ADMINISTRATOR MANUAL

DataWiki
The smart Wiki

Version 1.7.1

DIQA Projektmanagement GmbH
Pfinztalstraße 90
76227 Karlsruhe
http://www.diqa-pm.com

Document Information

Document	DataWiki 171 Administration manual v11.docx		
Type	User Guide		
created	2012-12-01	0.1	DH
released	2012-12-13	1.0	**Hansch**
Version	1.1 for DataWiki 1.7.1		
Status	Final		

History:

Version	Date	modified by	Description
1.0	2012-12-10	DH	• Final version
1.1	2012-12-13	DH	• Included revisions from MER in chapter 1.3

Approved	Place, Date	Name
	Karlsruhe, December 13th, 2012	DH

Pfinztalstrasse 90
76227 Karlsruhe, Germany
Fone: +49 721 609 517 26
Email: info@diqa-pm.de

This document is a revision of the “Administrator Guide for SMW+ 1.7.0” that was retrieved in July 2012 from:

http://www.smwplus.com/index.php/Help:Administrator_Guide_for_SMW%2B_1.7.0

This document makes use of the user manuals of the following software products:

- Deployment framework (Wiki Administration Tool)
- Enhanced retrieval extension
- SMW+ bundle
- SMW+ Community Edition Windows Installer, Amazon AMI, Vmware
- SMW+ sandbox
- SMWHalo

Contents

1 Installation

1.1 Windows Installer

1.1.1 Requirements

DataWiki 1.7.1 Windows installer is pre-configured to be installed in a MS Windows Environment. DataWiki is approved to run on the following Windows versions:

- Windows 7 (Professional, Ultimate or Enterprise)
- Microsoft Windows Server 2008 R2 (R2 is important!)

Supported Browsers:

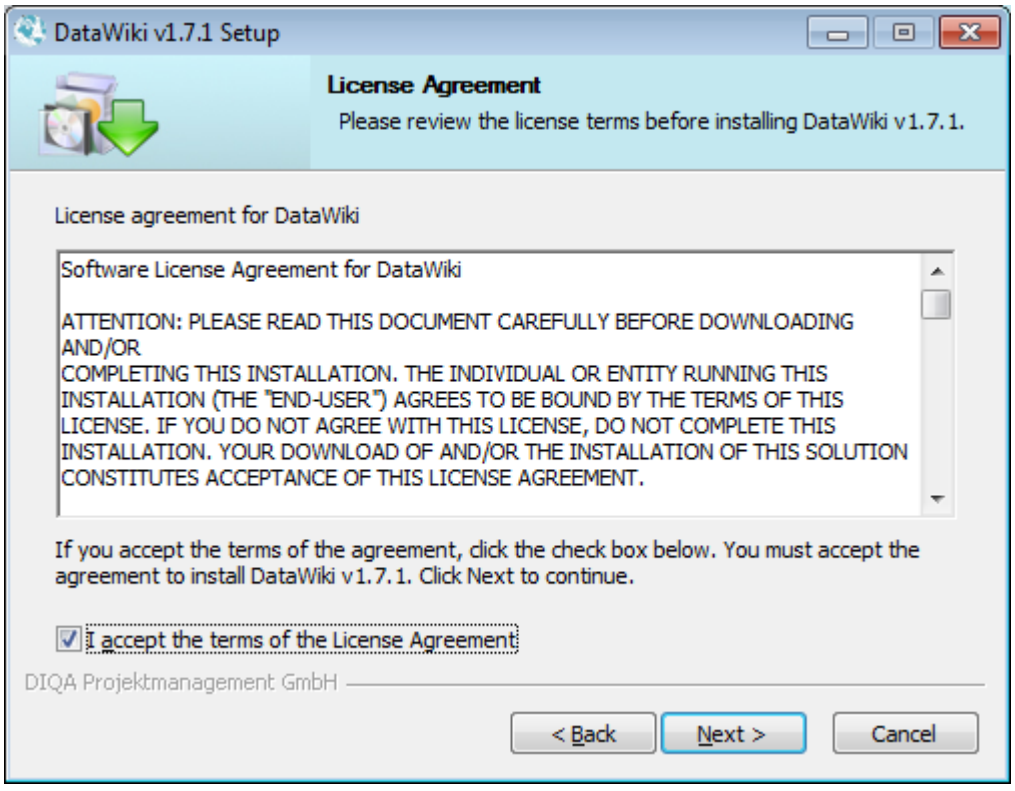
The DataWiki front-end runs in the web browser. DataWiki 1.7.1 has been approved for the following browsers:

- IE 8.x (which MUST NOT run in compatibility mode)
- Mozilla Firefox (later than 7)
- Google Chrome (later than 15)

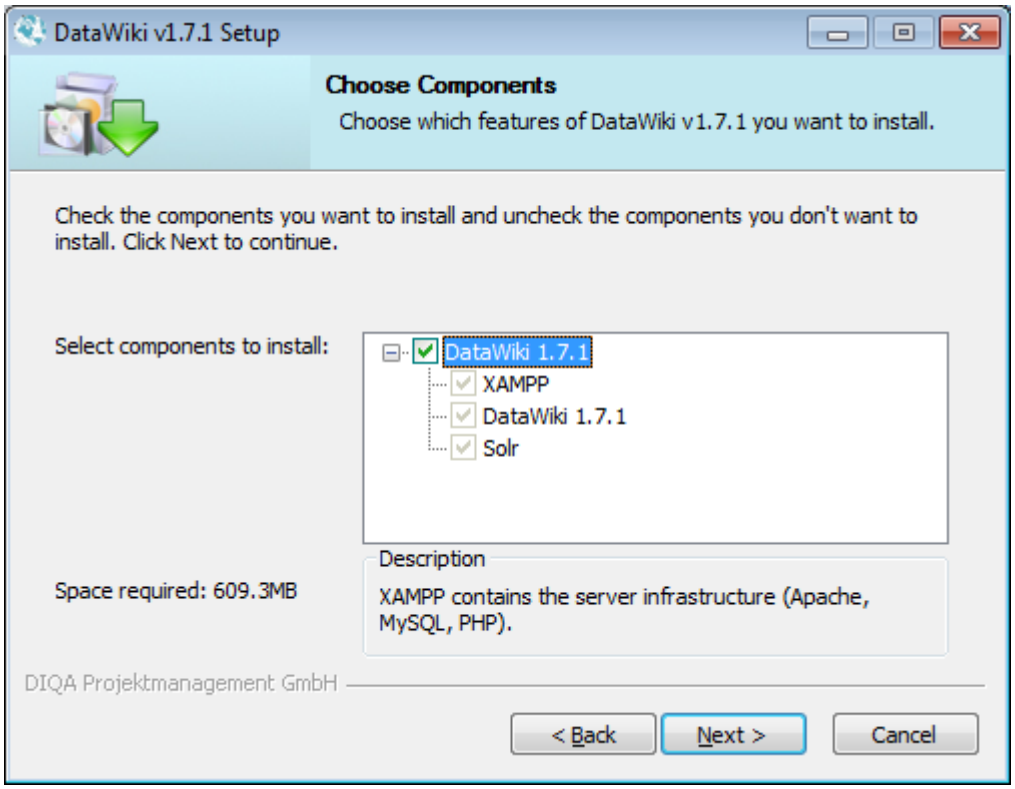
1.1.2 The Installation Wizard

1. Stop Skype or other processes on your server that block the TCP/IP port 80 before you start the installation.
2. Start the installation process by copying the DataWiki distribution into the file system of the server where DataWiki is to be operated on. **Double click** the file named: "DataWiki_1.7.1_B17.exe" to launch the installation tool.
3. A successful installation start shall be accompanied by a welcome screen similar to the one illustrated below:

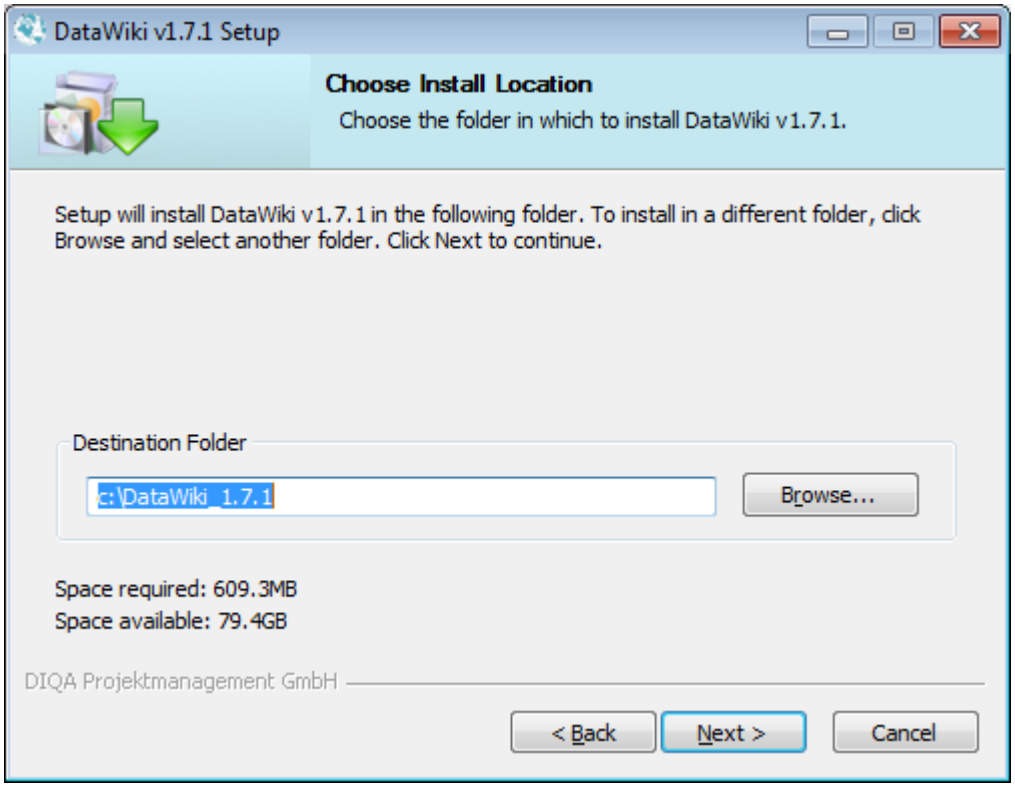
4. Click **Next** to proceed with the installation process. This displays the **License agreement**.
5. Read the License agreement and check the **I accept the terms in the License Agreement** if you do and click **Next.** If you do not accept the terms, you may click **Cancel** – doing this will abort the installation process.



6. In the next screen, you are asked to select the components you want to install. The default setting must be applied to install DataWiki completely:



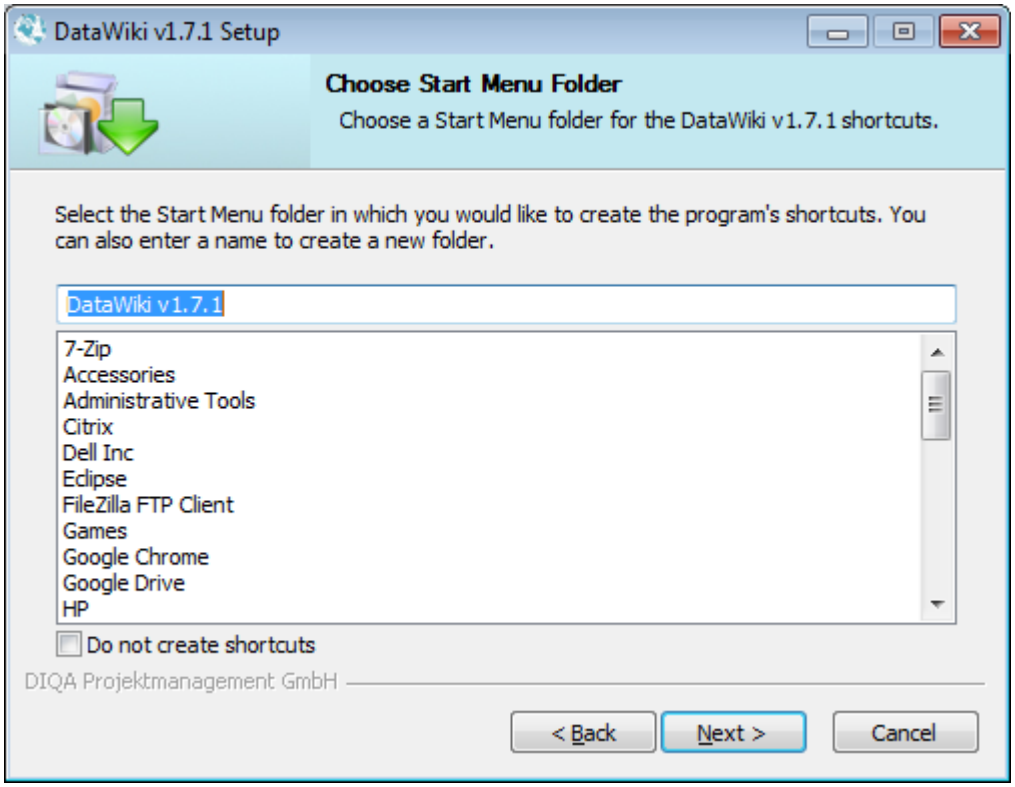
7. The installation proposes the destination folder of your DataWiki installation. This is the folder where all files and third party components needed to operate DataWiki will be installed in. Browse to a different destination folder on your server by clicking on **Browse** which opens a folder selection dialog. Once you verify your desired destination folder in your server's file system, proceed with the installation process by clicking **Next**:



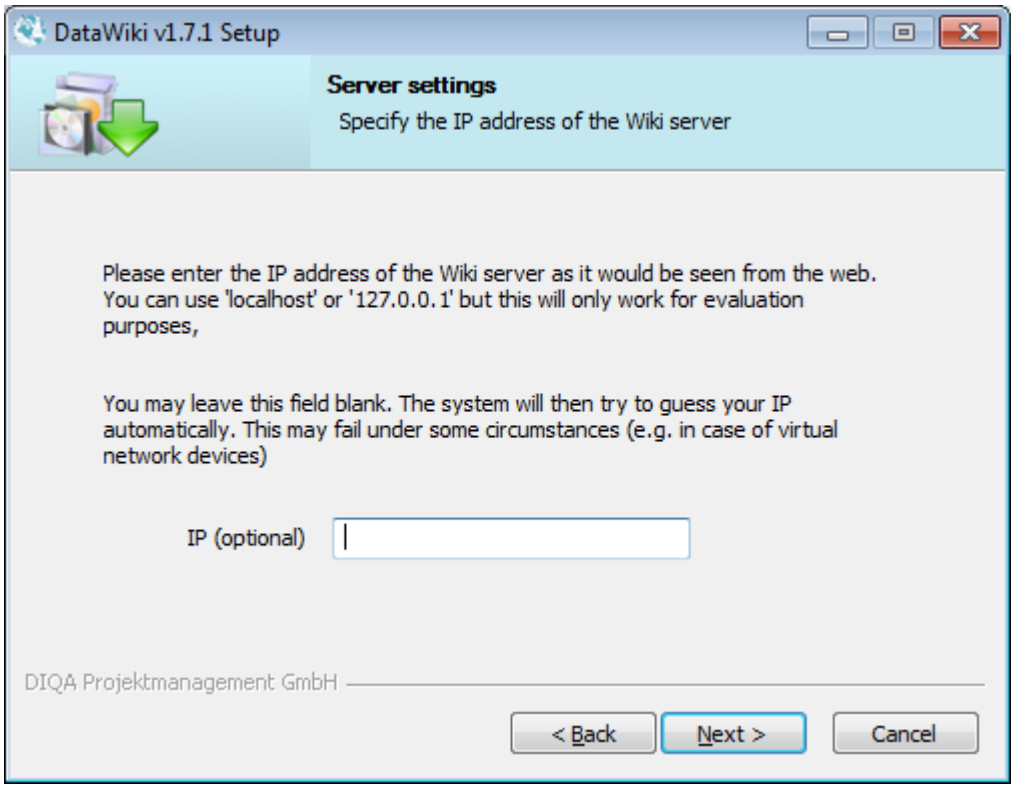
Important Note:

Note that DataWiki must not be installed in directories containing parentheses like "Program ". Note that you don't have full write privileges for "C:\Program files", even as an Admin.

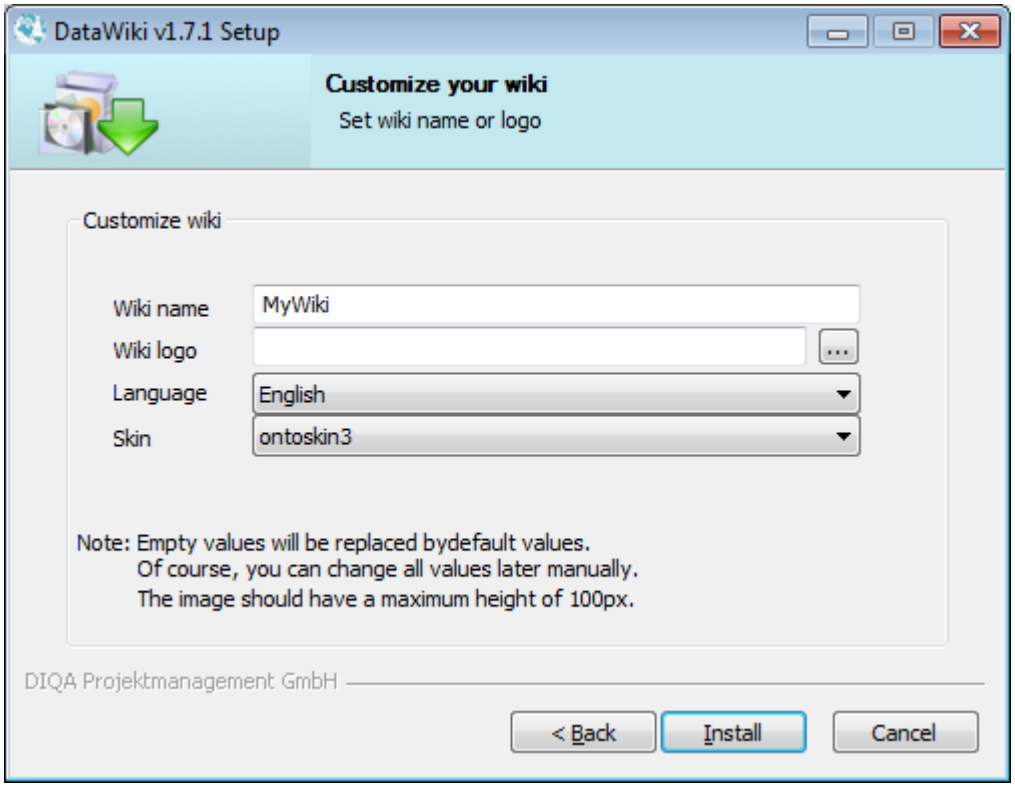
8. At this level, you can decide whether shortcuts should be created on the start menu folder or not. If you do not want the shortcuts, just activate the 'Do not create shortcuts' box.



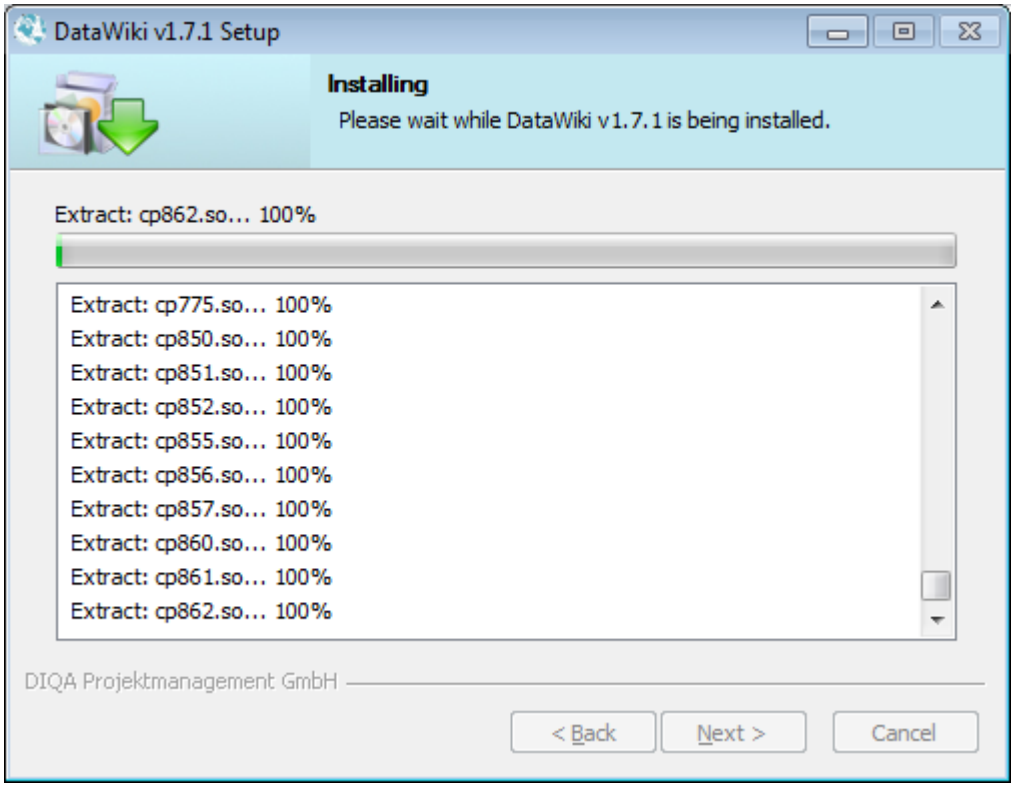
9. Now enter the IP address of your server. You may leave the field blank and the system will in this case automatically obtain your IP address. In some cases, such as virtual network devices, it is recommended that one enters the IP manually:



10. The installation tool will at this point offer options which may be used to customize your DataWiki installation. The options available are:
    a. **Wiki name**: This should be the name that will be displayed in the browser's title bar when using DataWiki.
    b. **Wiki logo**: This logo will be displayed on the upper left corner of every wiki page.
    c. **Language**: One may currently use English or German as the wiki content language.
    d. **Skin**: Choose ontoskin3.



