

# Saurus CMS CE Installation Guide

Document version: English, 4.7.0  
© Saurused Ltd 2000-2010

# Contents

<b>CONTENTS.....</b>	<b>2</b>
<b>SYSTEM REQUIREMENTS.....</b>	<b>3</b>
SERVER PLATFORMS.....	3
OTHER REQUIREMENTS.....	3
USED LGPL AND OTHER COMPONENTS.....	3
SUPPORTED WEB BROWSERS.....	4
DISK SPACE REQUIREMENTS.....	4
<b>LICENSING.....</b>	<b>4</b>
<b>INSTALLATION.....</b>	<b>5</b>
LINUX INSTALLATION (USING SHELL).....	5
INSTALLATION USING FTP.....	6
REQUIRED PHP SETTINGS.....	7
REQUIRED .HTACCESS DIRECTIVES.....	7
INSTALLATION TROUBLESHOOTING.....	8
<b>VERSION UPGRADE.....</b>	<b>8</b>
VERSION UPGRADE IN LINUX (USING SHELL).....	8
VERSION UPGRADE USING FTP.....	8
VERSION UPGRADE IN WINDOWS.....	9
<b>CONFIGURING.....</b>	<b>10</b>
CONFIGURING ALIAS MODULE.....	10
<b>MORE INFORMATION.....</b>	<b>11</b>

# System requirements

The easiest way to check all criteria at once is to [download Check Requirements script](#) and place it to your website.

## Server Platforms

Saurus CMS is currently available for following server platforms:

- Operating Systems: Linux, Microsoft Windows NT/2000/XP.
- Web Server Software: Apache.
- Program language: PHP 5.0.0 until 5.2.11.
- Database: MySQL 4.x until 5.1.35.

Saurus CMS should be able to run on any web server which is supported by PHP. Saurus CMS is officially tested and supported with Apache running on Linux and Windows. Other platforms might also work but they are currently untested and unsupported.

**Note:** When using Apache it is not recommended to run PHP as a CGI binary, use the Apache module instead. The Apache module is not only faster, but it also causes less load on the server.

**Note:** PHP Safe mode is not officially supported.

## Other requirements

- GD library (<http://www.php.net/gd>) must be installed to use Gallery module.
- Apache URL Rewrite Engine module (mod\_rewrite) must be installed to use Saurus CMS Alias module. See more in section "Configuring the Alias module".
- PHP PCRE library (configure option `--with-pcre-regex`), <http://www.php.net/pcre>
- PHP mbstring library, Multibyte String Functions <http://www.php.net/mbstring>

## Used LGPL and other components

- Smarty template engine is used for creating Saurus API templates. Smarty is licensed under the LGPL (Lesser GPL, <http://www.gnu.org/copyleft/lesser.html>), The unmodified source is included in the Saurus CMS package. See: <http://smarty.php.net/>
- Text editor uses FCKeditor that is licensed under the LGPL. The unmodified source is included in the Saurus CMS package. See: <http://www.fckeditor.net/>
- jQuery JavaScript framework is used under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). See: <http://jquery.com/>.

- Prototype JavaScript framework is used under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). See: <http://www.prototypejs.org/>.
- Thickbox JavaScript component is used for admin pages under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). See: <http://jquery.com/demo/thickbox/>.
- jQuery UI Date Picker component is used for Calendar displaying under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). See: <http://kelvinluck.com/assets/jquery/datePicker/v2/demo/>.
- PclZip library is used for handling ZIP archives licensed under the LGPL (Lesser GPL, <http://www.gnu.org/copyleft/lesser.html>).
- jQuery lightBox plugin is used for Saurus 4 default gallery templates, licensed under CC Attribution-No Derivative Works 2.5 ([http://creativecommons.org/licenses/by-nd/2.5/br/deed.en\\_US](http://creativecommons.org/licenses/by-nd/2.5/br/deed.en_US)). See: <http://leandrovieira.com/projects/jquery/lightbox/>.
- jQuery ScrollTo plugin is used by the File Manager under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). See: <http://flesler.blogspot.com/2007/10/jqueryscrollto.html>
- SWFUpload is used for multiple file uploads under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). See: <http://swfupload.org/> .
- Services\_JSON library is used for encoding JSON strings, licensed under BSD License (<http://www.opensource.org/licenses/bsd-license.php>). See: <http://pear.php.net/pepr/pepr-proposal-show.php?id=198> .

## Supported Web Browsers

- For website visitors: any HTML 4.0-compliant web browser with CSS 2 support (including Macintosh, Unix, and Windows environments).
- For editors and administrators: Internet Explorer 6.0+ and Mozilla Firefox 1.0+ web browsers with JavaScript support on PC, Linux and Mac OS workstations.

