in to the back end, and in which order. The choices are backend and frontend (comma-separated).
Example: [interfaces] = backend

[loginLabels]
The input options of the login screen can be overwritten here with other expressions, in German, for example.
Example: [loginLabels] = Username | Password | Interface | Log In | Log Out | ...
2.5 Configuration Options in TYPO3_CONF_VARS

[notificationPrefix]
This option allows a header to be set for messages from the system to the administrator.

Example: [notificationPrefix] = Miracles may happen...

2.5.5 [FE]: $TYPO3_CONF_VARS["FE"]
Configuration parameters in the following section referred to the front end, that is, the websites published by TYPO3.

[png_to_gif]
Boolean value (0,1). The value 1 enables the conversion of all PNG files generated in the front end to GIF files. This leaves behind a large number of temporary files in the typo3temp/directory.

Example: [png_to_gif] = 0

[tidy]
Boolean value (0,1). If 1 is set, the HTML code is cleaned and optimized with the tidy program. This option is recommended, especially during development periods, so that the HTML code generated can be more easily read. But remember that tidy, depending on the options, cleans or repairs defective HTML code. This option should be switched off on live systems, to prevent unnecessary load on the server. tidy can be obtained here: http://www.w3.org/People/Raggett/tidy/

Example: [tidy] = 0

[tidy_option]
Options: [all, cached, output]. all causes tidy to clean up all content before it is saved in the cache. cached causes content only to be cleaned before it is saved to the cache. output cleans HTML code only if it is requested from the cache.

Example: [tidy_option] = cached

[tidy_path]
Here the command tidy is specified, including the path and all necessary options, where appropriate. Apart from the default setting, other parameters can be defined in accordance with the documentation of tidy. To generate XHTML through tidy, for example, the expression --output-xhtml true should be added.
2 Installation

Example: [tidy_path] = tidy -i --quiet true --
          tidy-mark true -wrap 0

logfile_dir

Path. In the directory given here, TYPO3 writes log files in the notation of a web server for evaluation by statistical programs. The directory must be writable for the web server. The directory name must be concluded with a slash. More information can be found in Chapter 4.12.2.

Example: [logfile_dir] = /srv/www/logs/

logfile_write

The log files can be written using various methods. Without further settings, TYPO3 uses the UNIX command `echo`. Entering `fputs` causes TYPO3 to use the PHP function of the same name, which also works in `safe_mode` mode.

Example: [logfile_write] = fputs

publish_dir

Path to a directory in which TYPO3 should statically publish HTML pages. The directory must be writable for the web server. The pages can then be published from the Admin Panel in the publish area.

Example: [publish_dir] = /srv/www/htdocs/publish/

addAllowedPaths

Path, comma-separated list: other directories can be specified in which resources are stored for use in TypoScript. Paths must be specified relative to the web directory. The default is specified with a leading slash; without a slash, every directory is accepted that begins with the same expression.

Example: [addAllowedPaths] = b2b/, /b2c/

allowedTempPaths

Path, comma-separated list: additional paths where temporary images may be located for use by `imgResource` in TypoScript.

Example: [allowedTempPaths] = b2btemp/

debg

Boolean value (0,1). If enabled with 1, debug information is shown in the front end. This can also be set in TypoScript.

Example: [debug] = 1
2.5 Configuration Options in TYPO3_CONF_VARS

[simulateStaticDocuments]
Boolean value (0,1). Display of simulated static URL addresses is switched on by default with this entry, but needs to be enabled separately in TypoScript.
Example: [simulateStaticDocuments] = 1

[noPHPscriptInclude]
Boolean value (0,1). If enabled, PHP scripts are only called by TypoScript if they are located in the directory media/scripts/.
Example: [noPHPscriptInclude] = 0

[compressionLevel]
This value defines the compression of HTML pages in the front end through the zlib function in PHP. 1 is the lowest compression rate, and 9 the highest. Compression helps to spare bandwidth, but also puts more load on the server, the higher the compression rate. If you enter TRUE, the compression rate is automatically adjusted to the system load.
Example: [compressionLevel] = 0

[compressionDebugInfo]
Boolean value (0,1). If enabled, the size of the compressed and uncompressed versions of a page are shown at the bottom of the page. This should only be used for test purposes, as the pages for static evaluation are compressed twice.
Example: [compressionDebugInfo] = 0

[pageNotFound_handling]
String. With this option you can configure how TYPO3 should react to queries for unavailable pages. The default behavior is to display the next page. With TRUE or 1, an error message is displayed. Alternatively an HTML page can be specified, which will be displayed.