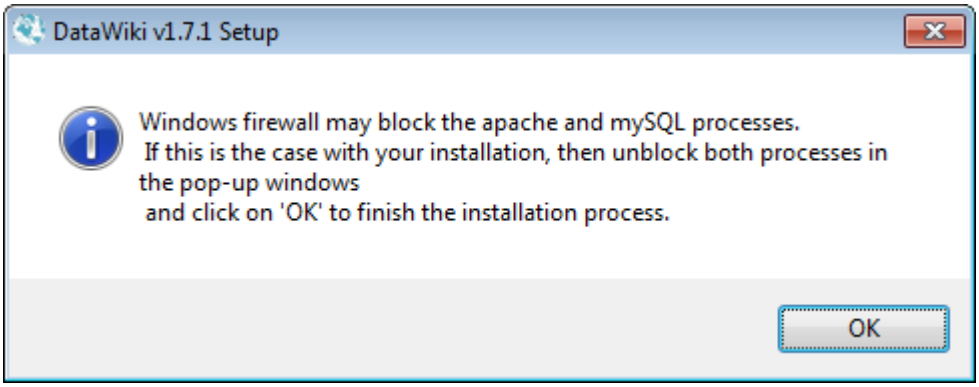
11. Select **install** to run the installation process.



Note:

DataWiki needs TCP port 80. If any application (like Skype) or a service blocks this port, you will be prompted to close this application

12. DataWiki will install three processes that need to freely access different ports. Windows may ask you if it should block these ports. In order to run the DataWiki, it is necessary that you unblock these processes.
    One process is the **Apache WebServer** which the wiki uses, the second one is the **MySQL database** which is used for storing the content. The last one is the **solr** service which is required for the wiki search.

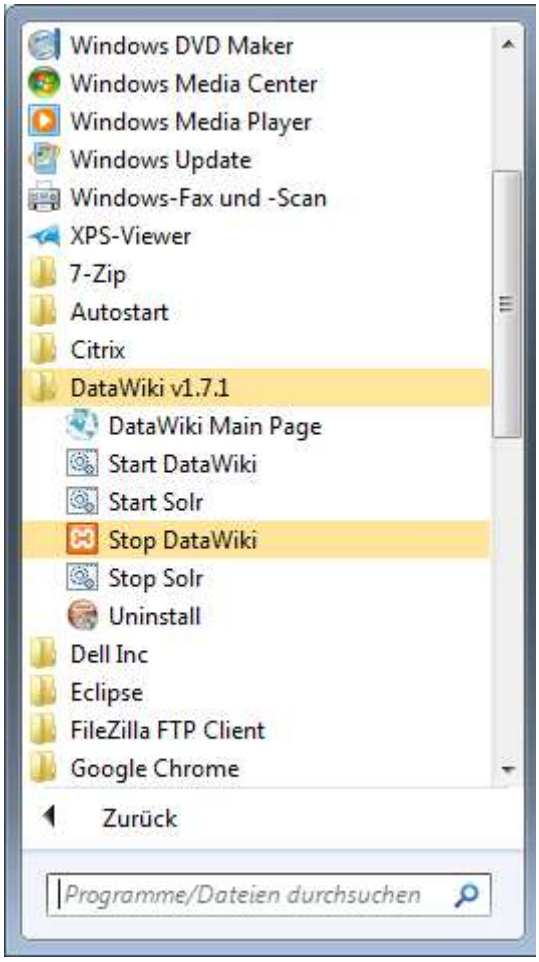


13. The installation process will have completed successfully if the installation tool displays the "Completing the DataWiki v.1.7.1 Setup Wizard" window.

Note:

We recommended that you activate the "Run wiki on startup (installs Windows services)" tick box – Activating this option adds the processes of Apache, MySQL, and Solr server to the Autostart of Windows. Additionally the processes Apache and MySQL are registered as Windows services per default.

14. Click on **Finish** to close the installation tool.
15. Six different DataWiki icons should now shop up in the start menu under DataWiki program path:
    a. **DataWiki Main Page** – Opens the Main Page
    b. **Start DataWiki** – Starts the Apache webserver and the database
    c. **Start Solr** – Solr is the popular, blazing fast open source enterprise search platform from the Apache Lucene project. Semantic MediaWiki plus uses Solr for the powerful faceted browsing feature
    d. **Stop DataWiki** – Stops the Apache webserver and the database
    e. **Stop Solr** – Stops Solr
    f. **Uninstall** – Uninstall DataWiki



1.1.3 Starting DataWiki

1. Open the start menu and select **All programs > DataWiki v1.7.1 > Start DataWiki** to start the Apache web server and MySQL database server.
2. Open the start menu and select **All programs > DataWiki v1.7.1 > Start Solr** to start the Solr server.
3. Open the start menu and select **All programs > DataWiki v1.7.1 > DataWiki Main Page** to open the Main Page of the wiki in your default web browser

Note:

You can alternatively enter the following URL into the URL-field of your web browser to open the Main Page of the wiki directly: 'http://localhost/mediawiki/index.php' (You can use "127.0.0.1" instead of "localhost")

1.1.4 Database Access

If you need direct database access for any reason, the default credentials are:

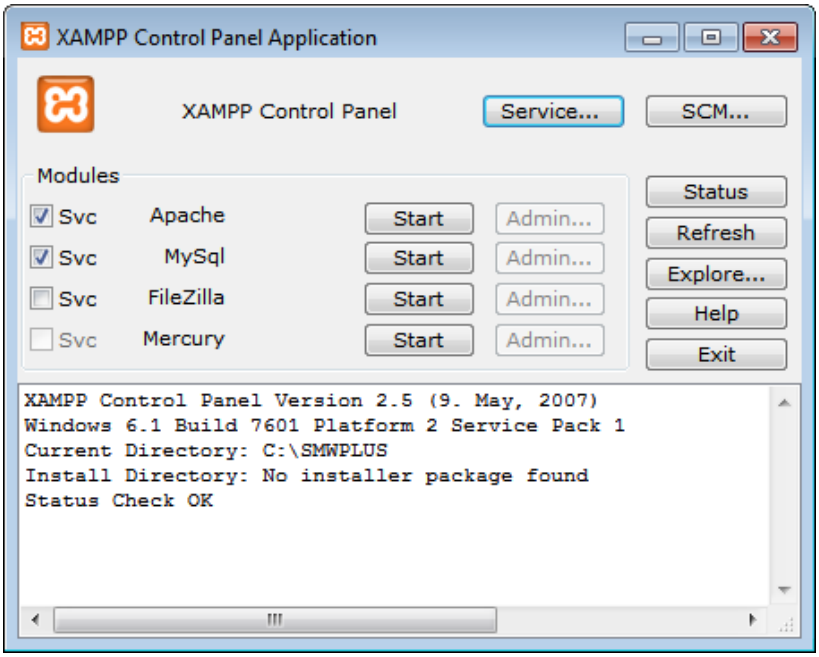
user: root
password: m8nix

We strongly recommend that you change these credentials for security reasons.

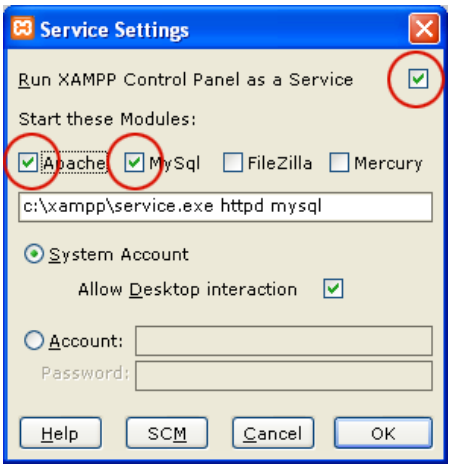
1.1.5 Configuring DataWiki as a Service

You can register DataWiki so that it automatically restarts after a shutdown. To do this:

1. Go to the **DataWiki installation directory**
2. Open **xampp-control.exe**
3. Click **Service**...



4. Register **Apache** and **MySQL** as Windows services:



5. Click **OK**

1.1.6 Logging into DataWiki

1. Log into the Wiki as administrator (a.k.a. "WikiSysop") by clicking on 'Log in' and entering the credentials for the administrator account.
   The default credentials for the administrator account are:

```
username: WikiSysop
password: root
```

Note:

We strongly recommend that you change the WikiSysop account password on the preferences page once you log into the wiki successfully

2. Open the special page **Special:Preferences** and select to the **Change Password** section to change the administrator password.

1.2 VirtualBox Image

The DataWiki VirtualBox image contains Ubuntu v12.04 (64 bit) and DataWiki 1.7.1.

1.2.1 Requirements

In order to use this VirtualBox image, you need to download and install:

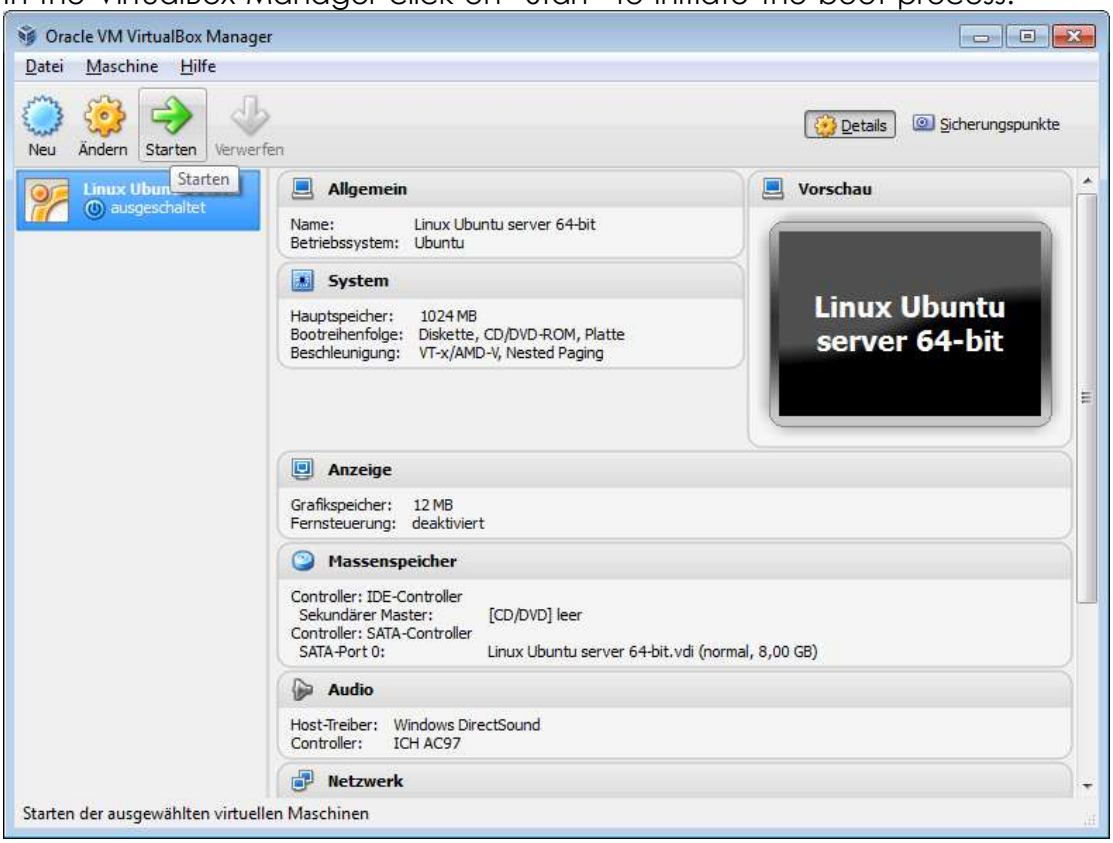
- VirtualBox software (free): https://www.virtualbox.org/wiki/Downloads

You have downloaded the DataWiki VirtualBox distribution that is present in a file named:

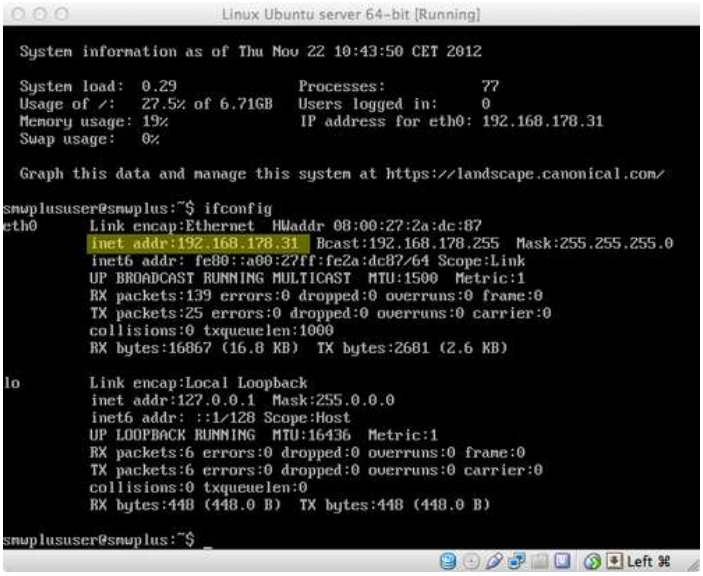
- "Linux Ubuntu server with DataWiki 64-bit.zip"

1.2.2 Installation and Configuration

1. Install VirtualBox on your machine. This can be Windows, Linux or any other operating system which is supported by VirtualBox.
2. Run the "VirtualBox Manager"
3. Extract the zip archive and open the contained .**vbox** file. This will add the image to the VirtualBox Manager.
4. In the VirtualBox Manager click on "Start" to initiate the boot process:



5. The virtual machine will now boot. Once this process is completed, you will be asked to login. Use these credentials:
   - Wikiserver login: smwplususer
   - Password: m8nix
6. Determine the IP address of your virtual machine by entering into the console log:
   - ipconfig
7. In the output, you find the IP adress under the *eth0* section right after *inet addr*:. In our example (see picture below), this is 192.168.178.31



8. Open your browser (on your own machine) and enter the IP address of the virtual machine, followed by /mediawiki. In our example, the URL would be 192.168.178.31/mediawiki. You will be forwarded to the DataWiki main page.

⚠ Important Note:

There is a bug in the current version of the image which influences the search functionality, such that changes and new pages are not added to the search results. Until we release a bugfix release, please use the following workaround:

- Edit the file /var/www/mediawiki/extensions/EnhancedRetrieval/includes/FacetedSearch/solrproxy.php
  - Make sure that the file is writeable by your user
- In line 44, replace the port number 8080 with the new value 8985
  - Save the file
- If you want to see your changes in the search results, you need to update the index manually
  - Go to the directory /home/smwplususer/solr/wiki
  - Run the command php createIndex.php
  - This executes a one-time update. If you need to refresh your index at another point, please re-execute this command

1.2.3 Additional Information about the Installation

- When the system is started, a Solr server that listens to port 8983 is also started. This service is needed for the faceted search. The index is built at boot time.
- If you use the wiki within a productive environment, you should run the indexing script as a cronjob.
- If you have a large wiki, you can update only the index, instead of rebuilding it from scratch.

- This image also includes a TripleStore Basic (contains Jena), which can be started at system start-up (see below).
- The wiki will be initialized automatically.
- Pressing the *Ctrl + Alt* key combination returns you to your computer while pressing the *Ctrl + G* key combination lets you give input to your virtual machine.

1.2.4 Users and Passwords

Note:

To login to the wikiserver, or to enter the commands mentioned in this article, you can either use the console or the SSH login. They both have the same effect, though the SSH console usually has your system's keyboard layout already set such that you do not have to make any alteration.

The system contains a normal user `wiki` and the `root` account. Only the wiki user can log in initially, but once logged in, one can become a root user as described below.

This is how you log in to the Wikiserver as a **wiki user:**

- Wikiserver login: `smwplususer`
- Password: `m8nix`

Once logged in as a wiki user you can **successively** log in again and become a root user by entering the following command:

```
sudo su
```

The database credentials can be found in the configuration file

`/etc/mediawiki/LocalSettings.php`

The database's root account has the same password as that of the wiki user.

Within the wiki, there is an account for the **WikiSysop**, which uses the following credentials:

Wikiserver login: `WikiSysop`
Password: `root`

Note:

We strongly recommend that you change all passwords, including those of MySQL, before you use this installation in a productive environment.

1.2.5 Changing Keyboard Layout

The default keyboard layout is US-International.
If you use a different keyboard layout, you can gain access to the VMware image via an SSH login instead of using the login of the console within the VMware product. This is so because the SSH login usually has your system's keyboard layout preconfigured, so that you don't need to configure this manually as described below.
However, if you intend to use the VMware console, you can change the keyboard layout after you login as a root user as described above. Once you do this, you should issue this command on the VMware console:

```
loadkeys de
```

This will for example load the German layout. You can accordingly enter another code instead of "*de*", in line with your desired layout.
To change the keyboard layout at boot time, become root user and issue this command:

```
install-keymap de
```

This will install the German layout. You can again accordingly enter another code instead of "*de*", in line with your desired layout.

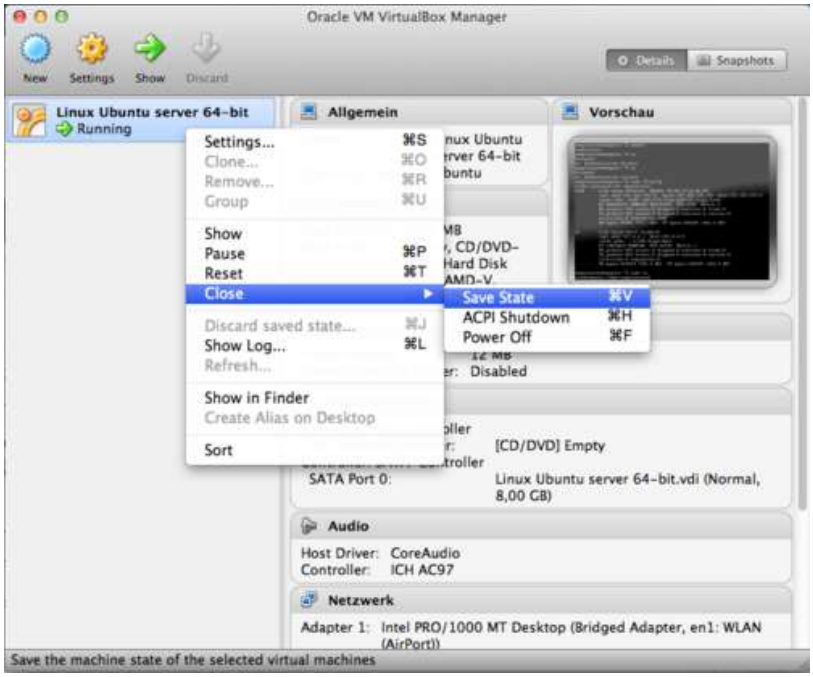
1.2.6 Shutting Down VirtualBox

There are two ways of shutting down VirtualBox:

1. In the *VirtualBox Manager*, right click the virtual machine and select Close -> In this way, the current state of your virtual machine is saved. When you run the machine again at a later point of time, the system will boot and proceed from the state saved in the previous session.
2. As root, enter

```
sudo halt
```

   into the VirtualBox Console. In this way, the current state of your virtual machine is **not** saved. When you start the machine again at a later point of time, the system will boot normally and you might have to restart scripts and services that have been added manually.



1.3 Installation from Source

Important Note:
As of this version, this section might contain incomplete or wrong information. We plan to review and update this section with the next revision of the document.

1.3.1 Installing DataWiki on top of MediaWiki

If you have a running MediaWiki installation and want to use DataWiki simply first install Semantic MediaWiki according to the instructions given at

http://semantic-mediawiki.org/wiki/Help:Installation_1.7.1

Once Semantic MediaWiki is installed and properly configured, continue to install DataWiki following the instructions provided in the next subsection.

1.3.2 Installing DataWiki on top of Semantic MediaWiki

1.3.2.1 Requirements

A web server such as Apache, PHP, MySQL, MediaWiki, and Semantic MediaWiki are properly installed.

Note:

DataWiki 1.7.x works only with MySQL 5.x, PHP 5.3.3, MediaWiki 1.17.0 and SMW 1.7.1

1.3.2.2 Installation Procedure

In the following instructions we will use `<mediawiki-dir>` to represent the location of your specific installation directory of MediaWiki. On Windows

system this might be the directory `C:\xampp\htdocs\mediawiki`. Change the commands accordingly.