## Disk space requirements

- Saurus CMS itself takes up approximately 15 MB of disk space.

# Licensing

Saurus CMS SE is open source software and is licensed under Mozilla Public License version 1.1. The detailed license info can be found in the "install/LICENSE" file of the installation package.

# Installation

Installation of Saurus CMS is simple: using the web-based installer you have to answer few questions and the rest will be handled automatically. No need to edit configuration files or manually set up the database. Follow the instructions below and you will have a working website in minutes.

Make sure you have PHP, MySQL installed before you continue.

## Linux installation (using shell)

1. Unpack the downloaded Full Installation package to the folder where web server can access them. Eg:

```
mkdir saurus  
tar -xvpf SaurusCMS-4.7.0.tgz -C saurus/
```

2. Execute the following shell commands within unpacked folder:

```
cd install/  
bash ./pre_install.sh
```

Script `pre_install.sh` will temporarily change file permissions for `config.php`.

3. Open the web browser and go to the URL of the website, e.g.:

<http://www.yoursite.com/install.php>

Installation script will install the database and the `config.php` file will be adjusted with the required database access variables.

4. After successfully finishing `install.php` steps, execute the following shell command within `install/` folder:

```
bash ./post_install.sh
```

Script `post_install.sh` will remove temporary writing permissions for the `config.php` files and delete installation file `install.php` and `install/` folder.

5. Be sure that the following upload directories have write permissions for the web server:

**public/** - files used for content management  
**shared/** - files used for content management  
**extensions/** - Saurus extensions  
**classes/smarty/cache/** - Saurus API templates cache  
**classes/smarty/templates/** - Saurus API templates  
**classes/smarty/templates\_c/** - Saurus API compiled templates code

## Installation using FTP

Having shell access on the server is the easiest way to install Saurus CMS, but if you have only FTP access, follow the instructions below:

1. On your desktop computer: unzip the downloaded Full Installation package file. The website folder will be created.
2. Use binary mode (not ascii mode!) for file transfer and upload created folder to your website server using FTP.
3. Using FTP program set temporary read & write permission for everybody (666) for following files:
  - config.php
  - config-old.php
4. Run the web-based installation script <http://www.yoursite.com/install.php>.

In case of web hosting you probably have received your MySQL account information from your hosting company. During the installation you will be asked for database's name, username and password.

5. Delete install.php file and install/ folder from your website folder.
6. Using FTP program change file permissions back to normal: read & write to owner, read to everybody (644) for files:
  - config.php
  - config-old.php
7. Your website is now installed and accessible on URL <http://www.yoursite.com>
8. Be sure that the following upload directories have write permissions for the web server:
  - public/** - files used for content management
  - shared/** - files used for content management
  - extensions/** - Saurus extensions
  - classes/smarty/cache/** - Saurus API templates cache
  - classes/smarty/templates/** - Saurus API templates
  - classes/smarty/templates\_c/** - Saurus API compiled templates code

## Required PHP settings

Saurus CMS uses default PHP settings.

Required values for PHP settings are:

```
file_uploads – On
register_globals – Off
safe_mode – Off
short_open_tag - On
```

Recommended values for PHP settings are:

```
post_max_size – 16M
upload_max_filesize – 16M
memory_limit – 16M (if memory limit is enabled in PHP)
```

## Required .htaccess directives

Usage of .htaccess files may be prohibited by webserver configuration or by service provider's security policy. Depending on server environment rules you may have limited access to use some commands in the .htaccess files.

Saurus CMS uses .htaccess directives for following purposes:

- securing upload directories, code libraries and templates

In case you are not allowed to use these directives byself, you should ask your system administrator or service provider to add them to your virtual host configuration.

Default .htaccess files in Full Installation package:

### **/ (website root)**

```
DirectoryIndex index.php index.html
```

### **public/**

```
<Files *>
SetHandler default-handler
</Files>
```

### **shared/ classes/**

```
deny from all
```

## Installation troubleshooting

- If you have trouble with the installation then run script **check\_requirements.php** in the install/folder. This script will check if your server meets the minimum requirements.
- **Technical FAQ** can be found on page <http://www.saurus.info/technical-faq>

# Version upgrade

## Version upgrade in Linux (using shell)

1. Download version upgrade package **SaurusCMSupdateX.X.X.tgz** and unpack it to the temporary folder (outside of the website folder):  
  

```
tar -xvzf SaurusCMSupdateX.X.X.tgz
```

  
Following files will be unpacked:  
saurus\_cms\_updX.X.X.tgz  
README  
CHANGELOG
2. **Unpack** file saurus\_cms\_updX.X.X.tgz to your website folder:  
  

```
tar -xpvzf saurus_cms_updX.X.X.tgz -C /path/to/your/website/root/
```
3. Open browser and run web-based upgrade script **<http://www.yoursite.com/install.php>**. Follow instructions.
4. **Delete** file **install.php** from your website folder after successful upgrade.
5. **Delete** version upgrade **package** from the temporary folder.