[pageNotFound_handling_statheader]
String. If the option [pageNotFound_handling] is enabled, the string set here is always sent as the header.
Example: [pageNotFound_handling_statheader] = Status: 404 Not Found
2 Installation

[userFuncClassPrefix]

This prefix must be the first part of each function or of the name of a class that is called from TypoScript, e.g. in the stdWrap function.

Example: [userFuncClassPrefix] = user_

[addRootLineFields]

Comma-separated list. A list of additional database fields in the pages table; should be used for Rootline queries.

Example: [addRootLineFields] =

[checkFeUserPid]

Boolean value (0,1). If enabled, login forms in the front end must specify the page ID (pid) of the front end user under which these are stored. If this is disabled, with 0, the eval configuration of uniqueInPid in the $TCA for the fe_users.username field should be changed to unique. The entry then looks like this:

```php
$TCA['fe_users']['columns']['username']['config']['eval'] = 'nospace,lower,required,unique';
```

The storage location no longer has to be specified in the TypoScript page on which the login form is located; all FE users are globally valid for the TYPO3 instance.

Example: [checkFeUserPid] = 1

[defaultUserTSconfig]

String. TSConfig entries can be predefined here for all front end users and groups.

Example: [defaultUserTSconfig] =

[defaultTypoScript_constants]

String. Option to redefine TypoScript for constants systemwide.

Example: [defaultTypoScript_constants] =

[defaultTypoScript_editorcfg]

String. Option, to define the TypoScript editorcfg configuration systemwide. Is used by the CSS Styler (extension key: tstem-plate_cssanalyzer).

Example: [defaultTypoScript_editorcfg] =
2.5 Configuration Options in TYPO3_CONF_VARS

**[dontSetCookie]**

Boolean value (0,1). If enabled, the system does not set any cookies in the front end, which also causes logins to be switched off.

Example: [dontSetCookie] = 0

**[get_url_id_token]**

String. In the front end, users can log in without a cookie if the TypoScript option `config.ftu` is enabled. In this case the user session is managed via a GET parameter, the name of which is specified here. In principle this type of session administration is not recommended, as it is more error-prone than the cookie variant.

Example: [get_url_id_token] = SESSID

**[content_doctypes]**

String. Page types (type number of the field `pages.doctype`) that are to be recognized by the system as pages or SysFolders are defined here as a comma-separated list.

Example: [content_doctypes] = 1,2,5,7

**[enable_mount_pids]**

Boolean value (0,1). With this option the function of Mountpages can be switched off globally (0).

Example: [enable_mount_pids] = 1

**[pageOverlayFields]**

String. The specified fields are used in database queries for multi-language websites. This option is relevant for extensions that add their own fields to the `pages` table.

Example: [pageOverlayFields] = title,subtitle, nav_title,media,keywords,description,abstr...
2.5.6 Other Options

[MODS: $TYPO3_CONF_VARS["MODS"]]

Used to contain options for configuring modules, but was replaced by the extension system.

[USER: $TYPO3_CONF_VARS["USER"]]

Used to contain options for configuring parameters for your own scripts, but was replaced by the extension system.

[SC_OPTIONS: $TYPO3_CONF_VARS["SC_OPTIONS"]]

This section is used to make available your own configuration options for any scripts at all (in general, BE modules) in TYPO3.

[EXTCONF: $TYPO3_CONF_VARS["EXTCONF"]]

Here you can add configuration options for your own extensions. During the installation in the EM, these should be displayed using the file ext_conf_template.txt.

Example: $TYPO3_CONF_VARS["EXTCONF"]

['my_extension_key']['my_option'] = 'my_value';

2.6 Separation of Production Server/Live Server

In cases where increased performance capacity is required and also for security reasons, it can be useful or even necessary to share the processing of pages and other contents, as well as their presentation across several different servers.

A simple possibility would be the publication of the website as static HTML pages, without a database or TYPO3 on the live system. But in the standard scenario, the website itself contains these dynamic elements generated from a database, which make TYPO3 necessary on the live system. The essential problem in such a scenario therefore lies in the synchronization of database content between different systems, where several different servers might be involved, perhaps to spread the load.

Depending on the database used, some very different synchronization mechanisms are available for this purpose. Please consult the documentation of your database vendor.

With direct synchronization, you cannot make any further decisions in the publication: what is live on the production system is automatically placed
2.6 Separation of Production Server/Live Server

online. Alternatively the synchronization can be performed manually, depending on the RDBMS used, in case a further release step is necessary.