1. **From our SourceForge site[1] download the following files:**
   a. smwhalo-1.7.x.zip
   b. arclibrary-1.7.x.zip
   c. enhancedretrieval-1.7.x.zip
   d. scriptmanager-1.7.x.zip
   e. wikiadmintool-1.7.x.zip
   f. wysiwyg-1.7.x.zip
2. **Extract all the downloaded archives.** Each archive contains one subfolder under the extensions directory. Copy all these folders into `<mediawiki-dir>/extensions`.
3. The SMWHalo extension also creates a "skins" folder. Copy its contents into `<mediawiki-dir>/skins`.
4. The WikiAdminTool extension contains a folder "deployment". Copy all its content into a new folder in the MediaWiki root folder `<mediawiki-dir>/deployment`.
5. **Edit the configuration file `<mediawiki-dir>/LocalSettings.php`:**
   - Add the following code in front of other extensions (if there are some installed), to let the Wiki Administration Tool check and update the installed extensions if needed:

```
/*start-wikiadmintool*/
require_once($IP.'/deployment/Deployment.php');
/*end-wikiadmintool*/

/*start-scriptmanager*/
require_once($IP.'/extensions/ScriptManager/SM_Initialize.php');
/*end-scriptmanager*/

/*start-arclibrary*/
include_once($IP.'/extensions/ARCLibrary/ARCLibrary.php');
/*end-arclibrary*/

/*start-applicationprogramming*/
require_once("$IP/extensions/ApplicationProgramming/ParserFunction
s/ParserFunctions.php" );
require_once("$IP/extensions/ApplicationProgramming/Variables/Vari
ables.php" );
include_once("$IP/extensions/ApplicationProgramming/URLArguments/U
RLArguments.php");
$wgPFEnableStringFunctions = true;
```

   - Then add the following code after the Semantic MediaWiki call:

```
/*start-smwhalo*/
include_once('extensions/SMWHalo/includes/SMW_Initialize.php');
enableSMWHalo();
$wgUseAjax=true;
/*end-smwhalo*/

/*start-wysiwyg*/
require_once($IP."/extensions/WYSIWYG/WYSIWYG.php");
$wgCKEditorUrlparamMode = true;
/*end-wysiwyg*/
```

[1] http://sourceforge.net/projects/datawiki/files/?source=navbar

```
$wgGroupPermissions['user']['annotate']=true;
$wgGroupPermissions['user']['edit']=true;
$wgGroupPermissions['user']['wysiwyg']=true;
$wgGroupPermissions['sysop']['userrights']=true;
$wgGroupPermissions['sysop']['ontologyediting'] = true;

$smwgNamespacesWithSemanticLinks[NS_TEMPLATE] = true;
$smwgNamespacesWithSemanticLinks[NS_CATEGORY] = true;
$smwgNamespacesWithSemanticLinks[SMW_NS_PROPERTY] = true;
$smwgNamespacesWithSemanticLinks[SF_NS_FORM] = true;
$smwgNamespacesWithSemanticLinks[NS_HELP] = true;
$smwgNamespacesWithSemanticLinks[NS_HELP_TALK] = true;
$smwgNamespacesWithSemanticLinks[NS_TEMPLATE] = true;
$smwgNamespacesWithSemanticLinks[NS_MEDIAWIKI] = true;

/*start-enhancedretrieval*/
require_once($IP.'/extensions/EnhancedRetrieval/includes/EnhancedR
etrieval.php');
$fsgEnableIncrementalIndexer=true;
/*end-enhancedretrieval*/
```

6. **Add the WikiAdminTool to the PATH environment variable:**

   On Windows:
   - Right-click My Computer > Properties
   - Click Advanced tab
   - Click Environment variables
   - In the section System Variable select the variable "PATH"
   - Click Edit
   - Add a semicolon ";" after the last value and then enter the appropriate path: `<mediawiki-dir>\deployment\tools`
   - Click OK and close the window

   On Linux:
   - Enter this into the console
     ```
     PATH=$PATH: <mediawiki-dir>/deployment/tools
     export PATH
     ```

7. **Grant write access to the following directories for the Wiki Administration Tool**: Make sure that the user who runs the web server can write into the given folders. It is of course not appropriate for the web server to run on root due to security issues. It should use its own user which is pre-configured by the Linux distribution. This user is called "www-data" on Debian/Ubuntu for instance (It is a Linux user, not a wiki user). It should own the files in the wiki directory. To change this you can use these commands:
   ```
   cd <mediawiki-dir>
   chown -R www-data:www-data *
   ```

> **Note:**
>
> Make sure that the files and folders in the `<mediawiki-dir>` can be accessed by the web server and the Wiki Administration Tool at all times.

The credentials used to login into the web-GUI of the wiki admin tool are wiki credentials (all sysop accounts can be used). This is independent of the Linux users. The following files and directories must be writable by the web-server.

- `<mediawiki-dir>\LocalSettings.php`
- `<mediawiki-dir>\deployment\tools\webadmin\sessiondata`
- `<mediawiki-dir>\deployment\tools\webadmin`
- `<mediawiki-dir>\deployment\tools\repositories`
- `<mediawiki-dir>\extensions`
- `%TEMP%`

8. **Upgrade database tables via command line**:
   - Open a command line window in Admin mode
   - Navigate to your MediaWiki installation, i.e. cd <mediawiki-dir>
   - To setup the database schema for DataWiki and to install some wiki pages enter
     ```
     php extensions\SMWHalo\maintenance\SMW_setup.php
     ```
9. **Patch now Mediawiki and SMW with the delivered patches**
   - Therefore navigate in your command line window in the SMWHalo directory:
     ```
     cd <mediawiki-dir>\extensions\SMWHalo
     ```
   - Execute the following commands:
     ```
     php patch.php -p patch_for_MW1.17.0.txt -d <mediawiki-dir> --onlypatch
     php patch.php -p patch_for_SMW1.17.1.txt -d <mediawiki-dir> --onlypatch
     ```
10. **Optionally change the skin:**
    In LocalSettings.php you can change the skin to be the new skin delivered with DataWiki providing a professional look and feel with customizable menus and more.
    ```
    $wgDefaultSkin='ontoskin3';
    ```

Now you have installed DataWiki

> **Note:**
>
> The configuration of the LocalSettings.php above does not correspond to the configuration that comes with the Windows installer tool. It's a slimmed but working configuration. The special page 'CheckInstallation' will output error messages, because the extensions are installed without a deploy descriptor.

1.3.3 Installing DataWiki using the Wiki Administration Tool

1.3.3.1 Requirements

A web server such as Apache, PHP, MySQL, MediaWiki, and Semantic MediaWiki are properly installed.

> **Note:**
>
> DataWiki 1.7.x works best with MySQL 5.x, PHP 5.3.3, MediaWiki 1.17.0 and SMW 1.7.1

Additionally the PHP cURL extension is required. To find out if cURL is installed check the web[2].

1.3.3.2 Installation Procedure

In the following instructions we will use `<mediawiki-dir>` to represent the location of your specific installation directory of MediaWiki. On Windows system this might be the directory `C:\xampp\htdocs\mediawiki`. Change the commands accordingly.

1. Download the file 'wikiadmintool-1.7.x' : http://sourceforge.net/projects/datawiki/files/?source=navbar

2. Extract the downloaded archive.

3. It contains a folder "deployment". Copy all its content into a new folder in the MediaWiki root folder `<mediawiki-dir>/deployment`.

4. On Linux: make "smwadmin.sh" executable. In a command shell navigate to the installation directory of the Wiki Administration Tool and make the script executable:
   ```
   cd <mediawiki-dir>/deployment/tools/
   chmod +x smwadmin.sh
   ```

5. Add the WikiAdminTool to the PATH environment variable:

   On Windows:
   - Right-click My Computer > Properties
   - Click Advanced tab
   - Click Environment variables
   - In the section System Variable select the variable "PATH"
   - Click Edit
   - Add a semicolon ";" after the last value and then enter the appropriate path: `<mediawiki-dir>\deployment\tools`
   - Click OK and close the window

   On Linux:
   - Enter this into the console
     ```
     PATH=$PATH: <mediawiki-dir>/deployment/tools
     export PATH
     ```

6. Make sure that the PHP executable is also set in the system path (either Windows or Linux).

7. Make sure that the MySQL executable is also set in the system path (either Windows or Linux).

8. Edit the configuration file `<mediawiki-dir>/LocalSettings.php`
   a. Verify that it contains the credentials for the administrator-account of your mysql installation. If not, replace the example

[2] http://www.wallpaperama.com/forums/how-to-find-out-if-php-is-compiled-with-curl-extension-installed-enabled-t1576.html

values given for `$wgDBuser` (e.g. 'wikiadmin') and `$wgDBpassword` (e.g 'adminpass') with the name and password of the administrator user in your MYSQL installation.

b. Additionally the credentials for the database administrator have to be added, too, .e.g.

```
$wgDBadminuser = 'root';
$wgDBadminpassword = 'm8nix';
```

c. Add the following code in front of every other extension (if there are some installed), to let the Wiki Administration Tool check and update the installed extensions if needed:

```
/*start-deployment*/
require_once($IP.'/deployment/Deployment.php');
/*end-deployment*/;
```

d. Then add the following code after the Semantic MediaWiki call:

```
$smwgNamespacesWithSemanticLinks[NS_TEMPLATE] = true;
$smwgNamespacesWithSemanticLinks[NS_CATEGORY] = true;
$smwgNamespacesWithSemanticLinks[SMW_NS_PROPERTY] = true;
$smwgNamespacesWithSemanticLinks[SF_NS_FORM] = true;
$smwgNamespacesWithSemanticLinks[NS_HELP] = true;
$smwgNamespacesWithSemanticLinks[NS_HELP_TALK] = true;
$smwgNamespacesWithSemanticLinks[NS_TEMPLATE] = true;
$smwgNamespacesWithSemanticLinks[NS_MEDIAWIKI] = true;
```

9. Grant write access to the following directories for the Wiki Administration Tool: Make sure that the user who runs the web server can write into the given folders. It is of course not appropriate for the web server to run on root due to security issues. It should use its own user which is pre-configured by the Linux distribution. This user is called "www-data" on Debian/Ubuntu for instance (It is a Linux user, not a wiki user). It should own the files in the wiki directory. To change this you can use these commands:

```
cd <mediawiki-dir>
chown -R www-data:www-data *
```

Note:

Make sure that the files and folders in the `<mediawiki-dir>` can be accessed by the web server and the Wiki Administration Tool at all times.

The credentials used to login into the web-GUI of the wiki admin tool are wiki credentials (all sysop accounts can be used). This is independent of the Linux users. The following files and directories must be writable by the web-server.

```
<mediawiki-dir>\LocalSettings.php
<mediawiki-dir>\deployment\tools\webadmin\sessiondata
<mediawiki-dir>\deployment\tools\webadmin
<mediawiki-dir>\deployment\tools\repositories
<mediawiki-dir>\extensions
%TEMP%
```

10. Open a command line interface in **Admin mode**

11. Navigate to the directory of the Wiki Administration Tool Admin tool:

```
cd <mediawiki-dir>\deployment\tools>
```

12. Enter the following command to check if the Wiki Administration Tool is installed correctly:
```
smwadmin
```

13. If you get a list of commands, then everything is fine and you can process by typing the command to install Semantic MediaWiki:
```
smwadmin -i smw
```

> **Note:**
>
> The tool asks you to create a restore point before processing the installation. This is highly recommended, you can restore old installations from a restore point using the command:
> `smwadmin -r <name of restore point>`

14. Type the following command to install the SMWHalo extension incl. its dependencies:
```
smwadmin -i smwhalo
```
When the system tells you to type in the location of your wiki, you can enter any value, e.g. 'localhost'. It is only used for RDF exports made by Semantic MediaWiki.

15. Type `smwadmin --finalize` to finalize the installation

16. To install the Enhanced retrieval extension providing faceted search features type:
```
smwadmin -i enhancedretrieval
```

17. To install the the WYSIWYG extension type:
```
smwadmin -i wysiwyg
```

18. Install the **SMWplus content bundle** and the **SMWplus sandbox**, then you will get some forms, templates and pre-configured menus
```
smwadmin -i Smwplus
smwadmin -i Smwplussandbox
```

19. Start Apache and MySQL
20. Open your wiki in the browser
21. Login as WikiSysop (default credentials are: WikiSysop/root)
22. Go the special page Special:SMWHaloAdmin
23. Click Initialize

Now you have installed DataWiki

1.3.4 Installing Solr Server

The Faceted Browsing feature needs a new index that is based on the triples stored in the wiki. This index for semantic data uses Apache Solr and contains information about the triples like subject name, property name, property value, namespace and category. Solr can index documents with arbitrary fields. These fields can be used as facets for faceted browsing.

Note:

If you have installed DataWiki using the Windows Installer, Solr is already installed and configured.

1.3.4.1 Requirements

- **Java 1.5** or greater. Download via Sun: http://www.oracle.com/technetwork/java/javase/downloads/index.html.
  - Note: Gnu's GCJ is not supported and does not work with Solr.
- **PHP cURL** extension
- **Enhanced Retrieval** extension has to be installed
- **Solr release** – For Windows and Linux download from Sourceforge: http://sourceforge.net/projects/datawiki/files/?source=navbar

1.3.4.2 Installing with Wiki Admin Tool

SOLR is now part of the DIQA repository. Users can install it via the console or the the web interface. For Wiki Administration Tool it is supposed to be an external application just like the TSC. That means it requires an entry in `config/externalapps` to let WAT know where it is located, for example:

```
solr=c:\SMWPLUS\solr
```

The Windows installer automatically adds this entry. Users who want to update from DataWiki 1.6 have to add this entry manually and install the solr-package manually because in 1.6 SOLR was not known as a bundle.

Service scripts

SOLR is now registered as Windows service during installation. The SOLR bundle provides two maintenance scripts for installing/uninstalling SOLR. They are mainly for users who do not use the Windows installer.

- `installAsService.bat`
- `uninstallAsService.bat`

Note:

Note:

The cmd scripts expect a 64-bit Windows (and Java) being installed. In case that it is 32-bit, please adjust the variable `PLATFORM` to the value "`win32`" in `installAsService.bat`.

1.3.4.3 Installing via Sourcecode

1. Unpacking

Unzip the Solr release to the directory of your wiki, for example `C:\xampp\solr`

2. Configuring smwdb-data-config.xml

Go to directory `solr\wiki\solr\conf` and edit the file `smwdb-data-config.xml`. This is a new file that defines how the data is imported into the index. Note the specification of the database at the beginning of the file:

```
<!DOCTYPE dataConfig [
    <!ENTITY DBPREFIX ""> <!-- You can specify a MediaWiki
database prefix here. -->
]>
<dataSource
      type="JdbcDataSource"
      driver="com.mysql.jdbc.Driver"
      url="jdbc:mysql://{{wgDBserver}}:{{wgDBport}}/{{wgDB
name}}"
      user="{{wgDBuser}}"
      password="{{wgDBpassword}}"
      batchSize="-1"
      />
```

Add the credentials of your wiki database, see example below:

```
<!DOCTYPE dataConfig [
    <!ENTITY DBPREFIX "mw_"> <!-- You can specify a
MediaWiki database prefix here. -->
]>
<dataSource
      type="JdbcDataSource"
      driver="com.mysql.jdbc.Driver"
      url="jdbc:mysql://localhost:3306/semwiki_en"
      user="root"
      password="m8nix"
      batchSize="-1"
      />
```

If Solr is running on the same server as the wiki, you may use `url="jdbc:mysql://localhost:3306/semwiki_en"`, otherwise add the URL of the Solr server.

3. Configuring the Faceted Search

The following settings are the default settings. They are defined in `FS_Settings.php`. Normally you do not have to change them. The indexer (i.e. SOLR) is by default addressed by a proxy (i.e. `solrproxy.php`). Thus SOLR can be hidden for the public.

```
$fsgFacetedSearchConfig = array(
    'indexer' => 'SOLR',
    'source'  => 'SMWDB',
    'proxyHost'    => $wgServer,
    'proxyServlet' =>
"$wgScriptPath/extensions/EnhancedRetrieval/includes/Facet
edSearch/solrproxy.php",
    'indexerHost' => 'localhost', // must be equal to
$SOLRhost in solrproxy.php
    'indexerPort' => 8983         // must be equal to
$SOLRport in solrproxy.php
);
```

Note:

You only need to initialize `fsgFacetedSearchConfig` when using another configuration

Explanation of the settings
The array configures the indexer that is used for faceted search. Please note the distinction between the proxy and indexer settings. The proxy is a PHP script that communicates with the browser on the one side and the indexer (SOLR) on the other side. The indexer contains the actual data about the content of the wiki. It is normally only accessible by the proxy.

The configuration has the following key-value pairs:

- **`indexer`**: Type of the indexer. Currently only 'SOLR' is supported.
- **`source`**: The source for indexing semantic data. Currently only the database of SMW is supported: 'SMWDB'
- **`proxyHost`**: Protocol and name or IP address of the proxy to the indexer server as seen from the client e.g. 'http://www.mywiki.com' or $wgServer
- **`proxyPort`**: The port number of the indexer server e.g. 8983 as seen from the client. If the solrproxy is used this can be omitted.
- **`proxyServlet`**: Servlet of the indexer proxy as seen from the client.
    - If the solrproxy is used it should be `$wgScriptPath/extensions/EnhancedRetrieval/includes/FacetedSearch/solrproxy.php`.
    - If the indexer is addressed directly it should be '`/solr/select`' (for SOLR)
- **`indexerHost`**: Name or IP address of the indexer server as seen from the wiki server e.g. 'localhost'. If the solrproxy is used and the indexer host (SOLR) is different from 'localhost', i.e. SOLR is running on another machine than the wiki server, the variable $SOLRhost must be set in solrproxy.php.
- **`indexerPort`**: The port number of the indexer server e.g. 8983 as seen from the wiki server. If the solrproxy is used and the port of the indexer

host (SOLR) is different from 8983, the variable $SOLRport must be set in solrproxy.php.

Note:

The default settings are defined in the file FS_Settings.php. You can overwrite them in LocalSettings.php after the inclusion of Enhanced Retrieval.

Proxy

Accessing SOLR directly via its standard port 8983 is sometimes not possible. Normally you should use the SOLR proxy on port 80, which is the default.

Important Note:

For security reasons we recommend to use a proxy on public servers. If SOLR is accessible to the public, hackers can manipulate or destroy your index. You should block the SOLR port (8983) on these servers.

If you still want to address SOLR directly you have to modify the original configuration slightly. Please note the additional protocol http:// in front of the host name.

```
$fsgFacetedSearchConfig = array(
    'indexer' => 'SOLR',
    'source'  => 'SMWDB',
    'proxyHost'    => 'http://host where SOLR is located',
// e.g. $wgServer
    'proxyPort'    => 8983,
    'proxyServlet' => "/solr/select",
    'indexerHost' => 'localhost', // must be equal to
$SOLRhost in solrproxy.php
    'indexerPort' => 8983         // must be equal to
$SOLRport in solrproxy.php
);
```

The proxy script is located in `/extensions/EnhancedRetrieval/includes/FacetedSearch/solrproxy.php`. It tries to access SOLR on localhost at port 8983. If you want to change this, you have to adapt the corresponding settings in that file:

```
$SOLRhost = 'localhost';
$SOLRport = 8983;
```

If you are running the wiki and Solr not on localhost, note that the $wgServer should contain the domain name or IP where your wiki is accessible. In some installations MediaWiki might not be able to determine the correct server

name. You can set $wgServer directly by assigning the correct value somewhere at the begining of the LocalSettings.php:

```
$wgServer = "http://yourdomain.com";
```

4. Configuring solrconfig.xml (optionally)
Go to directory `solr\wiki\solr\conf` and edit the file `solrconfig.xml`

This is the general configuration file for the Solr server. For the full import of semantic data from the SMW database a Data-Import-Handler has to be registered by adding the following lines:

```
<requestHandler name="/dataimport"
class="org.apache.solr.handler.dataimport.DataImportHandle
r">
<lst name="defaults">
      <str name="config">smwdb-data-config.xml</str>
</lst>
</requestHandler>
```

5. Starting Solr with the wiki configuration
Navigate to the folder of Solr, in this example xampp\solr\wiki and run:

- startSolr.bat (on Windows)
- startSolr.sh (on Linux)

After Solr is started the index is created automatically.
At any time you may also recreate the full search index from the wiki data by executing the script:

```
php createIndex.php
```

Note:

This Solr example server references SolrCell jars outside of the server directory with <lib> statements in the solrconfig.xml. If you make a copy of this example server and wish to use the ExtractingRequestHandler (SolrCell), you will need to copy the required jars into solr/lib or update the paths to the jars in your solrconfig.xml.

Important links

Solr administration page is available under the URL:
http://localhost:8983/solr/admin/

A small development environment for the data import is available at:
http://localhost:8983/solr/admin/dataimport.jsp?handler=/dataimport

1.3.4.4 Advanced Configuration

If you use a Solr release from Apache Download mirrors, you have to add the configuration files manually. Two configuration files connect Solr with the wiki.

solr/conf/solrconfig.xml

This is the general configuration file for the Solr server. For the full import of semantic data from the SMW database a Data-Import-Handler has to be registered by adding the following lines:

```
<requestHandler name="/dataimport"
class="org.apache.solr.handler.dataimport.DataImportHandle
r">
    <lst name="defaults">
      <str name="config">smwdb-data-config.xml</str>
    </lst>
</requestHandler>
```

solr/conf/schema.xml

The schema defines the fields of each document that is indexed. Each wiki article becomes an indexed document in Solr and semantic data is stored in fields of the document.

