La traduction n'est pas terminée

Install Tiki directly from the SVN repository

This method requires **a**. a server with shell (command line) access and **b.** a running SVN instance on this server.

For the SVN part please review the page Get Code on dev.tiki.org. There and if necessary on the related pages you find notes about the SVN commands and the particular paths you need to use in the command line to automatically checkout or upgrade the Tiki into the intended installation directory (= tikiroot folder).

Example for a fresh checkout (recommended also for major upgrades)

Method 1
Checking out to a new folder (that you will specify)

```
$: svn checkout https://svn.code.sf.net/p/tikiwiki/code/trunk mynewfolder
```

Method 2
Checking out to the current folder (when you created the intended tikiroot already and changed into this directory prior to the checkout):

```
$: svn checkout https://svn.code.sf.net/p/tikiwiki/code/trunk .
```

Example for a minor upgdate:

Do not forget to switch into the tikiroot of the Tiki you want to update

```
$: svn up
```

After SVN has downloaded the Tiki files, the Tiki is not yet complete. You have to run the composer
The composer automatically downloads and installs the so called 'vendor files' from ecternal sources.
Vendor files are external scripts and libraries, which we pack to the Tiki code on release time but we do not manage in our own repository.
Examples of those vendor files are the Bootstrap code, the jQuery library, Tablesorter, Elfinder etc..
Usually (= on most servers) you additionally have to fix certain file permissions.
Both can be accomplished by simply answering a number of questions after starting the setup.sh script in the shell (= command line).
This procedure is quite self-explaining. Just follow the instructions of the script.

Start setup.sh, after having switched into the tikiroot (in case you are not there anyway):

```
$: sh setup.sh
```

For more information on using setup.sh see also the section 4.1 Troubleshooting below. In this section we treat only the composer part, which is only applicable when you use an installation from the repository.

To start the composer you have to confirm the option "c" after starting the setup.sh script:

```
$: sh setup.sh Your choice [c]? c
```

or

```
$: sh setup.sh Your choice [f]? c
```

FAQ: On my server the composer script is not starting. What can I do, when I get the following error message?

```
Wrong PHP version: phpABC < required PHP version. A version >= phpXYZ is necessary.
```

The problem is, that many shared hosting providers keep the setting of the default local PHP version of the shell (command-line) to an outdated PHP version which does not match the PHP requirements of Tiki or of the composer. Alternativ PHP settings in the .htaccess or in the php.ini file or in the user.ini file have no effect to the PHP version in the shell (command line).

However most of these servers have newer PHP versions available in the shell (command line), which can be used alternatively.
For Tiki 16, likely soon backported to Tiki 15, we provide an option, which allows to run the composer with an alternative up-to-date PHP version, given any one is available on the particular server.

Another issue to be addressed is, that the providers have different naming conventions how to distinguish their available PHP versions. To address this issue we implemented the option "-p" for our setup.sh script to provide the option to manually change the PHP version used by the script:

```
$: sh setup.sh -p PHPVERSION
```

In a first step, the setup.sh script automatically guesses three typically used namings of up-to-date versions, commonly used on many servers and uses the first one that matches:

```
php55 php5.5 php5.5-cli
```

If the script finds one of these, you should get the following message prior to an autostarting composer:

```
Wrong PHP version: phpABC < required PHP version. A version >= phpXYZ is necessary. Searching for typically named alternative PHP version ... ... correct PHP version phpDEF detected and used Local PHP version >= required PHP version XYZ - good Loading composer repositories with package information Installing dependencies from lock file (...)
```

Now you are done. Problem solved.

But if none of these options work out, you will get the following error message:

> Wrong PHP version: phpABC < required PHP version. A version >= phpXYZ is necessary. Searching for typically named alternative PHP version ... ... no alternative php version found. Please provide an alternative PHP version with the -p option. Example: sh setup.sh -p phpXYZ. You can use the command-line command 'php[TAB][TAB]' to find out available versions.

This should be quite self explaining.
You simply have to use the php command in the shell, followed by twice typing the TAB key without any space or other key and enter. then the shell (command line) will provide you a list of the available PHP versions which you can use (obviously using the naming convention used on this particular server):

Example:

> $: php[TAB][TAB] php php-53 php-54 php-55 php-56 php-70

How to use this information:
Given the required PHP version would be php 5.5, but your provider uses php 5.3 by default, you simply need to type the following command to start setup.sh with the right PHP version for the composer (in this particular case php-55) ...

> $: sh setup.sh -p php-55 Your choice [c]? c

And swoooosh the setup.sh will start the composer with php 5.5 after the following message:

> Wrong PHP version: phpABC < required PHP version. A version >= php55 is necessary. Searching for typically named alternative PHP version ... ... correct PHP version php55 detected and used Local PHP version >= required PHP version 55 - good

Now you need only to follow the setup.sh dialog to fix directory permissions (option f is reasonably secure works on most average servers. If you need a more secure setting or if your server needed a less restrictive setting (better change the server then), you can use another option. To find out which permission option is best n your server, you can use Permission Check.

You are done now and can go to your website to run the installer.