For the TYPO3 installation delivering the site and offering no editing functions, it is advisable to disable the back end logins of regular users altogether by setting the following parameter in the All Configuration tab of the Install Tool: 

\$TYPO3_CONF_VARS['BE']['adminOnly'] = '1';

2.6.1 Static Pages

A second and simpler variation is to publish static pages, already mentioned briefly in section 2.4.4. The following entry in the Installation Tool and configuration in the Admin Panel of the front end causes all webpages to be saved in a predefined directory of the server.

Apart from the TYPO3-like systems for generating static pages, there are many other software solutions for all standard platforms. An example is HTTrack, available under both Linux and Windows, and which can be obtained without charge as an Open Source tool at http://www.httrack.com/.

2.7 Backup

You can use various methods to create backups. Most companies have proper backup strategies, and the safest way is to include the relevant directories of the web server in these backup routines, not forgetting the directory where the database files are stored.

If it is just a matter of creating backups from TYPO3 alone, there are various extensions available to perform this task. However, a relatively inexperienced administrator can easily set up an automatic backup system at the operating system level, which is normally a better solution, because the backups do not have to be manually started each time.
2 Installation

With an entry in the crontab system (UNIX), this script can be called up regularly: in the example below, this is done every 24 hours, at 1 o’clock in the morning.

After this, the backup files should be copied automatically onto a physically separate system. A script such as the following one could perform this conveniently and automatically via rsync. To do this, an SSH key is necessary to enable the data to be transmitted in encrypted form.

Here is an example script (with thanks to Harald Oest from http://www.ixsys.de/):

```
#!/bin/sh
# script rsync_backup.sh
# backup of web server document root via rsync to
# backup server
# additionally do a dump of typo3-db

# ip or fqhn of backup server
SERVER="my_backup_server"
# user account at backup server
USER="my_username"
# ssh-key (without passphrase!) used for login
SSHKEY="/root/.ssh/backup_server_key"
# destination dir at backup server
DSTDIR="/typo3_bkp"
# name of local typo3 database
DB="typo3_db"
# user account to access typo3 database
DB_USER="typo3_db_user"
# password to access typo3 database
DB_PASS="typo3_db_password"
# these directories will be rsynced with backup server
DIRS="/srv/www"

# do a mysql-dump and store result in source dir
/usr/bin/mysqldump --password=$DB_PASS -u $DB_USER $DB > \ /srv/www/typo3db_bkp.sql

# rsync all requested dirs
for DIR in $DIRS; do
    logger "rsync backup $DIR to $SERVER"
    rsync --rsh="ssh -i $SSHKEY" -a "$DIR $USER@$SERVER:$DSTDIR"
    done;

# get actual size of backup
ACT_SIZE=`ssh -i $SSHKEY $USER@$SERVER "du -sh $DSTDIR"

logger "total backup size: $ACT_SIZE"
```

22  IInnssttaallllaattiioonn

TYPO3, by Packt Publishing
You can generate an SSH key without a password query as follows:

```
linux:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): /root/.ssh/backup_server_key
Enter passphrase (empty for no passphrase): [Enter]
Enter same passphrase again: [Enter]
Your identification has been saved in /root/.ssh/backup_server_key.
Your public key has been saved in /root/.ssh/backup_server_key.pub.
The key fingerprint is:
croot@local_box
```

The key pair has now been created, and the private key is already in its correct location (/root/.ssh/backup_server_key). The public key now needs to be sent to the backup server:

```
linux:~# scp /root/.ssh/backup_server_key.pub user@my_backup_server:/home/.ssh
```

Now make a connection to my_backup_server and activate the SSH key:

```
root:my_backup_server:~# cat /home/.ssh/backup_server_key.pub >> /home/.ssh/authorized_keys
```

In order to clear up after this:

```
root:my_backup_server:~# rm /home/.ssh/backup_server_key.pub
```

And the same on the local computer:

```
linux:~# rm /root/.ssh/backup_server_key.pub
```

You can now make a connection to the backup server without a password:

```
linux:~# ssh -i /root/.ssh/backup_server_key user@my_backup_server
```

Here is an example of an appropriate crontab entry, which calls up the rsync backup at 1 a.m. in the morning:

```
# call backup script every night at 01.00
0 1 * * * root test -x /root/bin/rsync_backup.sh &
```

```
& /root/bin/rsync_backup.sh
```
2.8 Updates

TYPO3 updates are particularly enjoyable for the administrator, since they rarely last more than a few minutes. The reason for this lies in the often mentioned symbolic links, which summarize the actual TYPO3 version in a single directory; all the other files are either version-independent, and remain where they are, or are also symbolic links.

However the first step in every update is to make a backup of the entire installation, or at least of the database. With the command

```
linux:/srv/www># mysqldump -u user -p databasename
backup.tgz
```

you can create a backup file, after entering the password of the database user in MySQL. In emergencies, you can unpack this backup with the command

```
linux:/srv/www># tar xzf backup.tgz
```

and then, with the command

```
linux:/srv/www># mysql -u user -p databasename < backup.sql
```

write it back to the database.