Some fixed fields are added for general semantic meta-data:

```
<!--
  Fields for SMW
-->
<field name="smwh_categories" type="text_ws"
indexed="true" stored="true" multiValued="true"
omitNorms="true" />
<field name="smwh_properties" type="text_ws"
indexed="true" stored="true" multiValued="true"
omitNorms="true" />
<field name="smwh_attributes" type="text_ws"
indexed="true" stored="true" multiValued="true"
omitNorms="true" />
<field name="smwh_title"      type="text_ws"
indexed="true" stored="true" omitNorms="true" />
<field name="smwh_namespace_id" type="int" indexed="true"
stored="true" omitNorms="true" />
```

Semantic relations and attributes of an article have to be stored as well. But as their names are neither fixed nor known from the beginning, we need dynamic fields for storing their names and values. The names of the fields depends on the type of the relations (string) and attributes (number, data, string, text,...). Dynamic fields are described by regular expressions that are mapped to data types:

```
<dynamicField name="*_i"  type="int"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_s"  type="string"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_l"  type="long"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_t"  type="text"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_b"  type="boolean"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_f"  type="float"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_d"  type="double"
indexed="true"  stored="true" multiValued="true"/>

<dynamicField name="*_dt" type="date"
indexed="true"  stored="true" multiValued="true"/>
```

Solr can index fields in different ways. For example, strings can be tokenized or be processed as a whole. To benefit from different kinds of indexes, Solr can copy fields automatically. We store strings both as strings and tokenized text:

```
<copyField source="*_t" dest="*_s"/>
```

Now you can proceed with creating a new index

solr/conf/smwdb-data-config.xml
This is a new file that defines how the data is imported into the index. See DIHQuickStart and Data Import Request Handler for further details.

This is the content of the configuration:

```
<dataConfig>
  <dataSource
    type="JdbcDataSource"
    driver="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost:3306/testdb"
    user="root"
    password="mydatabasepassword"
    batchSize="-1"
    />
  <script>
        <![CDATA[
          function Relations(row) {
            // The values of all properties are stored as string.
            var prop = 'smwh_'+row.get('prop')+'_t';
            var val = row.get('obj');
            row.remove('prop');
            row.remove('obj');
            // Add the property and its value
            row.put(prop, val);
```

```
          // Store the names of all properties in the article
          row.put('smwh_properties', prop);
          return row;
        }

        function TextAttributes(row) {
          // The values of all text attributes are stored as string.
          var attr = 'smwh_'+row.get('attr')+'_xsdvalue_t';
          var text = row.get('text');
          row.remove('attr');
          row.remove('text');
          // Add the attribute and its value
          row.put(attr, text);
          // Store the names of all attributes in the article
          row.put('smwh_attributes', attr);
          return row;
        }

        function Attributes(row) {
          var prop    = row.get('prop');
          var valXSD  = row.get('valueXSD');
          var valNum  = row.get('valueNum');
          var valUnit = row.get('valueUnit');
          var type    = row.get('type');
          row.remove('prop');
          row.remove('valueXSD');
          row.remove('valueNum');
          row.remove('valueUnit');
          row.remove('type');
          var typeSuffix = 't';
          var isNumeric = false;
          if (type == '_dat' ||
            prop == 'Modification_date' ||
            prop == 'Creation_date') {
            // Given format of a date: 1995/12/31T23:59:59
            // Required format: 1995-12-31T23:59:59Z
            var dateTime = valXSD.split("T");
            var date = dateTime[0];
            date = date.replace('/', '-');
            time = dateTime.length > 1 ? dateTime[1] : '00:00:00';
            valXSD = date + 'T' + time + 'Z';
            typeSuffix = 'dt';
          } else if (
            type == '_txt' ||
            type == '_cod' ||
            type == '_str' ||
            type == '_ema' ||
            type == '_uri' ||
            type == '_anu' ||
            type == '_tel' ||
            type == '_tem' ||
            type == '_rec') {
            typeSuffix = 't';
          } else if (type == '_num') {
            typeSuffix = 'd';
            isNumeric = true;
          } else if (type == '_boo') {
            typeSuffix = 'b';
          }

          var propXSD = 'smwh_'+prop+'_xsdvalue_'+typeSuffix;
          row.put(propXSD, valXSD);
          row.put('smwh_attributes', propXSD);
          if (isNumeric) {
            row.put('smwh_'+prop+'_numvalue_d', valNum);
            if (valUnit.length > 0) {
              row.put('smwh_'+prop+'_unit_s', valUnit);
            }
          }

          return row;
        }
      ]]>
</script>
<document>
  <!-- Find all pages in the <page> table. They will be our documents. -->
  <entity
```

```
      name="pages"
      query="SELECT p.page_id as pid,
                   p.page_namespace as pns,
                   CAST(p.page_title AS CHAR) as pt
            FROM page as p
      ">
      <field column="pid" name="id" />
      <field column="pns" name="smwh_namespace_id" />
      <field column="pt"  name="smwh_title" />

      <!-- Store all categories for each subject -->
      <entity
        name="categories"
        query="SELECT CAST(c.cl_to AS CHAR) cat
               FROM categorylinks c
               WHERE cl_from='${pages.pid}'">
        <field column="cat" name="smwh_categories" />
      </entity>

      <!-- Store all SMW IDs for pages -->
      <entity
        name="smwids"
        query="SELECT s.smw_id as smwID
               FROM smw_ids s
               WHERE s.smw_namespace='${pages.pns}' AND
                     s.smw_title='${pages.pt}'">
        <field column="smwID" name="smwh_smw_id" />

        <!-- Store all subject, predicate and object names for each subject of a
             relation. -->
        <entity
          name="rels"
          transformer="script:Relations"
          query="SELECT CAST(pids.smw_title AS CHAR) as prop,
                        CAST(oids.smw_title AS CHAR) as obj
                 FROM smw_rels2 AS r
                 LEFT JOIN (smw_ids as pids) ON (pids.smw_id = r.p_id)
                 LEFT JOIN (smw_ids as oids) ON (oids.smw_id = r.o_id)
                 WHERE r.s_id='${smwids.smwID}'">
        </entity>

        <!-- Store all subject and attribute names and the values for each subject
             of an attribute. -->
        <entity
          name="atts"
          transformer="script:Attributes"
          query="SELECT CAST(pids.smw_title AS CHAR) as prop,
                        CAST(a.value_xsd AS CHAR) as valueXSD,
                        a.value_num as valueNum,
                        a.value_unit as valueUnit,
                        CAST(spec.value_string AS CHAR) as type
                 FROM smw_atts2 AS a
                 LEFT JOIN (smw_ids as pids) ON (pids.smw_id = a.p_id)
                 LEFT JOIN (smw_spec2 as spec) ON (a.p_id = spec.s_id)
                 WHERE a.s_id='${smwids.smwID}' AND
                       (LEFT(spec.value_string,1) ='_' OR spec.s_id IS NULL)">
        </entity>

        <!-- Store all text values for each subject -->
        <entity
          name="text"
          transformer="script:TextAttributes"
          query="SELECT CAST(pids.smw_title AS CHAR) as attr,
                        CAST(t.value_blob AS CHAR) as text
                 FROM smw_text2 AS t
                 LEFT JOIN (smw_ids as pids) ON (pids.smw_id = t.p_id)
                 WHERE t.s_id='${smwids.smwID}'">
        </entity>
      </entity>
    </entity>
  </document>
</dataConfig>
```

Add the specification of the database at the beginning of the file, like described in step 3 in the section **Installing via Sourcecode.**

1.3.4.4.1 Troubleshooting: server can't access the web

Some wiki servers which have no web access, although accessible from outside, need a customization, because they cannot resolve the request to semantic-mediawiki.org made in the javascript part:

```
function Attributes(row) {
       var        swivtNS        =        'http://semantic-
mediawiki.org/swivt/1.0#';
       var prop    = row.get('prop');
       var type    = row.get('type');
       ...
```

The SOLR process starts correctly, but the index is prevented from being built.

Solution

1. Save the content of http://semantic-mediawiki.org/swivt/1.0# in a local file
2. Put the file on wiki server
3. Change the lines above to make reference to localhost instead:

```
function Attributes(row) {
      var swivtNS = 'http://localhost/1.0#';
      var prop    = row.get('prop');
```

Now indexing should start correctly!

1.3.4.5 Verifying the Installation

To verify if the installation and configuration of Solr was successful, go to the special page Special:FacetedSearch and check if you can browse through your data.

1.4 Completing the installation

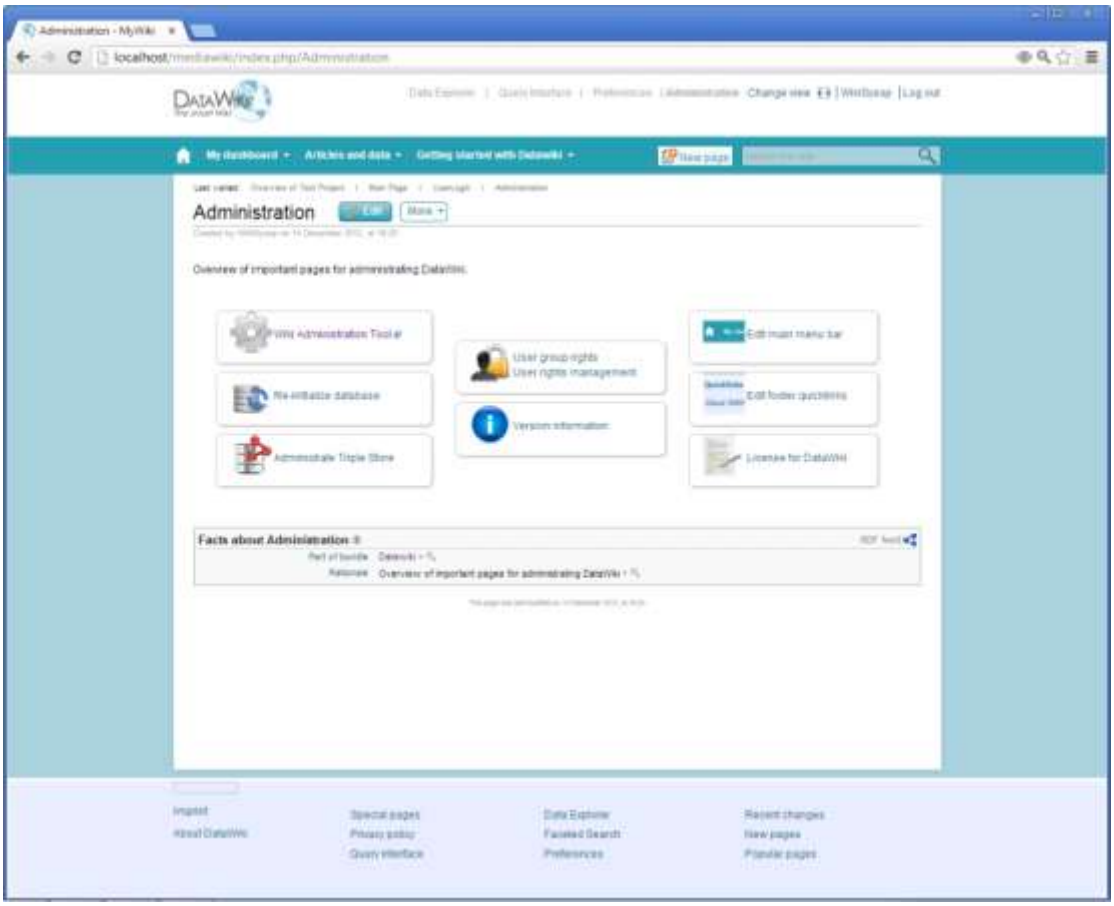
After you have installed DataWiki make sure that your installation is up to date and all needed services are running. It is important that you follow these instructions here before giving other users access to the Wiki or starting productive work. We also highly recommend to repeat these instructions from time to time.

1.4.1 Prerequisites

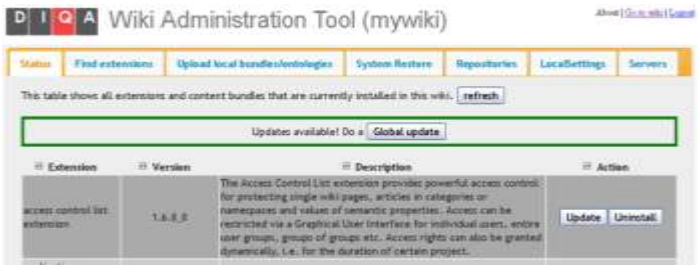
Make sure that you are logged in as a user with administration privileges.

1.4.2 Open Wiki Administration Tool

1. Open your wiki and log in as WikiSysop
2. Click on the Administration-link and select the item "Wiki Administration Tool":



3. The log-in page of the Wiki Administration Tool opens. Enter your credentials again and submit the form.
4. The Web Interface of the Wiki Administration Tool is opened in your browser:



1.4.3 Verify Wiki Services

The Wiki needs the following services (i.e. servers) to properly work: Apache, mySQL, solr and memcached. The service "tsc" is optional.

1. Open the tab **Servers** and check that the status for these servers is "running": Apache, mySQL, solr, memcached.
2. Start (or restart) services that are not running! Click on "Execute" to start such services.

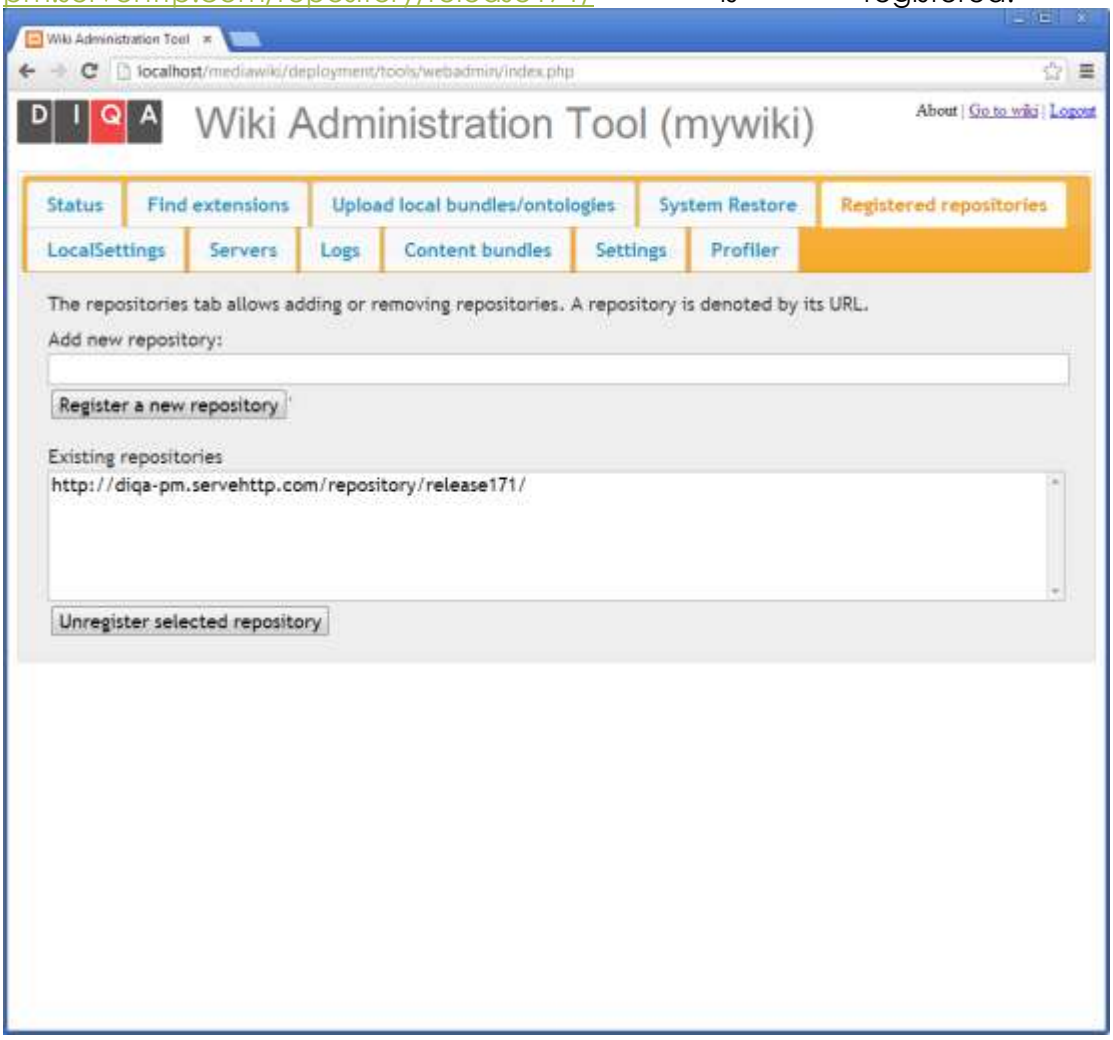
Result: all required services are running: Apache, mySQL, solr and memcached.

1.4.4 Check for software updates

DIQA regularly publishes bugfixes or updates to DataWiki. It is highly recommended to always install the latest upgrades!

1. Open the tab **Registered Repositories** and verify that the repository for DataWiki 1.7.1 that is named http://diqa-

pm.servehttp.com/repository/release171/ is registered:



2. Open the tab **Status** and click on the "Refresh"-button in the tab. If updates are available then this is indicated by the label "Updates available" in the head of the page.
3. If updates are available then click on the button "**Global update**" to download and install all updates to your Wiki.

> ⚠ **Important Note:**
>
> The update process may take 10-15 minutes. Don't interrupt it or work in the Wiki, this might cause damages in the system

Result: Your Wiki installation is now up to date! You have the latest bugfix releases and updates installed.

1.4.5 Create System Restore Point

You should regularly create System Restore Points in order to back up your Wiki installation.

1. Open the tab **System Restore** and enter a name for the system restore point (we recommend to include the date into the name).

2. Click on "**Create**" to initiate the production of the System Restore Point.

> **Note:**
>
> The process to create a System Restore Point may take 10-15 minutes. Don't interrupt it or work in the Wiki!

1.4.6 Install the product key

Make your DataWiki installation a DataWiki Professional installation by purchasing a license key from DIQA.

With purchasing a license key for DataWiki you get product warranty for 1 year and you are entitled to get a support plan.

1. Go to our website and order a license for DataWiki Professional: http://www.diqa-pm.com/de/DataWiki
2. After purchasing the license key you will get an email containing a product key. Save this key (XML file) on your local file system.
3. Go to you Wiki installation
4. Login as WikiSysop
5. Go to the page **Special:Upload**
6. Click **Browse** and select the product key file that you have received and stored in your file system
7. Click **Upload file**

1.5 De-Installation

1.5.1 Deinstall DataWiki Windows Installer Version

1. Open the task manager with administrator privileges on your server and stop these processes:
    a. XAMPP
    b. MySQL
    c. Apache http
    d. SOLR
2. Go to the DataWiki folder and launch `Uninstall.exe` - The default DataWiki folder location is `C:\SMWPLUS` , **OR**
   Use the *Programs* menu to navigate to the DataWiki Program menu and select the `Uninstall` option.
3. The uninstall process will now start showing the uninstall progress. Press **Show details** to display the detailed view of the uninstall process.
4. A warning window will come up during the de-installation process to indicate that you must stop all DataWiki processes before you proceed with the uninstall process.
5. Once the uninstall process is completed, you can click on **Close.**

DataWiki is now successfully uninstalled.

1.5.2 Deinstall DataWiki Virtual Machines

If you want to remove a virtual machine from your computer, please refer to the last section of the respective installation guide.

Before deleting the VirtualBox image, please shutdown the virtual machine as described in the section “Shutting Down VirtualBox” . After the shutdown is complete, switch to the VirtualBox Manager and right-click the DataWiki image. Select *Remove...* and then either Delete all files to wipe the entire image off your system. Please note that all files will be deleted and the data that was in your wiki will be lost.

1.6 Upgrading from DataWiki 1.6.0

Important Note:
As of this version, this section might contain incomplete or wrong information. We plan to review and update this section with the next revision of the document.

We encourage you to always keep your installation of DataWiki up-to-date; This ensures that you have always the latest features and bug fixes.

This section describes how you upgrade your DataWiki installation from version DataWiki 1.6.0 to version DataWiki 1.7.1. You use the Wiki Administration Tool (already part of DataWiki) to perform the upgrade.

1.6.1 Requirements

- You have an installation of DataWiki 1.6.0
- The Wiki Administration Tool is installed (this is already part of DataWiki)

1.6.2 Preparations

Important Note:

A global update with the Wiki Administration menu will overwrite the menus. If you have defined your own menus, it is important to save them before doing the update! Just go to the special page Special:Export and export all pages in the MediaWiki namespace.
The same holds for a modified skin. Make a copy of all the modified skin files before starting an update. Otherwise all skin files will be overwritten.

1.6.3 Update process

Note:

Follow this procedure if you have installed DataWiki via Deployment Framework or via Windows installer and have no console access to the server.

1. Follow the link `<your-wiki-url>/mediawiki/deployment/tools/webadmin/index.php` or navigate in your wiki to **Administration > Wiki Administration Tool** to open the web interface of the Wiki Administration Tool
2. Log in as WikiSysop (default credentials are: WikiSysop, root)

Important Note:

Before attempting an update to a new major release create a backup of your MediaWiki folder and the database.

3. Open the tab Repositories

4. You should see the default DIQA repository. If not, add it: Change the repository again to the default one: http://diqa-pm.servehttp.com/repository/release171/
5. Go to the tab Status
6. You can either update the complete wiki installation by clicking **Global update** OR you can update single extensions by clicking the **Update** button in the corresponding line
7. In case you get an error warning, update first only the Wiki Administration Tool itself (select **deployment**) and after that perform the global update
8. The 'Status tab should now display the latest versions of the extensions from the DataWiki 1.7 branch. This includes SMW 1.7 and MW 1.17.
9. You can also verify this by visiting the special page **Special:Version** and checking the version numbers there

2 Wiki Administration Tool

The Wiki Administration Tool, which is part of DataWiki provides two interfaces to administrate your wiki:

1. Web interface or
2. Windows console

Both methods access repositories from DIQA where the DataWiki software is published. Each DataWiki version has its own repository. The repository for DataWiki 1.7.1 is located in:

```
http://diqa-pm.servehttp.com/repository/release171/
```

2.1 Web Interface of the Wiki Administration Tool

The new Wiki Administration Tool web interface lets you administrate your wiki in a more comfortable and intuitive way - you interact with the web interface instead of giving commands in the command-line. This not only saves you time, but also enhances the usability of the Wiki Administration Tool.