## Version upgrade using FTP

1. On your desktop computer unzip the downloaded **SaurusCMSupdateX.X.X.zip** package.  
  
Following files will be unpacked:  
saurus\_cms\_updX.X.X.zip  
README  
CHANGELOG
2. **Unpack** file saurus\_cms\_updX.X.X.zip



3. Use binary mode (not ascii mode!) for file transfer and upload created files (from saurus\_cms\_updX.X.X.zip) to your website server using FTP.
4. Open browser and run web-based upgrade script **<http://www.yoursite.com/install.php>**. Follow instructions.
5. **Delete** file **install.php** from your website folder after successful upgrade.
6. **Delete** version upgrade **package** from your desktop computer.

## Version upgrade in Windows

1. Download update package **SaurusCMSupdateX.X.X.zip** and unzip it to the temporary folder (outside of the website folder).

Following files will be unpacked:

saurus\_cms\_updX.X.X.zip

README

CHANGELOG

2. Unpack file saurus\_cms\_updX.X.X.zip to your website folder.
3. Open browser and run web-based upgrade script **<http://www.yoursite.com/install.php>**. Follow instructions.
4. **Delete** file **install.php** from your website folder after successful upgrade.
5. **Delete** version upgrade **package** from the temporary folder.

# Configuring

## Configuring Alias module

1. Install Apache URL Rewrite Engine module (mod\_rewrite), see: [http://httpd.apache.org/docs/mod/mod\\_rewrite.html](http://httpd.apache.org/docs/mod/mod_rewrite.html).
2. Open the Apache configuration file and add following lines to your virtual host configuration:

```
DirectoryIndex index.php index.html

Options +SymLinksIfOwnermatch

# rewrite rules for CMS alias's
Options +SymLinksIfOwnerMatch
RewriteEngine On

RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_URI} !^/admin/(.*)
RewriteCond %{REQUEST_URI} !^/classes/(.*)
RewriteCond %{REQUEST_URI} !^/editor/(.*)
RewriteCond %{REQUEST_URI} !^/extensions/(.*)
RewriteCond %{REQUEST_URI} !^/idcard/(.*)
RewriteCond %{REQUEST_URI} !^/js/(.*)
RewriteCond %{REQUEST_URI} !^/public/(.*)
RewriteCond %{REQUEST_URI} !^/px/(.*)
RewriteCond %{REQUEST_URI} !^/px_custom/(.*)
RewriteCond %{REQUEST_URI} !^/shared/(.*)
RewriteCond %{REQUEST_URI} !^/styles/(.*)

RewriteRule ^(.*)$ /map.php?mod_rewrite=/&cmd=$1 [L,PT,QSA]

# RewriteLogLevel 0 will disable logging, we suggest set it to 2
# for debugging problems
# RewriteLogLevel 0

# set correct log file path here, we suggest set it only when debugging problems:
# RewriteLog /var/log/httpd/rewrite.log
```

**NB!** This block must be placed outside of <Directory> directives block.

**NB!** In case you have any additional directories or files under your website root directory, which are not part of the CMS installation, you should exclude them in the alias configuration. Add following lines for each directory or file:

```
RewriteCond %{REQUEST_FILENAME} !^/notcmsdir/(.*)
```

We suggest to keep the logging level at "0" on the production/live sites, because of remarkable space usage. Set RewriteLogLevel to "2" when developing the new site and debugging aliases.

### Using subfolder

If your website is located in a subfolder, eg <http://www.yoursite.com/web/> then the subfolder name must be added to rewrite conditions:

```
DirectoryIndex index.php index.html

Options +SymLinksIfOwnermatch

RewriteEngine On

RewriteCond %{REQUEST_FILENAME} ^/web
RewriteCond %{DOCUMENT_ROOT}%{REQUEST_FILENAME} !-d
RewriteCond %{DOCUMENT_ROOT}%{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !^/web/admin(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/classes(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/editor(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/extensions(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/js(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/public(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/px(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/shared(/.*)
RewriteCond %{REQUEST_FILENAME} !^/web/styles(/.*)
RewriteRule ^/(.*)$ /web/map.php?mod_rewrite=/web&cmd=$1 [L,PT,QSA]

# RewriteLogLevel 0 will disable logging, we suggest set it to 2
# for debugging problems
# RewriteLogLevel 0

# set correct log file path here, we suggest set it only when debugging problems:
# RewriteLog /var/log/httpd/rewrite.log
```

## More information

### Saurused OÜ

Official website

<http://www.saurus.info>

Product support pages

<http://www.saurus.info/support>

User forum

<http://forum.saurus.info>