Back to the update: to use the new version instead of the old one, you must delete the old symbolic link. For example, with

```
linux:/srv/www># {rm typo3_src}
```

your site is now offline!

With the command

```
linux:/srv/www># {ln -s ../typo3-src-3.8 typo3_src}
```

a new symbolic link is set to the new source that was previously placed on the server (see section 2.3.1).

Under Windows and generally with zip-distributions, if you are not using Junction, all the TYPO3 directories (typo3, t3lib, tslib) have to be replaced manually.

So if the symbolic link that points to the TYPO3 version is replaced, the update on the file level is already complete.

Please remember that after an update (or a downgrade) of TYPO3, you should first call up the Database Analyzer of the Installation Tool. With the COMPARE function, necessary changes to the database are identified and displayed. These are new database fields as well as changes in the
field definition of existing fields. Select **Implement All Changes** and carry out the update. The **COMPARE** function can also remove database tables that are no longer required from uninstalled extensions. It goes without saying that you should know what you are doing when carrying out this function, and of course: back up, back up, back up! The tables and fields to be removed are first renamed and then provided with the prefix `zzz_`. They can be reactivated by removing this prefix, for example with phpMyAdmin.

It is recommended, for obvious reasons, not to carry out the update on heavily visited sites. Instead you can duplicate the database and the file system, adjust the access data to the new database in the copy, and carry out emergency test scenarios. This helps you switch over in a few seconds, by changing the web server or swapping directories.

### 2.9 Help with Problems

The TYPO3 community maintains various contact points regarding installation problems. These can be wide-ranging due to the various possible combinations of web server, database, and PHP distributions used. Two mailing lists exist for installation questions, one each for Linux and Windows operating systems, which can also be used by nntp-client (newsgroups). In addition, there is an online archive you really should browse through to make sure that a question has not already been answered. Bear in mind that all those providing help are volunteers: as someone seeking help, you should try not to distract them with unnecessary or repeat questions. If you have discovered a bug not yet described anywhere, you can register it at [http://bugs.typo3.org/](http://bugs.typo3.org/).
TYPO3: Enterprise Content Management

TYPO3 is an extremely successful Open Source Content Management System, with a reputation for being very powerful but also complex. With this book we have tried to draw an overall picture of TYPO3, providing an insight into how it can be used by users, administrators, and developers.

This book is not a replacement for the references and tutorials on TYPO3.org; rather, it is a connecting link for developing a coherent picture of TYPO3 for beginners, users, and developers, allowing them to navigate on their own through the flood of information. The entire text is annotated with footnotes and so-called "softlinks", which take you to the relevant information on the TYPO3 online documentation.

What This Book Covers

Chapter 1 and Chapter 2 are concerned primarily with the theoretical basis of content management. With this background, the advantages of TYPO3 become clear, and from this we can form the basis for decisions on its strategic use. In addition, these introductory chapters take readers who have no previous knowledge on an excursion through the subject, presenting the most important terms and concepts. The installation and configuration of TYPO3 is discussed in detail.

In Chapter 3 we demonstrate the system using practical situations of content production. A complex tool must prove its value in a particular way, through the user-friendliness of its interface. After presenting the TYPO3 options and their functions, the section closes with a practical example on working effectively with TYPO3.

Chapter 4 part deals with the administration of the system, and consequently with adjustments to conditions and processes that are defined by producers in their work with the system. In doing so, it is shown, using examples, how the means available intertwine, and how they are used in practice.
Chapter 5 describes the production of websites using TYPO3. Starting with the installation, this section discusses the programming of templates with TypoScript and shows the different methods available.

In Chapter 6 and 7 we introduce the extension interface of TYPO3, the Extension System, describing the basics and ways to develop your own functional extensions in the TYPO3 framework. Here we can observe, from a developer perspective, the integration with core functions and the different parts of the TYPO3 architecture which can be extended.

The entire text is annotated with footnotes and so-called "softlinks". The footnotes are aimed at encouraging further reading in areas not directly involved with technical aspects of TYPO3. The softlinks connect the book to TYPO3.org and other resources.

By entering the number code on the TYPO3.org website (http://www.typo3.org/book/) you will be taken to the corresponding topic in the online documentation, or to sources going into more detail. In this way, technical references and documentation are included, which are as up-to-date as possible, but the reader is also introduced to the thematic structural online resources, providing him with a sense of orientation in an ever-growing profusion of information.

Where to buy this book


Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.