Apart from using the web interface to install new extensions, update and uninstall existing extensions, there is a tool that is meant to help you edit your wiki's LocalSettings.php from the comfort of your web browser - this tool is unique.

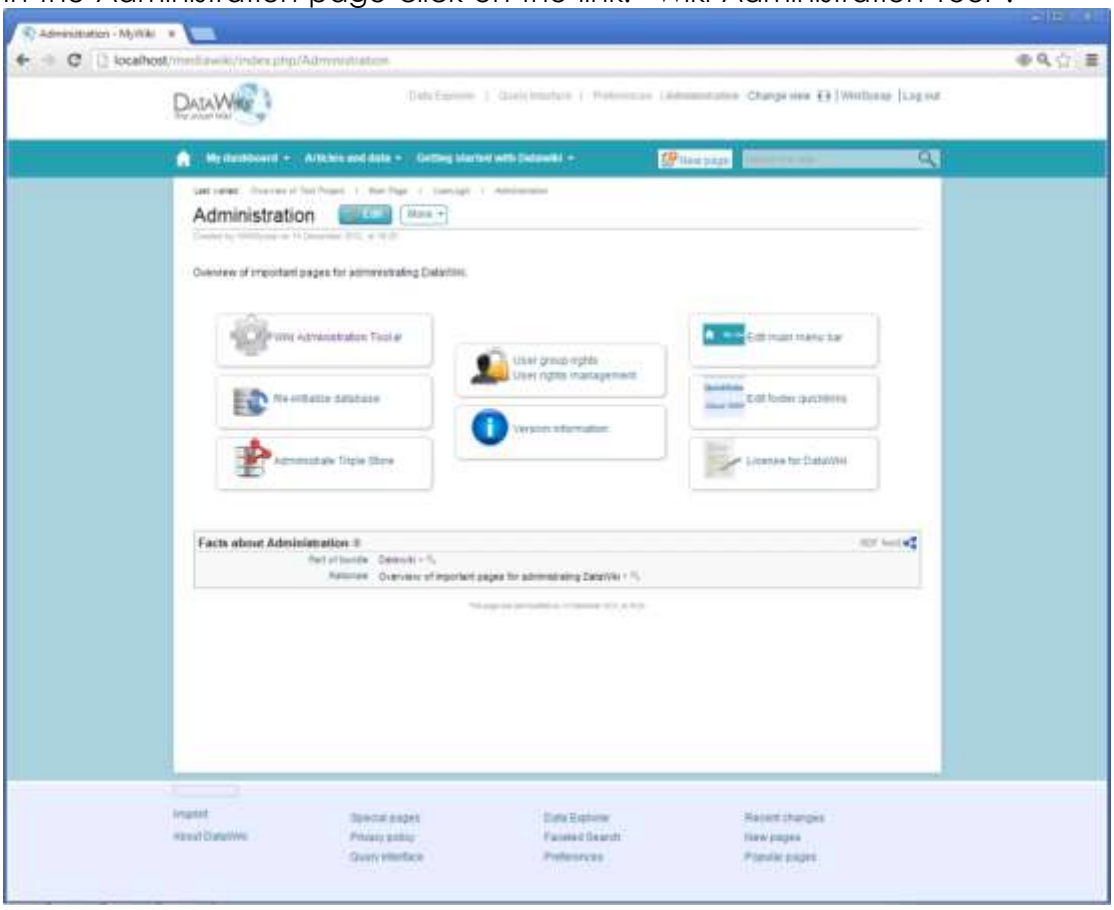
With DataWiki 1.7.1, you can now monitor the status of the servers (Apache, mySQL, solr, tsc and memcached) and start or stop them easily.

The web interface is capable of performing all Wiki Administration Tool operations that were previously done using the console.

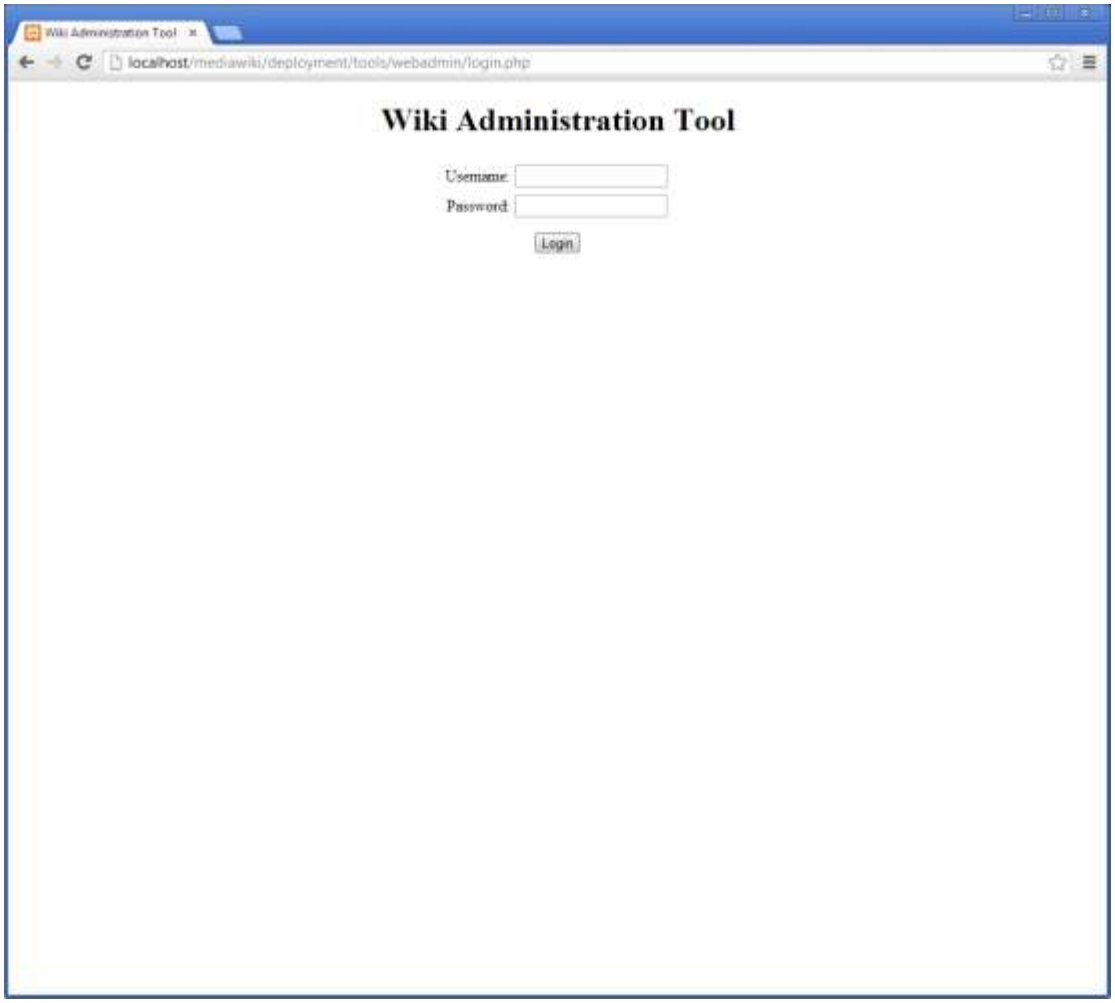
2.1.1 Accessing the Web Interface

1. Log in to your wiki as Administrator.
2. Click on the link named "Administration" in the header of the wiki.

3. In the Administration page click on the link: "Wiki Administration Tool":



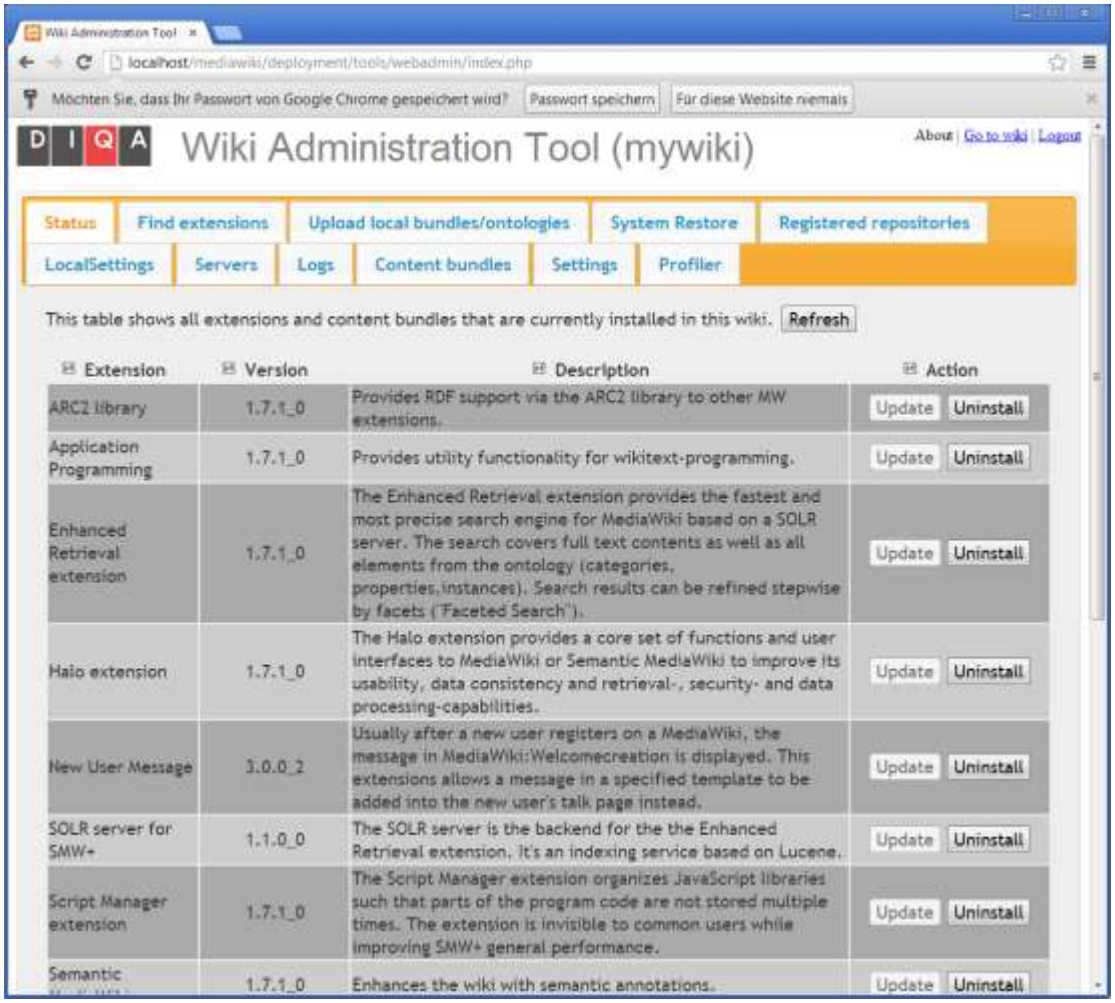
4. Login with the same credentials as those used in step 1 to access the tool:



2.1.2 Overview of the Web Interface

Having logged in, the Wiki administration tool displays a menu bar with different items. The selected menu is in orange. The menu items are:

- Status - view all installed extensions, their description and version and update or uninstall them.
- Find extensions - search and view all extensions, their description and version, check dependencies and install extensions
- Upload local bundles/ontologies - upload local bundles
- System Restore - create or load restore points
- Registered Repositories - manage repositories (add/modify/remove)
- LocalSettings - edit LocalSettings.php file
- Servers - monitor status and Start/Stop servers (Apache, mySQL, solr, tsc and memcached)
- Logs
- Content Bundles
- Settings
- Profiler

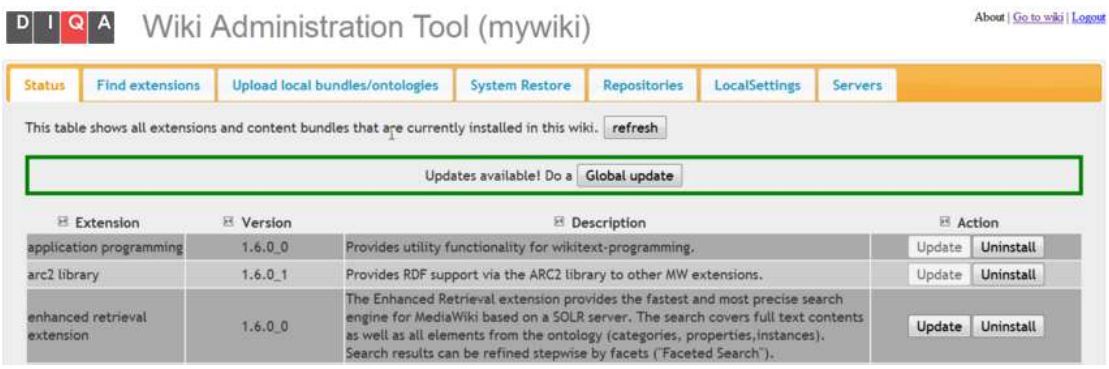


If you have installed large content bundles or extensions that have a lot of dependencies using the Wiki Administration Tool, it might take some time before the selected tab is loaded - just wait a bit as the files are loaded. This delay may be experienced while creating a restore point of a wiki with a lot of data.

Once any of the tabs above is clicked, the web interface expands the view, and retrieves the necessary data letting you perform the desired action.

2.1.3 The Status Menu – Update or Uninstall Extensions

The first menu item Status is the default selection whenever you open the web interface. It generates a list of all installed extensions and content bundles of your wiki, their version, and description and lets you update or uninstall any of the listed entries.



The extensions and bundles are listed alphabetically according to their names with the following columns:

- **Extension** - name of extension.
- **Version** - shows the version of the selected extension
- **Description** - gives a brief description of the selected extension
- **Action column** - has the following buttons:
  - **Update** - update the extension - this button is only activated if there is an update version available.
  - **Uninstall** - uninstall an extension. Button is deactivated for the non-optional extensions, i.e. Wiki Administration Tool and mw. You cannot uninstall these packages with the Wiki Administration Tool - these packages are required!

2.1.3.1 Updating extensions

The **Status** menu will notify you if there are any updates available for the installed extensions. Updates are of two types:

- **Global Update** - applies all available updates
- **Single update** - applies an update on a selected extension.

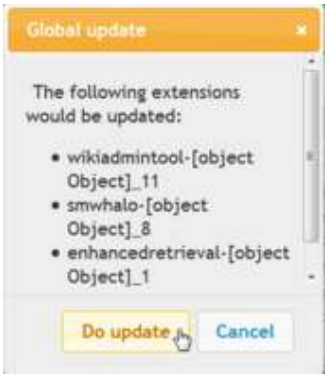
2.1.3.1.1 Performing a Global Update

> ⚠ **Important Note:**
>
> We recommend to deactivate all non-Halo / DataWiki extensions before performing a global update because they might be incompatible to a new MediaWiki version. This can block the update process.

To perform a **Global Update** thereby applying all available updates:

1. Press the **Global update** button. This opens up a popup listing the extensions that will be updated.

2. Press the **Do update** button. This updates the listed extensions



The update progress can be followed in the popup that opens thereafter. Be patient while the system updates the listed extensions - the system will give notice of successful update.

2.1.3.1.2 Updating a single extension

To update a single extension:

1. Press the **Update** button on the respective extension from the **Action** column - this opens up a popup listing the extension that will be updated.
2. Press the **Do update** button - this updates the listed extension

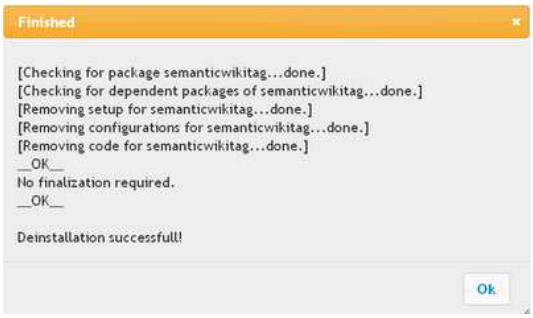
The update progress can be followed in the popup that opens thereafter. Be patient while the system updates the listed extensions - the system will give notice of successful update.

2.1.3.2 Uninstalling extensions

With the Wiki Administration Tool web interface you may uninstall extensions with only two clicks:

1. Click the **Uninstall** button in the corresponding row. This opens up a popup letting you validate the uninstall process.
2. Click **Yes** - the system will automatically uninstall the extension.

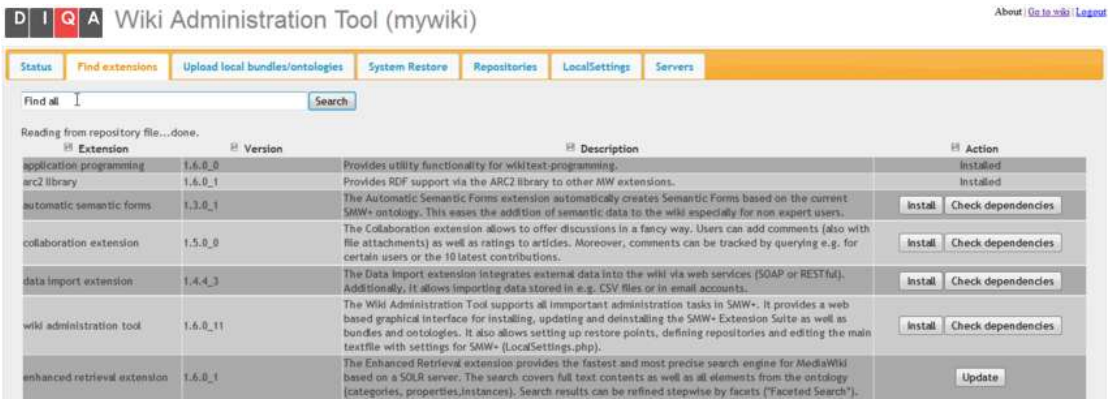
You can follow the progress in the information that is displayed in the popup. Depending to the extension, de-installation might take a while. The system will give notice of successful de-installation.



If you try to uninstall an extension or a bundle where there are other bundles which depend on it, you're prompted and the tool tells which bundles will be removed additionally. You can then decide to continue or not. In the web interface you have a settings for this (default is off, ie. stop with an error).

2.1.4 Find Extensions Menu – Search and Install Wiki Apps and Extensions

The Find extensions section gives you an overview of all extension which are available in the registered repository and lets you install any of these extensions.



2.1.4.1 Searching for Extensions

The following options may be used while searching for extensions:

- **"Find all" search option** - this is the default search option. Pressing the Search button with this option displays a table with all extensions in your repositories.
- **Empty search option** - if you leave the search entry box empty and press the Search button, a table with all extensions in your repositories will be displayed.
- **Enter a string and search** - this option will display all the extensions which contain the string either in their name or in their description.

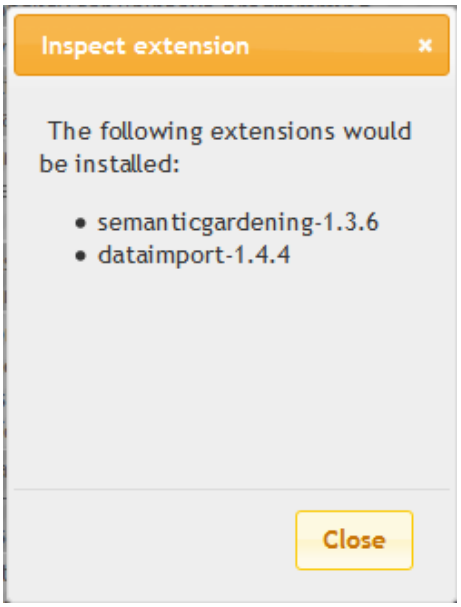
The 4th column of the table that is generated will display an extension as:

- **Installed** - if the extension has been installed
- **Update** - if the extension has been installed and there is an update
- If an extension is not installed - it will have the **Install** and **Check dependencies** buttons

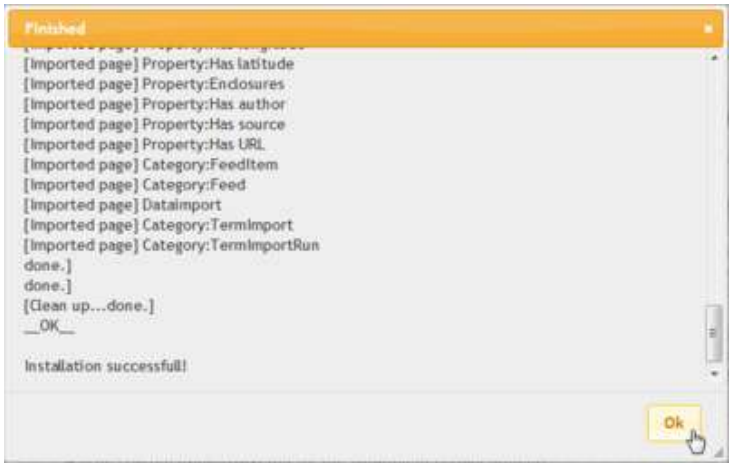
2.1.4.2 Installing Extensions

1. Before installation, we recommend that you first check dependencies - to do this, hit the **Check dependencies** button. This opens up the

Inspect extension popup. It has a list of all the extensions that need to be installed to acquire full functionality of the selected extension.



2. Click **Close**
3. The Installation of multiple bundles at once simplifies setting up a wiki. You have check boxes in front of every bundle in the search tab. You can select an arbitrary set of bundles and WAT will compute the dependency graph and select other required bundles too, if there are any.
4. Click the Install button. This automatically installs the extension together with its dependencies (these are other extensions which are required to attain full functionality). You can follow the progress of the installation in the popup that opens thereafter - this popup will report successful installation as: "Installation successful".
5. Close the pop-up window to exit installation.



To verify that the installation was done successfully, got to the special page Special:Version - the extension should be listed on this page after the installation.

2.1.5 Uploading Local Bundles/Ontologies Menu

The **Upload local bundles/ontologies** section lets you upload content bundles directly from your local hard disk. Just click on the upload button and choose the bundle from your local file system.

Please note that only bundles in zip-format can be uploaded while ontology files can have any of the following formats : *.owl/*.rdf/*.n3/*.ttl/*.ntriple/*.nt/*.obl For more details, see section 5.4 "Importing ontologies".

The Wiki Administration Tool will only upload the bundle and not install it directly. Once you upload a bundle, it shall be seen on the list and you can install it or remove it from the list.

2.1.6 Creating Bundles

Content bundles can be exported via the WAT web tool. In the tab "Content bundles" you can select a bundle and hit 'Export'. When the bundle is created it is stored in an archive on the server from where you can download it. The list of exported bundles in the archive is shown below.

The ontology section is taken into account during a bundle export. If a page contains a bundle section for this particular bundle only content between these two markers is exported.

```
<!--BEGIN ontology: bundle-ID -->
...
<!--END ontology: bundle-ID -->
```

Bundles can contain an ontology (at most one). It's registered in deploy.xml like this:

```
<ontologies>
   <file loc="myontology.rdf"/>
</ontologies>
```

The location of the file is relative to the bundle (including install dir).

2.1.7 System Restore Menu

The **System restore** menu has all your wiki's restore points.

Restore points save the actual status of your installation and your wiki's database. We recommend that you save your installation before you install new features or edit the settings. This is especially necessary since a restore point can be used to revert and restore your installation just in case anything were to go wrong thus breaking your wiki database and making your wiki inconsistent.

2.1.7.1 Creating a Restore Point

To create a restore point, enter the name of your restore point and press the **Create** button. This opens up a popup, which informs you about the progress. It will inform you of successful restore point creation and add the restore point to the list.

2.1.7.2 Restoring the Wiki with a Restore Point

1. Select a restore point from the list and hit the **Restore** button. A popup will warn you of the effects of restoring the wiki and ask for your confirmation.

2. Hit the Yes button to restore the wiki with the selected restore point.

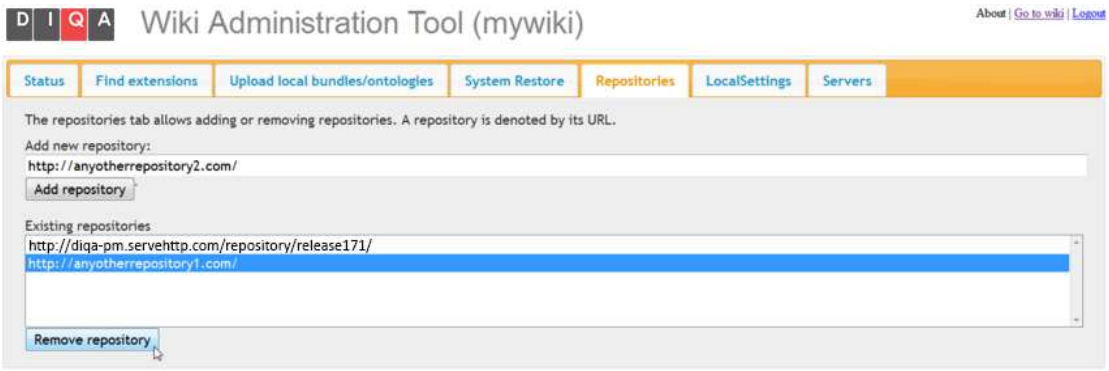
Important Note:

Restoring will REPLACE both the wiki setup and its ENTIRE contents by a previous version. YOU WILL LOOSE YOUR CURRENT WIKI CONTENT!

2.1.8 Managing Repositories

This section has a list of all registered repositories. You can easily add or remove software repositories from the list. Each DataWiki version has its own repository.
The repository for DataWiki 1.7.1 is located in:

http://diqa-pm.servehttp.com/repository/release171/



If you search for extensions from the **Find extensions** section, the Wiki Admin Tool will search through the repositories that have been defined in the **Repositories** section.

Note:

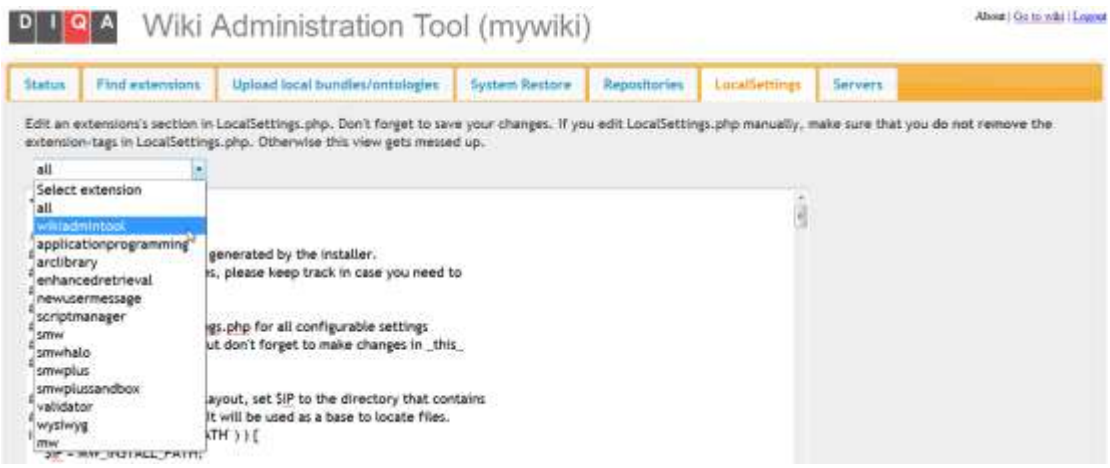
The local file of the repository file is **repositories** - it is located in: `InstallDirectory\htdocs\mediawiki\deployment\config`. If you encounter any issues, it helps to remove your repositories from this file

2.1.9 Editing LocalSettings

The LocalSettings menu enables you to edit the wiki settings file LocalSettings.php directly from the web interface. You can edit the several sections of the several extensions in the php file as follows:

1. Select extension from the dropdown menu. The text field will be activated and the code of the corresponding section in the LocalSettings.php file will be displayed.
2. Edit the section in the text field. Once you make any changes to the syntax, the **Save** button will be activated.
3. Click **Save**. The system will now overwrite LocalSettings.php.

A popup will report when the file has been saved successfully.



2.1.10 Monitor Status and Start/Stop Services

With DataWiki 1.7.1, you can now monitor the status of the servers (**Apache, mySQL, solr, tsc and memcached**) and start or stop them easily. This menu has four columns:

1. **Service** - lists these servers: Apache, mySQL, solr, tsc, memcached.
2. **Status** - lists the status of the corresponding service as follows:
   a. Green background + running means that this service is up-and running - as expected.
   b. Red background + not running means that this service down.
3. **Operation** - you can select restart, start or stop from the drop-down menu beside a service.
4. **Command** - displays the command that the selected operation does.
5. **Execute** - press this button to perform the selected operation.

Users which updated to DataWiki 1.6 or which do not use the installer can hardly use the server administration feature of WAT because they have to create the scheduled tasks used for this in Windows manually. There is a maintenance script which does this job automatically: `deployment/maintenance/installServerTasks.php`. It must be run as admin and creates the task structure needed for the WAT tool and assumes the basis is an DataWiki 1.5.6 installation (the last without this feature). If this is not the case the user has to adjust the tasks later with the Windows tool for 'Planned tasks'. He has to configure the location of the batch scripts which

start/stop the servers. These scripts can be anywhere which is the case in user-defined environments (ie. no explicit DataWiki environments).

Example

1. Run installServerTasks.php as admin
2. Open 'Planned tasks' feature of Windows.
3. Select Job: start_mysql (for example, others are ok too)
4. Right-click->Properties. Choose "Actions" and edit the first action. You'll see that all actions run the command interpreter cmd.exe and hand-over the actual start script as an argument. This script's location has to be changed to the actual location.

2.1.11 Profiler

On the top there is the button to enable/disable the profiling and to download the log as a whole. Below you find a selection box with the profiled requests. To update the request list, hit "refresh", to clear it hit "Clear log". A click on a request lists the set of function calls with details about their runtime behavior. You see the function's name, the number of calls, the total time, the time for each call and the percentage of time needed with regard to the total request time. At the bottom there is a text field for filtering for the names of the function calls.

2.1.12 Settings

The settings are:

Always overwrite user-modified pages

This settings forces WAT to *always* overwrite a wiki page with the version contained in a bundle that is installed or updated. Otherwise pages modified by the user are skipped.

Always merge pages of other bundles

Merge an already existing ontology page with the same page of another ontology. Otherwise the page is skipped.

Install optional extensions as well

Some extensions have optional "dependencies". They are rather installation suggestions than dependencies. They are *NOT* needed for proper functionality.

De-install automatically (super-)extensions which are dependent to the one which is de-installed

Be careful: This settings removes everything which depends on the extension to be removed. For example: You have SMW, Halo extension and semantic forms extension installed. If you de-install SMW then Halo extension and semantic forms extension would be also removed. Otherwise there would be an error.

Apply (partially) failed patches

Also apply patches if they could not completely applied. It is rarely the case that a patch which was not properly applied affects the whole system. In most cases only a particular feature is broken.

Always create restore point on install/de-install and update operations
Create restore points on any operation that is going to change the wiki. Will increase the required installation time.

Create hidden annotations on ontology import
Semantic data of imported ontologies can be displayed on the rendered page or not.

Use namespace prefixes for imported ontologies
Every ontology will get a namespace prefix separated with a slash, e.g. MDM/Category:Person. This is not changeable at the moment.

2.2 Command line interface of the Wiki Administration Tool

This article explains how to maintain your wiki installation with the Wiki Administration Tool by using the console. It gives an overview of the commands, in addition to the commands used for installing, de-installing or updating a package , rolling-back last changes as well as mentioning other notable points to note while maintaining your wiki with the Wiki Administration Tool.

> **Note:**
>
> This feature is reserved for **advanced users** - we recommend that you administrate the wiki with Wiki Administration Tool web interface (see previous chapter) which lets you perform administrative tasks **easily and intuitively** without having to enter the commands or work with the command prompt

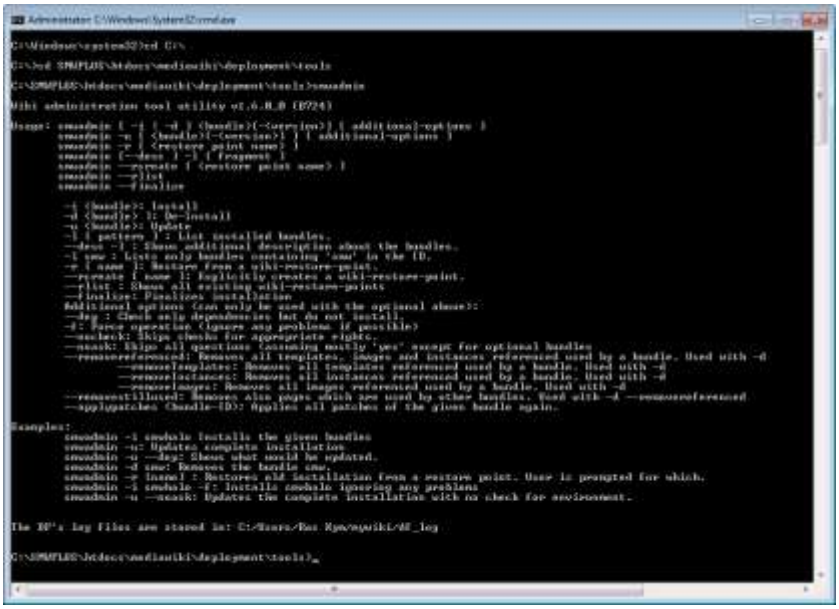
2.2.1 Prerequisites

- **Check that you have administrator privileges on the Wiki server**
  - The tool runs on the command prompt and needs administrator privileges to work or at least full read/write access to all the wiki files.
- **Check that your installation of the Wiki Administration Tool is properly configured:**
  1. Open your commandline tool and navigate to the directory:
     `<Wiki Administration Tool-install-dir>/deployment/tools/`
     `<Wiki Administration Tool-install-dir>` is the directory where the Wiki Administration Tool installed. The default location is `C:\SMWPLUS\htdocs\mediawiki`
  2. Call `smwadmin` from this location. If the call fails then check the installation status by ensuring that all aspects mentioned in the section "Installation of the Wiki Administration Tool" were followed.

- **Make sure that you run the command line as an administrator** – the Wiki Admin Tool will otherwise exit with this warning: "You have to run <command> as admin or root."

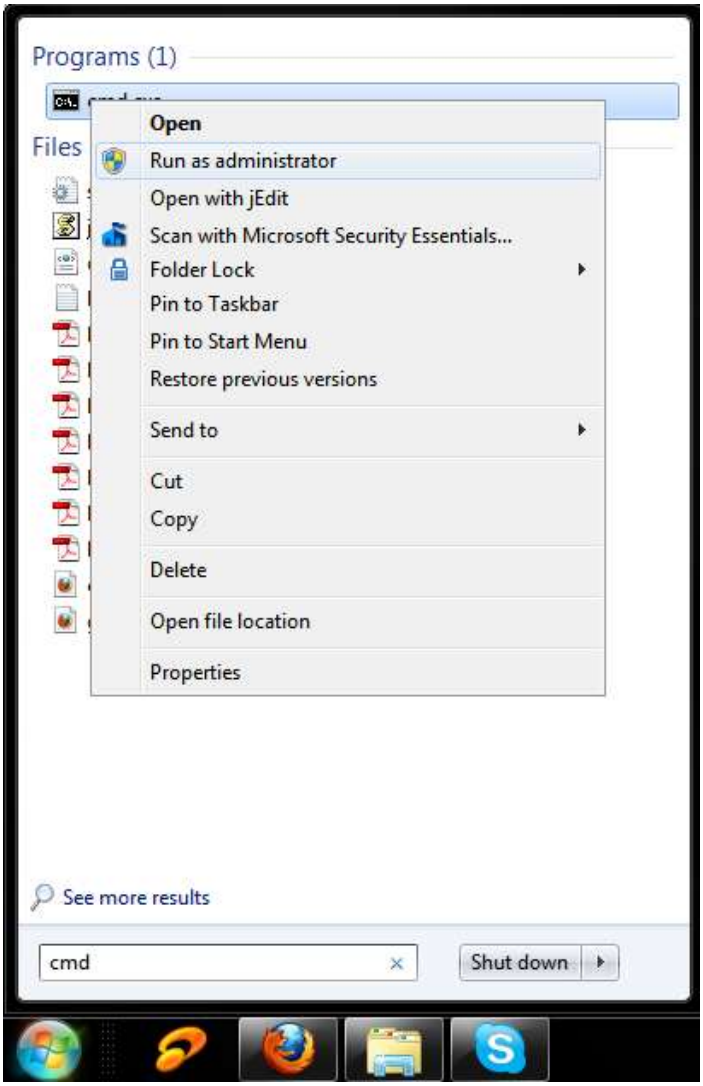
- **Installation of packages**: MediaWiki extensions can only be installed via the Wiki Administration Tool if there is no manually installed version of the extension in your wiki installation.
- **Removal of packages**: MediaWiki extensions can only be removed via Wiki Administration Tool if they were installed using it.
- **Update of packages**: MediaWiki extensions can only be updated if they were installed via the Wiki Administration Tool and were not customized in the meantime.
- The install, removal and update prerequisites apply also for required extensions. So required packages (e.g. SMWHalo requires SMW) should also be installed, updated and removed via the Wiki Administration Tool.
- Your wiki must be up and running.

The administration tool may produce unexpected results if you bypass it, e.g. by installing an extension manually, which may collide with an automatically installed extension.



2.2.2 Opening the command line as Administrator (Windows 7)

1. Click the Windows start button
2. Enter “cmd” in the search pane. The system will search and display the "cmd.exe" application
3. Right click on "cmd.exe" and select "Run as Administrator" from the context menu. This opens the "cmd.exe" application with administrator rights



4. Navigate to the "tools" directory with the following syntax (example):

```
cd C:\SMWPLUS\htdocs\mediawiki\deployment\tools
```

5. Run the desired command from this location

```
C:\SMWPLUS>cd htdocs\mediawiki\deployment\tools
C:\SMWPLUS\htdocs\mediawiki\deployment\tools>
```

2.2.3 Overview of Commands

In order to run the Wiki Administration Tool which comes with DataWiki open a command prompt and navigate to the directory `<Wiki Administration Tool-install-dir>/deployment/tools/`. Execute the following Wiki Administration Tool commands:

```
smwadmin
```

This displays a list of all available commands and options.

2.2.3.1 List installed and available packages

Lists installed and available packages. The packages available are fetched from the repository that is currently configured. Note that the DataWiki repository is pre-configured.

```
smwadmin -l
```

Workaround for the 'Access denied' error when accessing the DataWiki repository

1. Navigate to `deployment/io` and open the file `DF_HttpDownload.php`
2. Remove the following line:

```
$in .= "Host: $host\r\n";
```

2.2.3.2 Installing a package

Installation From repository

```
smwadmin -i <package>
```

This install/updates the installation, imports pages, copies resources if needed.

> **Note:**
>
> Package names are the same as listed by smwadmin -l

Examples

```
smwadmin -i smwhalo
```

Install the latest version of the Halo Extension

```
smwadmin -i treeview
```

Installs the Semantic Treeview extension. Any configuration options which might be necessary are asked interactively. In most cases you can simply accept the default values. One exception of this would be the Gardening extension which requires the location of the PHP-Interpreter.

```
smwadmin -i dataimport
```

Installs the Data Import extension.

Installation of multiple bundles at once simplifies setting up a wiki. MW and WAT are installed manually, then all other bundles follow with one command. On the command line you simply specifiy a list of bundles repeating the -i command. Example:

```
smwadmin -i smw -i semanticforms -i maps
```

Installation from local hard disk

```
smwadmin -i <path-to-the package>
```

This install/updates the bundle from local hard disk.

Example

```
smwadmin -i C:\temp\smwplus\smwplus
```

2.2.3.3 Finalizing the installation (optional)

Optional command, this finalizes the installation by executing setups, importing pages or copying resources. Normally not needed, because this happens automatically when using `smwadmin -i`

```
smwadmin --finalize
```

2.2.3.4 Checking dependencies for a package

```
smwadmin -i <package> --dep
smwadmin -u --dep
```

Example

```
smwadmin -i wysiwyg --dep
```

Checks dependencies of the WYSIWYG extension and shows the required packages.

2.2.3.5 De-installing a package

```
smwadmin -d <package>
```

Example

```
smwadmin -d smw
```

Deinstalls Semantic MediaWIki

2.2.3.6 Updating a package

```
smwadmin -u <package>
```

Example

```
smwadmin -u
```

Updates complete installation, i.e. the installed packages (MW extensions and content bundles) which can be updated without conflicts. Note that MW itself will not be updated.

```
smwadmin -u --dep
```

Shows what would be updated.

2.2.3.7 Restore points

Restore points save the actual status of your installation and your wiki's database. You can create a restore point at any time. If you want to restore your installation to an older status of your installation (database), you may as well easily load one of your created restore point.

```
smwadmin --rcreate <name>
```

Creates a restore point with the given name.

```
smwadmin --rlist
```

Shows a list of all restore points.

```
smwadmin -r <name>
```

Restores wiki to a specific restore point.

```
smwadmin -r
```

Restores wiki to the last restore point.

2.2.3.8 Log

The Wiki Administration Tool writes a log about the operations. It is stored in your system's temporary folder:

On Windows

DataWiki 1.7x stores its log file in this location:

```
C:\Users\<John Doe>\mywiki\df_log
```

where <John Doe> is the current user. To get the location of the log file, run the command `smwadmin` – the last line shows the location of the log file.

On Linux

DataWiki 1.7x stores its log file in this location:

```
/tmp/df_log
```

There are usually two log files for each run of smwadmin. One for the installation process and one for the initialization process.

3 Maintenance

⚠ Important Note:
As of this version, this section might contain incomplete or wrong information. We plan to review and update this section with the next revision of the document.

3.1 System Information and Statistics

This section explains how to handle the statistics and how to get information about the wiki.

3.1.1 Special page Version

The special page 'Version' provides information about DataWiki's configuration, license information and all extensions which has been installed. This version information is helpful when diagnosing errors you may face using DataWiki. If you enter a bug report or a inquiry about how to use single features of the software, the more detail you can provide about your installation and environment the faster we will be able to help.

3.1.2 Special page Statistics

The special page 'Statistics' shows the total number of pages and users in various categories.

3.1.3 Special page SemanticStatistics

This page gives an overview of all property values, properties, intended datatypes the wiki contains, unused properties, properties that lack a page and a list of wanted properties.

3.2 Backing up the Database

DataWiki stores its content in a MySQL database. To avoid accidental data loss the database should be updated on a regular basis. This can be achieved with the MySQL Administrator Software. Follow the step by step instructions to set up the backup system and learn how to restore a snapshot.

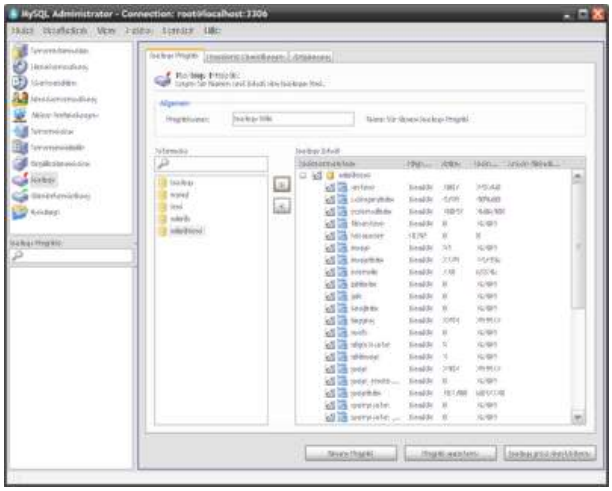
3.2.1 Scheduling Database Backups

Installing the Backup System

1. Download the MySQL GUI Tools from MySQL.org (http://dev.mysql.com/downloads/gui-tools/5.0.html). For windows choose the installer version (Windows (x86)).
2. Start the installation and accept the license agreement. Choose an installation directory. For backup purposes we only need the MySQL Administrator but the bundle contains additional GUI tools. So we choose the custom installation.
3. In the Custom setup page of the wizard disable all features but the Common components and the MySQL Administrator. Finish the installation.

Setup the Backup Scheduler

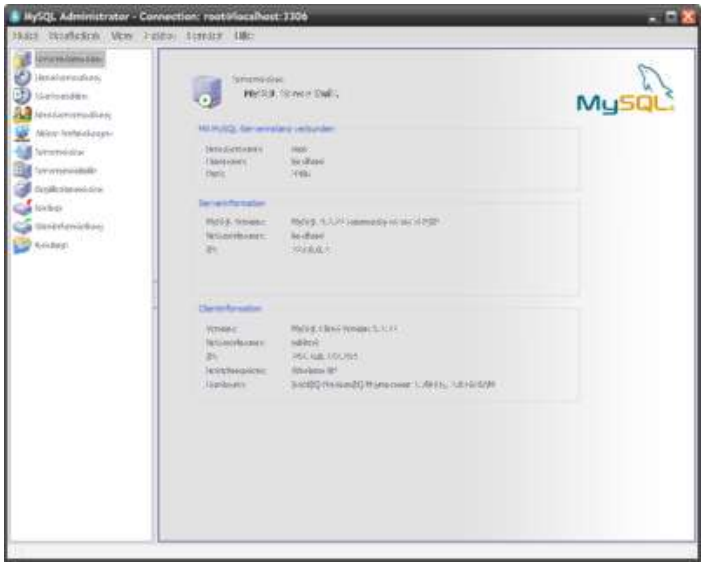
1. Start the MySQL Administrator on the computer that runs the wiki. A connection to the database is requested. Some values are already suggested.



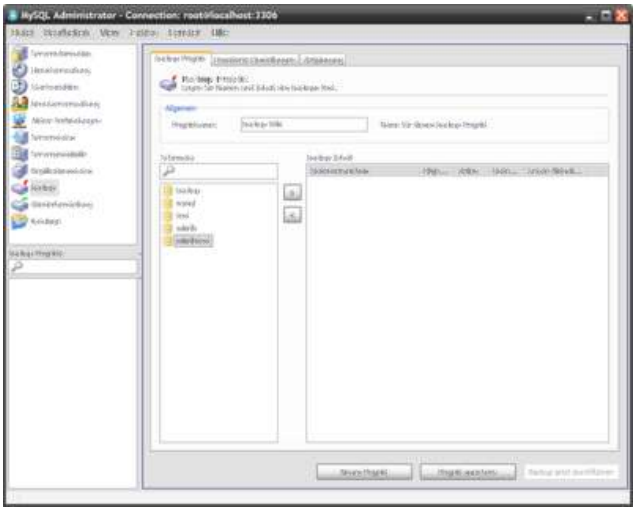
If you do not know the database settings of your wiki, you can look them up in LocalSettings.php in the wiki's installation directory. Locate the following variables:

```
$wgDBserver          = "localhost";
$wgDBname            = "wikidbtest";
$wgDBuser            = "root";
$wgDBpassword        = "agds8z";
```

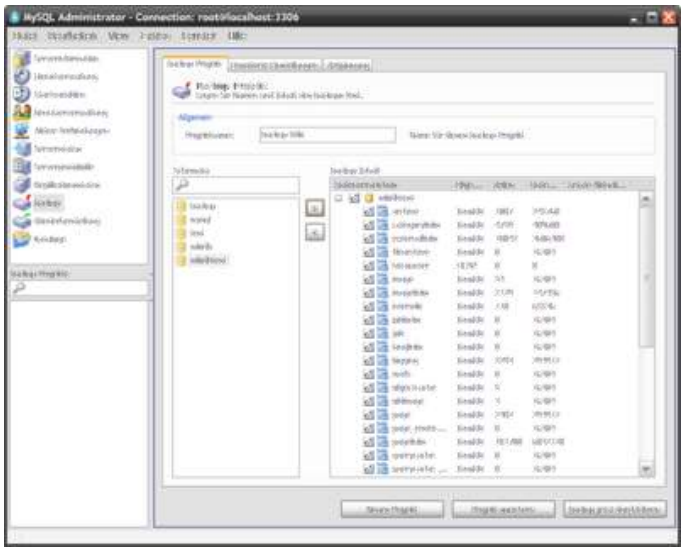
If the MySQL Administrator can successfully connect to the database server, the following screen should appear:



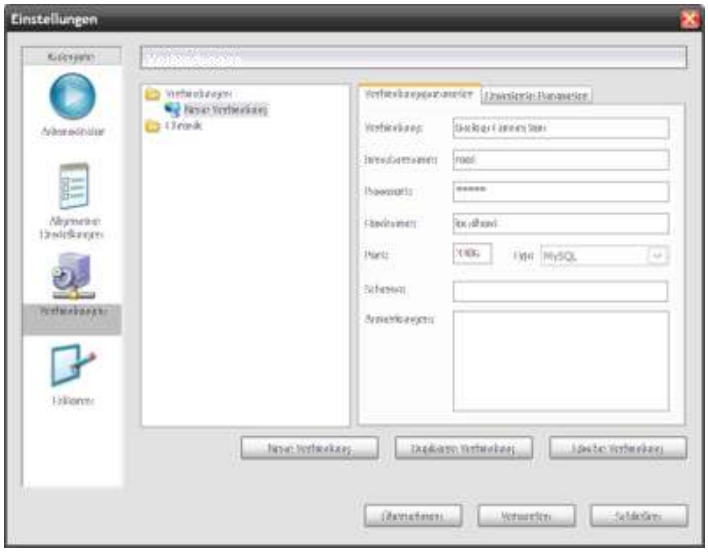
2. Select the Backup action in the left part of the window. The right part changes and shows the properties of a backup project. First, a new project has to be created ("New Project"). Enter a name for the new project e.g. Backup Wiki.



3. There is a list of all existing databases. Select your wiki database (see the variable $wgDBname in LocalSettings.php) and move it to the list of databases that should be backed up (Button: ">"). All tables of the added database will be shown.



4. The backup should be executed regularly. Switch to the tab "Time Planning" and configure the settings that satisfy your needs.
5. A scheduled backup needs a stored connection. MySQL Administrator suggests to open the "Connections" dialog when you set the checkmark for "Plan this Backup Project".

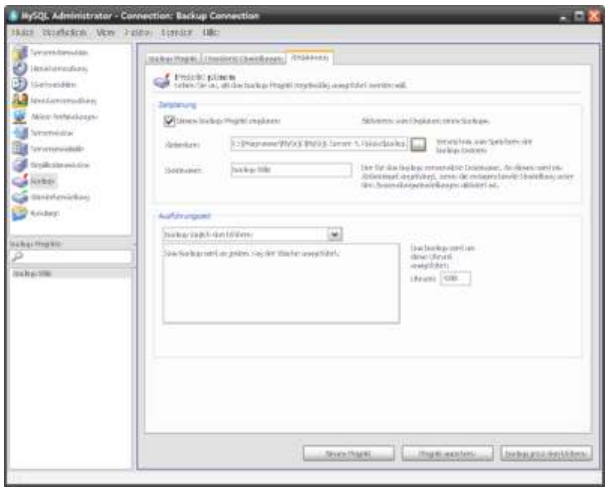


6. Enter your settings, confirm them and close the dialog.
7. The system requests that passwords must be hidden.

8. Open the settings dialog and checkmark “Save Passwords” with method “Obscured”.
9. Save your backup project. It will appear in a list in the lower left of the window.
10. From the menu at the top of the window select File > New connection. In the connection dialog, choose the new connection.

11. Select Backup. The backup settings will open again. Choose your backup project from the list of available backup projects.

12. Choose the tab "Time Planning" again, enable "Plan this Backup Project" and choose the schedule to your liking.

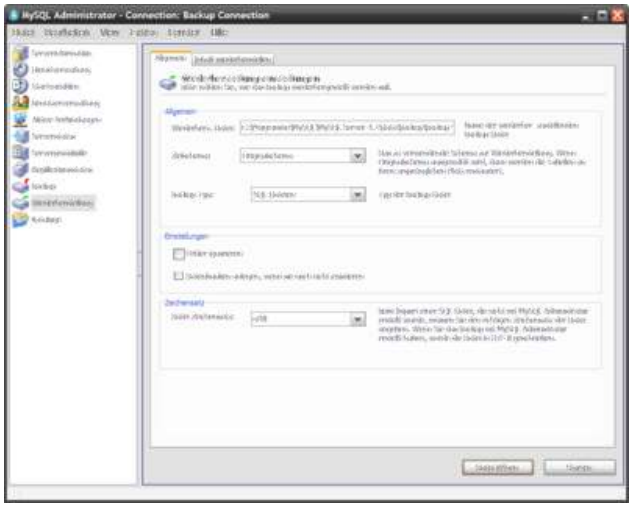


13. Save your project. The system will ask for the user account under which the backup will be started.

14. You are finished. Close the MySQL Administrator.

3.2.2 Restoring A Backup

1. Stop the Apache web server. This prevents accesses to the wiki database.
2. Start MySQL Administrator and select your backup connection.
3. Click "Restore". Click "Open File" in the right part of the window and select your backup file.



4. Click "Start" to restore the database. This will take several minutes.

5. Restart the Apache web server.

3.3 Repairing Data

Information about repairing data in DataWiki can be found here:
http://semantic-mediawiki.org/wiki/Help:Repairing_SMW's_data

4 Advanced Configuration for DataWiki

Important Note:
As of this version, this section might contain incomplete or wrong information. We plan to review and update this section with the next revision of the document.

DataWiki offers some additional optional features which have a far reaching effect in improving the effectiveness of your wiki once they are enabled. These advanced configuration options are available for the auto-completion, queries, deployment, database configuration, and close match search features.

DataWiki has several features that are optional or allow advanced configuration. The settings are made in the file `LocalSettings.php`, which is located by default in `C:\SMWPLUS\htdocs\mediawiki`, unless otherwise stated. Set respective parameters somewhere below the line

```
include_once('extensions/SMWHalo/includes/SMW_Initialize.php');
```

in your LocalSetting.php.

Example

```
include_once('extensions/SMWHalo/includes/SMW_Initialize.php');
$smwgHaloPropertyPageFromTSC = true;
```

4.1 Enabling Semantic Autocompletion

Semantic autocompletion also considers schema information in contrast to lexical AC. Be aware that semantic autocompletion causes higher server load.

```
$semanticAC = true;
```

Type: optional
Default: false

4.2 Activating the Factbox

The factbox is activated by default in DataWiki. It might be necessary to activate the factbox after a manual installation, therefore add the following variable:

```
$smwgShowFactbox = SMW_FACTBOX_NONEMPTY;
```

Type: optional
Default: n. a.

4.3 Enabling Queries

Enables inline queries.

```
$smwgIQEnabled = true;
```

Type: required
Default: true

4.4 Protecting the Wiki with a HTTP password

If you protect your wiki with a htaccess file and use the Auth Digest method then you need to set the http password in the variable $smwgHaloHttpAuthPassword. The username will be taken from the http request, and is not set here.

```
$smwgHaloHttpAuthPassword='your password';
```

Type: optional

Default: undefined

4.5 Activating 'Create new page' Dialogue when Following Red Links

You can allow the wiki to modify the red links so that the user is sent to a custom wiki page. This page (called Create_new_page) contains a dialogue where one can select on how one wants to add new content to the wiki. More information

> **Note:**
>
> This feature will not be used when `$smwgHaloNEPEnabled` is set in `LocalSettings`.

```
$smwgHaloRedLinkWithCreateNewPage = false;
```

Type: optional
Default: false

4.6 Changing the Hostname when Query Interface can't be opened in WYSIWYG Editor

Use this option when the Query Interface cannot be opened in the WYSIWYG editor. This can happen when the URL to access the wiki from outside is different as when accessing the wiki from the server itself. The latter is done to get the page content of the Query Interface when it's opened from within the wiki itself (e.g. the WYSIWYG editor or the Semantic Toolbar).

```
$smwgHaloQueryInterfaceHost4Wysiwyg='hostname';
```

Type: optional
Default: undefined

4.7 Restricting Access for Anonymous Users for Query Interface

This option is needed when you block your wiki for anonymous reading access. When opening the Query Interface inside the WYSIWYG extension the access is done without sending the authentication token, therefore the access must be checked by creating authentication tokens by a random string and this secret, so that the Query Interface cannot be accessed as an anonymous user.

```
$smwgHaloQueryInterfaceSecret='hash';
```

Type: optional
Default: n. a.

4.8 Renaming Categories in the Data Explorer

By default Mediawiki doesn't allow the move operation on categories. When a category is renamed, this is a move operation internally. Therefore if you want to rename categories in the Data Explorer a Mediawiki file must be modified to achieve this. Please edit the file `includes/Namespace.php` in your wiki installation. Replace the line:

```
return !( $index < NS_MAIN || ($index == NS_FILE &&
!$wgAllowImageMoving)  || $index == NS_CATEGORY );
```

with:

```
return !( $index < NS_MAIN || ($index == NS_FILE &&
!$wgAllowImageMoving) );
```

The location of this content is around line 50ff.

4.9 Enabling uploads

Before users are allowed to upload files to the Wiki system, you have to enable that feature. Make sure the Upload Directory is properly configured and writeable by the Apache web server process. Then set the $wgEnableUploads variable to true to allow uploading in the web user interface.

```
$wgEnableUploads = true;
```

Type: optional
Default: true

4.10 Extending the list of accepted file extensions

You may want to modify the list of accepted extensions, which is stored within the $wgFileExtensions array. Uploading files with extensions not in this list will trigger a warning.

```
$wgFileExtensions = array( "png", "jpg", "jpeg", "ogg", "doc",
"xls", "ppt", "mp3", "pdf");
```

Type: optional
Default: "svg"

4.11 Setting maximum file size for file upload

By default, the configuration code in php.ini limits the size of files to be uploaded to 2 megabytes (and the maximum size of a post operation to 8 megabytes). To allow uploading of larger files, edit these parameters in **`php.ini`**:

```
post_max_size,[1]
upload_max_filesize,[2]
```

Replace [1] and [2] with the required file sizes. Defaults are 8 MB and 2 MB respectively.

4.12 Enabling OntoSkin for All Users

Enables OntoSkin and all DataWiki features like the semantic toolbar to all users including those who are not logged in.

```
$wgDefaultSkin = 'ontoskin3';
```

Type: optional
Default: 'ontoskin3'

4.13 Changing the Logo

The `$wgLogo` variable specifies which graphical logo is displayed in the top left corner of all wiki pages.

```
$wgLogo = "$wgScriptPath/MyLogo.jpg";
$wgScriptPath="/mediawiki";
```

This replaces the default logo in the /mediawiki directory.

Type: optional
Default: 'wiki.jpg'

4.14 Changing Language

The `$wgLanguageCode` variable controls the language of your wiki's interface. While users can switch the language they see in their preferences, this variable sets the default language that all anonymous users and most registered users see. DataWiki supports DE, EN, FR and NL, last two are not complete.

```
$wgLanguageCode = "en";
```

Type: optional
Default: "en"

4.15 Changing Site Name

The $wgSitename variable holds the name of your wiki setup. This name gets included many times throughout the system.

```
$wgSitename = "DataWiki";
```

Type: optional
Default: "MyWiki"

4.16 Configuring the menus of the DataWiki user interface

The DataWiki user interface provides a skin named "OntoSkin3" that is activated by default.

4.16.1 Basic Functionality

OntoSkin3 is flexible. The design and customization of the menu is simple. Its functionality is quite similar to the MediaWiki:Sidebar – known from the older skin MonoBook. The navigation menu in OntoSkin3 is horizontal and is located in the upper section of the page. This makes content browsing easy.

You can build the several menu items in wiki pages with normal page code, and the content of the wiki page will be transcluded into the drop-down menu.

4.16.2 How to create a simple navigation menu

If no navigation menu has been defined in your wiki, you will get this warning in the navigation menu ribbon:

> "no menu defined, see smwplus.net for details"

The navigation menu is defined from the page *MediaWiki:Halomenu*. If your wiki address is `http://<your wiki address>/mediawiki/index.php`, then the menu can be modified from the page
`http://<your wiki address>/mediawiki/index.php/MediaWiki:Halomenu`
In order to illustrate the functionality, a simple navigation menu will be created first. It consists of 4 toplevel items: "Categories", "Community", "Help" and "Get started with DataWiki!". The structure is set in the article MediaWiki:Halomenu.

1. Go to your Wiki's MediaWiki:Halomenu and open it in the wikitext edit mode.

> ⚠ **Important Note:**
>
> We recommend the use of Wikitext edit mode (Edit (wikitext editor)! Avoid using WYSIWYG-mode to create menus since the html-code might not always be rendered correctly. You can open an article's wiki-text editor by appending the following string to the URL of the article: `?action=edit`

2. Enter the following code:

```
* Categories
* Community
* Help
* Get started with DataWiki!
```

3. Save the page. Your menu should look like this:

4. Now edit the page again and add to each top-level the page which you want to transclude on each menu item:

```
* Categories
** MediaWiki:Categories
* Community
** MediaWiki:Community
* Help
** MediaWiki:Help
* Get started with DataWiki!
** MediaWiki:Smwplussandbox
```

5. Save the page. Move the cursor over the menu items – you will notice a small bubble under each item.

6. The following pages need to be created:
   a. MediaWiki:Categories,
   b. MediaWiki:Community
   c. MediaWiki:Help

7. Create the page MediaWiki:Categories:
   Open the page MediaWiki:Categories and add following code:

```
<div style="width:200px; height:180px">
<table   border="0"   style="empty-cells:show;   margin-top:10px;   height:auto;
width:200px;">
<tr> <td style="vertical-align: middle;" colspan="2">
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div   class="table-cell">[[Culture   and   the   arts|Culture   and   the
arts]]</div>
    </div>
  </div>
</div>
</td>
</tr>
<tr>
<td style="vertical-align:middle;" colspan="2">
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div    class="table-cell">[[Geography    and    places|Geography    and
places]]</div>
    </div>
  </div>
</div>
</td>
</tr>
<tr>
<td style="vertical-align:middle;" colspan="2">
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div class="table-cell">[[History and events|History and events]]</div>
    </div>
  </div>
</div>
</td>
</tr>
<tr>
<td style="vertical-align:middle;" colspan="2">
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div class="table-cell">[[Society and social sciences|Society and social
sciences]]</div>
    </div>
  </div>
</div>
</td>
</tr>
<tr>
<td style="vertical-align:middle;" colspan="2">
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div class="table-cell">[[Technology and applied sciences|Technology and
applied sciences]]</div>
    </div>
  </div>
</div>
</td>
</tr>
</table>
</div>
```

8. Save the page and move the cursor over the menu items and you will see a drop-down menu:

9. Repeat step 6 to create the other pages which will be the content of your menu. Adapt the links accordingly as you like them.

Note:

A table was used in this example to generate the drop-down menu, but you may also use lists instead. You may also enrich your menu with images and css-styles.

4.16.3 How to create a navigation menu with icons

If you wish to have icons in front of every menu item, use the class table-cell-icon

1. Open the page MediaWiki:Smwplussandbox
2. Insert this code:

```
<table border="0" style="empty-cells:show; margin-top:10px; height:auto;
width:335px;">
<tr>
<td>
<div class="table-wrapper">
  <div class="table">
    <div class="table-row" style="border-bottom:1px solid #EBEBEB;">
      <div class="table-cell icon"
style="width:30px;">[[File:sandbox.png|link=Smwplussandbox]]</div>
      <div class="table-cell">[[Smwplussandbox|DataWiki Sandbox]]</div>
    </div>
  </div>
</div>
</td> </tr>
<tr> <td>
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div class="table-cell icon" style="width:30px;">[[File:Arrow-
round.png|link=Smwplussandbox/Basic Wiki-maintenance]]</div>
      <div class="table-cell">[[Smwplussandbox/Basic Wiki-maintenance|Lesson 1:
What you should do after the installation]]</div>
    </div>
  </div>
</div>
</td> </tr>
</table>
</div>
```

3. Save the article - the menu should now have icons. Ensure that all files and icons used are uploaded to your wiki

4.16.4 How to create a mega menu with icons and subtitles

1. Go to the page `MediaWiki:Halomenu`
2. Extend the menu with this code

```
* Articles and data
** MediaWiki:Articlesanddata
```

3. Now edit the page `MediaWiki:Articlesanddata` in wikitext edit mode and add this code:

```
<div style="width:525px; height:220px">
<div style="float: left; width:250px;">

<table   border="0"   style="empty-cells:show;   margin-top:10px;   height:auto;
width:250px;">
<tr>
<td>

<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div                    class="table-cell                    icon-double"
style="width:40px;">[[File:Icon_DataExplorer.png|link=Special:DataExplorer]]</d
iv>
      <div class="table-cell">[[Special:DataExplorer|Data Explorer]]</div>
    </div>
    <div class="table-row">
      <div class="table-cell" style="width:40px;"></div>
      <div  class="table-cell  about">[[Special:DataExplorer|Browse  and  edit  the
category tree,  instances and properties.]]</div>
    </div>
  </div>
</div>
</td> </tr>

<tr> <td>
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div                    class="table-cell                    icon-double"
style="width:40px;">[[File:Icon_QueryInterface_32x32.png|link=Special:QueryInte
rface]]</div>
      <div class="table-cell">[[Special:QueryInterface|Query Interface]]</div>
    </div>
    <div class="table-row">
      <div class="table-cell" style="width:40px;"></div>
      <div  class="tabel-cell  about">[[Special:QueryInterface|Build  and  execute
queries for data and visualise them.]]</div>
    </div>
  </div>
</div>
</td> </tr>

<tr> <td>
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div                    class="table-cell                    icon-double"
style="width:40px;">[[File:ContentBundles.png|link=All_content_bundles]]</div>
      <div class="table-cell">[[All_content_bundles|Content Bundles]]</div>
    </div>
    <div class="table-row">
      <div class="table-cell" style="width:40px;"></div>
      <div  class="table-cell  about">[[All_content_bundles|Get  the  list  of
ontologies or content bundles in the wiki.]]</div>
    </div>
  </div>
</div>
</td> </tr>
</table>
</div>
<div  style="float:  left;"><table  border="0"  style="empty-cells:show;  margin-
left:20px; margin-top:10px; height:auto; width:250px;">

<tr> <td>
<div class="table-wrapper">
  <div class="table">
    <div class="table-row">
      <div                    class="table-cell                    icon-double"
style="width:40px;">[[File:Icon_WikiActivity_32x32.png|link=WikiActivity]]</div
>
      <div class="table-cell">[[WikiActivity|Wiki Activity]]</div>
    </div>
    <div class="table-row">
      <div class="table-cell" style="width:40px;"></div>
      <div  class="table-cell  about">[[WikiActivity|Monitor  recent  changes,  new
users, new articles and new files]]</div>
    </div>
  </div>
</div>
</td> </tr>

<tr> <td style="vertical-align: middle;" >
<div class="table-wrapper">
```

4. Save the article and the menu should now look like the screenshot - ensure that all the files used are uploaded to your wiki.

4.16.5 Notes on the syntax

The syntax in the page `MediaWiki:Halomenu` contains the following code:

```
* Dashboard
** MediaWiki:Mydashboard
* Articles and data
** MediaWiki:Articlesanddata
* Get started with DataWiki!
** MediaWiki:Smwplussandbox
```

- Single asterisks ("*") are used as menu headlines shown directly in the OntoSkin3 menu bar.
- The lines beginning with two asterisks ("**") point to the given pages. These will be rendered like normal wiki pages and shown under the above header as menu content when hovering over it.
- You need to have a space between the * or ** and the actual menu entry.

From the example above it follows that the article `MediaWiki:Mydashboard` contains the content that will be rendered in the menu item `Dashboard`, and so on. The advantage of this approach is that you have more possibilities to create menus: you can use tables, images and css-styles in the pages.

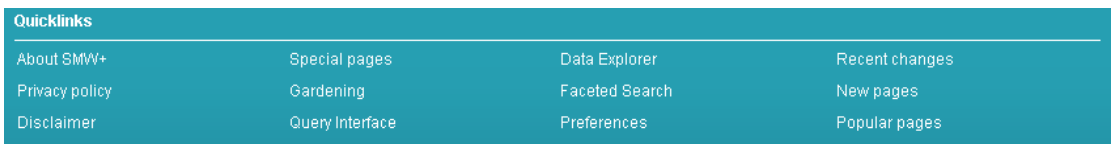
4.16.6 Other Menus

The following table summarizes all pages for customization of DataWiki

Page	Description
MediaWiki:Halomenu	Add horizontal navigation menu
MediaWiki:Halomenuconfig	Add the normal MediaWiki menu to skin
MediaWiki:Halopageoptions	Add items to 'More"-menu
MediaWiki:Haloquicklinks	Add content to footer

4.16.6.1 Customize the footer

You can define the content of the footer (the quicklinks section) at the page `MediaWiki:Haloquicklinks`.



4.16.6.2 The More menu

The more menu in DataWiki 1.6.0 is located in the upper left section of a page. It can be modified from the page `MediaWiki:Halopageoptions`

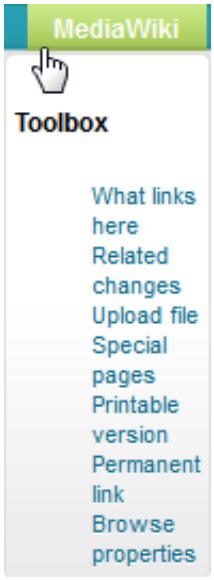
4.16.6.3 How to use the Classic MediaWiki menu

To use the classic MediaWiki menus with OntoSkin3:

1. Go to the page MediaWiki:Halomenuconfig
2. Add this code to the page:

```
showmediawikimenu=true
```

3. Save the page - the menu should now use the classic MediaWiki menu



5 Optimizing Response Times

Important Note:
As of this version, this section might contain incomplete or wrong information. We plan to review and update this section with the next revision of the document.

DataWiki is shipped with a configuration that is optimal with regard to response times. Nevertheless, it is advisable to regularly verify that the actual response times of an DataWiki installation correspond to the expected response times and to maintain the installation if a degradation becomes obvious

5.1 Usual Response Times

Here is a table of response time you can expect for this a default DataWiki configuration:

Action	Response Time (sec)
View a page	2.0 - 3.0
View a page with complex queries (with or without TSC)	3.5 - 4.5
Open an article in wikitext edit mode	2.0 - 3.5
Save an article in wikitext edit mode	3.0 - 5.0
Open an article in WYSIWYG edit mode	3.0 - 5.0
Save an article in WYSIWYG edit mode	3.0 - 5.0
Open an article in Advanced Annotation mode	2.5 - 3.5
Save annotations in Advanced Annotation mode	1.0 - 2.0

If your installation provides these response times then your installation is optimal.

5.2 How to Measure Response Times

You need an external tool to measure the responsetimes of your installation, we recommend to user one of these tools:

1. Developer Tools that are contained in the Chrome Webbrowser or
2. Net Panel of Firebug (a debugging tool for the Firefox webbrowser).

In the following we explain how to install the net panel tool.

5.2.1 Install Firebug for Firefox

You can measure the access times to a web page in Firefox with the plugin Firebug (https://addons.mozilla.org/de/firefox/addon/firebug/). It contains a net panel in the tab Net that gives detailed information about the load times of a web page and all of its pieces. Please read

- The Introduction to the Net Panel (http://www.softwareishard.com/blog/firebug/introduction-to-firebug-net-panel/) and
- The Net Panel Timings (http://www.softwareishard.com/blog/firebug/firebug-net-panel-timings/)

With this tool and knowing how to read its diagrams you are well equipped to measure the performance of your wiki.

Besides measuring the performance manually you can also automate this task, for example with Apache JMeter (http://jakarta.apache.org/jmeter/). However, this is far beyond the scope of this article.

5.2.2 Perform a Sequence of Tests

The wiki can do several things for you. You can read articles, edit and save them, log in and out, work interactively with toolbars that use ajax calls, etc.

Besides the detailed waterfall graph, the Net Panel shows the total time for loading a page at the bottom. When something like "5.13s (onload: 3.45s)" is reported, the first given time is what is interesting for the user. After this time the page is really complete and the user can work with it.

In order to compare your wiki with the typical response times given above, you must measure the following actions in your wiki:

1. View the Main Page
2. View a page with complex queries (with or without TSC)
3. Open the Main Page in wikitext edit mode
4. Save the Main Page in wikitext edit mode
5. Open an article in WYSIWYG edit mode
6. Save an article in WYSIWYG edit mode
7. Open an article in Advanced Annotation mode
8. Save annotations in Advanced Annotation mode

You can repeat these tests with different extensions enabled. Be aware that each extension will add an extra penalty to the response times. So keep your wiki as lean as possible.

You should measure each action you are interested in several times (e.g. 5 times) to get a meaningful average of access times. Normally the first time an action is requested the response is the slowest as the browser caches are filled. From then on the action should be much faster.

5.3 Improve the response times of your installation

In the following we give general advise before explaining typical symptoms and how to remedy them.

5.3.1 Use the Latest Browser Generation

DataWiki is optimized for the following browsers:

- Firefox 4.0.x
- Internet Explorer 8
- Firefox 3.6.x (very slow!)

The following browsers are not officially approved from us, but you can use them though:

- Firefox 6+
- Google Chrom 13+

Old browsers are usually very slow in processing complex script files. The latest browser generation is highly optimized for scripts and speeds them up considerably.

5.3.2 Are there Needless Extensions?

Every extension that is added to the system has a performance footprint. The more extensions are installed the slower they system will react. Get rid of extensions that you do not use.

5.3.3 Are all Required Wiki Services Running?

You can configure that DataWiki uses a Triple Store or the SOLR server. If this is the case you must make sure that these services are really running. The special page continues trying to connect to SOLR and show an appropriate message while doing so.
Please check for the presence of these services:

1. SOLR
2. memcached
3. TripleStore

5.3.4 Troubleshooting

Query processing is slow

The wiki accesses the TSC through a web service which is normally configured like this:

```
$smwgWebserviceEndpoint='localhost:8092';
```

But sometimes resolving the IP-address of 'localhost' takes a long time as the system may at first try to find the IP-v6 address and if it fails the IP-v4 address. Specifying the IP-address may solve this:

```
$smwgWebserviceEndpoint='127.0.0.1:8092';
```

Under Windows it is also possible to define the IP address in the 'hosts' file (typically located here: `c:\Windows\System32\drivers\etc\hosts`):

```
# localhost name resolution is handled within DNS itself.
127.0.0.1       localhost # (IP-v4)
#::1             localhost # (IP-v6)
```

DNS lookup problem

Sometimes the browser has problems resolving to domain name of the wiki site to an IP-address. Watch out for the DNS lookup times in Firebug. If problems occur you should use the IP-address of the site or preferably contact your system administrator.

Search engine ("SOLR server") not running or wrong IP adresse configured

- **Symptom**: saving an article takes longer than 20 seconds
- **Cause**: When saving articles, the fulltext index of the SOLR search engine, that is included in DataWiki, is updated. If the SOLR search engine is not started then a timeout occurs.
- **Remedy**: Start the SOLR search engine in your installation of DataWiki. You can verify that SOLR is property connected to DataWiki if you can successfully search for an article in it. Adjust the IP-adress of your server in the SOLR configuration file (pls. refer to the administration manual of DataWiki).
-

memcached-tool not running

- **Symptom**: saving an article takes longer than 10 seconds
- **Cause:** the memcached-tool that is delivered with DataWiki is properly working.
- **Remedy:** In Windows: stop and restart the memchached-service a couple of times.

5.4 Orchestrating the DataWiki Caches

Lots of information about accelerating wikis and web servers are available in the web. You should at least read the first article of the following list before reading on:

- Caching in MediaWiki:
  - Basics for caching in MediaWiki: http://www.mediawiki.org/wiki/Manual:Cache
  - Configuring MediaWiki's caches: http://www.mediawiki.org/wiki/Manual:Configuration_settings#Cache
- Apache:
  - Caching: http://httpd.apache.org/docs/2.2/caching.html http://httpd.apache.org/docs/2.2/mod/mod_cache.html
  - Compression of files: http://httpd.apache.org/docs/2.0/mod/mod_deflate.html
- Google caching recommendations: http://code.google.com/intl/de-DE/speed/page-speed/docs/caching.html
- Background information about HTTP caching in general: http://www.mnot.net/cache_docs/

Several caches accelerate the wiki in the following order according to the flow of the request for an article:

- Squid
- Apache cache
- MediaWiki's parser cache
- Query result cache

We will explain these caches in the following sections.

5.4.1 Squid

Squid is a high-performance proxy server that can also be used as a HTTP accelerator for the web server. Squid will store a copy of the pages served by web server and the next time the same page is requested, Squid will serve the copy.

Squid is only effective when articles are requested by anonymous users. At least, this holds for the HTML part of an article. However, requests for articles usually consist of several files like stylesheets (css), scripts (js) and images. These can be provided by the squid as well (for logged-in users as well as anonymous users).

If Squid can serve articles the performance is drastically improved. In SMWforum the request for the article's body can be answered in under 100ms instead of a few seconds.

See http://www.mediawiki.org/wiki/Manual:Squid_caching for installation instructions.

5.4.2 Apache Cache

When a request can not be answered by Squid it is sent to Apache. Static files can be cached by Apache if the cache is turned on. See the section "Apache Configuration" below for more details.

5.4.3 Parser Cache

The next cache in the chain is Mediawiki's parser cache. It stores the complete HTML that the parser has generated from the wiki text of an article and thus can speed up page access considerably. On complex pages this can save several seconds.

The parser cache has to be activated in `LocalSettings.php`. See http://www.mediawiki.org/wiki/Manual:Configuration_settings#Parser_Cache

However, in SMWforum the parser cache is disabled by several extensions. It has to be verified if this is really needed in the following cases, as the cache speeds up complex articles considerably.

- /extensions/HashingFunctions/HashingFunctions.php
  - lines 50 and 54
- /extensions/ApplicationProgramming/URLArguments/URLArguments.php
  - line 46
  - Question: Is the parser function actually used in an article or can this extension be disabled?
- /extensions/HTMLets/HTMLets.php
  - line 67
  - needed for the banner on the main page but seems to work also if option nocache="yes" is not set.
- /extensions/NoTitle/notitle.php
  - line 50

We recommend to find all extensions that disable the parser cache and evaluate if they do this for a valid reason.

5.4.4 Query Result Cache

As the name indicates, the QRC caches the results of a query. The idea behind this is that queries can take a long time if they retrieve lots of data or if they are complex. In this case the QRC simply returns a previously calculated result which is then processed as usual by the query result printers. Queries that are processed quickly are barely accelerated by the QRC.

5.4.4.1 When is the Query Result Cache not Used?

The QRC is not used when the content of a page is recalculated. This happens when it is

- edited and saved, (To be precise, this depends on the circumstances under which the article is saved. If the parser cache is disabled or invalidated, an article is parsed twice when it is saved. During the first parse cycle, the QRC is not used, because users should get the most current query results if they edit an article. During the second parse cycle, the QRC is used, if the parser cache does not provide valid entries.)
- purged (e.g. with ?action=purge),
- returned from the parser cache,
- returned by Squid.

5.4.4.2 When is the Query Cache Used and Effective?

The QRC is used if all of the following conditions are met:

- There is no Squid, or Squid is bypassed because the user is logged in.
- The parser cache is empty or it is disabled in general or depending on the content of an article. In the latter case extensions can disable the parser cache temporarily. (However instead of relying on the QRC you should investigate which extensions disable the parser cache and if this is really mandatory. The parser cache accelerates the wiki more than the QRC.)
- A page contains queries and these have been processed at least once.
- Processing of the queries by the Triple Store or SMW store takes a long time. (Otherwise the effect of the QRC is hardly noticeable.)

Example

We profiled an article contains 42 queries. 11 of them are answered by the triple store and 31 by the SMW store. The triple store needed 1.8 seconds and SMW 0.5 seconds when the QRC was empty. The second time the page was opened, the QRC was filled and all queries were answered in 0.5 seconds.

5.4.4.3 Other Benefits of the QRC

Even if the QRC can not accelerate the wiki because of the restrictions explained above, there is another great benefit. It can keep all wiki articles that contain queries up to date, even if they are cached.

Normally the caching strategy fails if the result of a query changes as the cache is ignorant of this. Let's assume that article A contains a query for facts that are defined in article B. If these facts are modified, the cache of B will be purged as the system knows that B was modified by an edit operation. However, article A will be delivered in its outdated form as the cache does not know the dependency to B.

The QRC knows which properties affect the query result in an article and it can invalidate the page's cache when such a property is changed. In this case A will be updated after the facts in B were changed.

5.4.5 Browser Cache

One of the most important caches is that of the user's browser. The body of a wiki article is never cached but all other files like stylesheets, scripts and

images. The browser cache can only be applied if the HTTP headers of these files are set correctly by the server (see section Web server settings). Furthermore the user should be aware of how to control the cache of his browser.

Load a page
If the user clicks a link to a wiki article, only the corresponding HTML files are loaded from the server. All other auxiliary files are fetched from the browser cache (if they are stored there). This is the fastest way of loading articles with only few requests.

Refresh a page
If the user refreshes a page (presses F5), the browser asks the server if any of the files the page consists of have changed. This can lead to a large number of requests which are answered by 304 Not Modified. In this case these files are fetched from the browser cache. However, the duration of these validity checks can be very long. With a fast internet connection this may be only a few milliseconds but with a slow one this can take several seconds.

CTRL-refresh a page
If the user triggers a complete refresh (presses CTRL+F5) the complete browser cache is purged and all files are reloaded from the server.

5.5 Improving PHP Performance

This section describes how the performance of the PHP interpreter can be improved.

5.5.1 eAccelerator

The whole server side wiki software is programmed in PHP. For every request the main entry script index.php has to load up to several hundred PHP files which have to be compiled to byte code which can then be executed by the PHP interpreter. There are several PHP accelerator that cache the byte code and provide it quickly when needed by the interpreter.

eAccelerator is one of these accelerators. Find the installation instructions at

- http://eaccelerator.net/wiki/InstallFromSource or
- http://eaccelerator.de/

With this acceleration a typical request is processed about twice as fast.

5.5.2 Memcache

MediaWiki has its own caches similar to the parser cache, a message cache and more. The data to be cached can be stored in memory across different requests and even different servers with the store Memcached. Its installation is recommended for servers with a high load. See http://www.mediawiki.org/wiki/Memcached for installation instructions.

For Windows there is a compiled binary version of Memcached ready to use. It can be downloaded at http://code.jellycan.com/memcached/. The easiest way to install Memcached is the following:

1. Download the zip archive `memcached-1.2.6-win32-bin.zip` and extract it into an arbitrary directory
2. Open a command line window and change to the directory where the memcached.exe was extracted to.
3. run once "`memcached.exe -d install`" which will install the memcached server as a service.
4. run then "`memcached.exe -d start`" to start the service

On the next reboot Memcached is started automatically.

The Memcached server appears in the windows list of all services and can be stopped and restarted there. To remove Memcached from your system:

1. stop the Memcached server (via the Windows administration panel or on the command line with "`memcached.exe -d stop`")
2. remove the service by running on the command line "memcached.exe -d uninstall"
3. delete the executable

5.6 Apache Configuration

Normally Apache is used as web server for MediaWiki. Some settings can improve the performance of Apache. If you change any settings of the Apache webserver or of the PHP installation you must restart the webserver so that the changes come into effect. On Linux this is done with

```
/etc/init.d/apache2 restart
```

If you use Windows with the XAMPP installation go into the XAMPP installation directory and double click on `xampp_restart.exe`.

Enable compression for Apache

All modern browsers can receive and process compressed files. Apache should be configured to compress them on demand. See http://httpd.apache.org/docs/2.0/mod/mod_deflate.html.
Enable it by:

```
a2enmod deflate
```

or in a XAMPP installation remove the # of the line:

```
LoadModule deflate_module modules/mod_deflate.so
```

You also need to set which files should be compressed in the Apache settings. To use compression on CSS, Javascripts, html and XML and plain text files edit the http.conf file and add:

```
<IfModule deflate_module>
    AddOutputFilterByType   DEFLATE   text/html   text/xml
text/css application/x-javascript text/javascript
</IfModule>
```

Finally restart the Apache.

Enable the cache of Apache
Static files that are often delivered by Apache can be cached. See http://httpd.apache.org/docs/2.2/mod/mod_cache.html for more details. The following command lines activate a memory cache:

```
cd /etc/apache2/
a2enmod cache mem_cache
/etc/init.d/apache2 restart
```

Before you may use the cache, it must also be configured in the http.conf file.

Add expiration to HTTP header of css, js and images
Certain HTTP headers are required by the browser cache. The field "Expires" can be added with the following command lines. See http://httpd.apache.org/docs/2.0/mod/mod_expires.html for more details.

```
a2enmod expires
```

or in an XAMPP installation edit the http.conf file and remove the # of the line:

```
LoadModule expires_module modules/mod_expires.so
```

Then edit the Apache config file:

```
# enable expirations
ExpiresActive On
ExpiresByType text/* "access plus 2 months"
ExpiresByType  application/x-javascript  "access  plus  2
months"
ExpiresByType image/* "access plus 2 months"
```

And then restart Apache.

Note:

We have observed that files that get compressed by the deflate module suddenly cannot be cached by the Browser anymore. So if you don't use any squid or other proxy, it may be useful to find a good trade off between compressing content and setting an expire header. One suggestion is to compress only html and text which is normaly generated by Medawiki anyway so that css, javascript and image files are not compressed but can be cached by the browser.

6 GNU Free Documentation License

6.1 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

6.2 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not

generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

6.3 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

6.4 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the

front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

6.5 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal

authors, if it has fewer than five), unless they release you from this requirement.

- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

6.6 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6.7 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection,

provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

6.8 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

6.9 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

6.10 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify,

sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

6.11 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

6.12 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.