Bitnami MediaWiki

MediaWiki is a wiki package originally written for Wikipedia. It is now used by several other projects of the non-profit Wikimedia Foundation and by many other wikis. MediaWiki was designed to be run on a large server farm for a website that gets millions of hits per day.

Please, take a look to the Quick Start Guide to know the basic use of this Stack.

How to start/stop the servers?

Cloud Server

Each Bitnami stack includes a control script that lets you easily stop, start and restart servers.

The script is located at */opt/bitnami/ctlscript.sh*. Call it without any arguments to restart all services:

```
$ sudo /opt/bitnami/ctlscript.sh start
```

Or use it to restart a specific service only by passing the service name as argument - for example 'mysql':

```
$ sudo /opt/bitnami/ctlscript.sh restart mysql
```

Virtual Machine

Each Bitnami stack includes a control script that lets you easily stop, start and restart servers.

The script is located at */opt/bitnami/ctlscript.sh*. Call it without any arguments to restart all services:

```
$ sudo /opt/bitnami/ctlscript.sh start
```

Or use it to restart a specific service only by passing the service name as argument - for example 'mysql':

```
$ sudo /opt/bitnami/ctlscript.sh restart mysql
```

Native Installer

You can use either the graphical manager tool or the command-line tool to start and stop the servers.

Graphical Tool

Bitnami stacks include a graphical tool to manage the servers easily (native installers only). Using this tool, you can start, stop and restart the servers and check the log files.

manager-servers.png

On Windows:

- Double-click the "manager-windows.exe" file in your installation directory.
- You can also start the Manager tool using the Start Menu path Start -> Program Files -> Bitnami Stack -> Manager.

On Linux and Mac OS X:

- Double-click the "manager-osx" or "manager-linux-*" tool in your installation directory.

Command-line Tool

Each Bitnami stack includes a control script that lets you easily stop, start and restart servers.

The control script is only available for Linux and Mac OS X native installers. The script is located in your installation directory and named *ctlscript.sh*. Call it without any arguments to restart all services.

On Linux:
If your installation directory is /home/*USER*/wordpress-4.0.1-0, call the control script with the ‘start’ argument to start all servers. For example:

```
$ cd /home/USER/wordpress-4.0.1-0
$ ./ctlscript.sh start
```

Or use it to restart a specific service only by passing the service name as argument - for example 'mysql':

```
$ cd /home/USER/wordpress-4.0.1-0
$ ./ctlscript.sh restart mysql
```

On Mac OS X:
If your installation directory is /Applications/wordpress-4.0.1-0, call the control script with the ‘start’ argument to start all servers. For example:

```
$ cd /Applications/wordpress-4.0.1-0
$ ./ctlscript.sh start
```

Or use it to restart a specific service only by passing the service name as argument - for example 'mysql':

```
$ cd /Applications/wordpress-4.0.1-0
$ ./ctlscript.sh restart mysql
```

Tabs end

How to change the default URL?

This approach describes how to configure your application to **run in the root URL directly**. Also, you will be able to **modify the URL to a NEW_DOMAIN** using the bnconfig tool. The details are described below.

Automatic Approach

This approach is based on the Bitnami Configuration Tool (bnconfig).

Bitnami Cloud Hosting

The best way to change your URL in BCH is to go to your application tab and modify it there. In the Bitnami Cloud Hosting console, select

Servers, choose your server, Manage and go to the Applications tab. Press there the pencil next to the application which URL you want to modify and choose Use Custom Domain.

Refer to this guide for more information.

Cloud Images and Virtual Machines

Moving the application to /

If your application is running in "/mediawiki" you can remove the prefix from the URL executing the following command:

```
$ sudo /opt/bitnami/apps/mediawiki/bnconfig --appurl /
```

(use --help to check if that option is available for your application)

Now you will be able to access to the application at http://YOUR_DOMAIN instead of http://YOUR_DOMAIN/mediawiki.

Updating the IP or hostname

Some applications require to update the IP/domain if the machine IP/domain changes. The bnconfig tool also has an option which updates the IP automatically during boot, called machine_hostname (use --help to check if that option is available for your application). Note that this tool changes the URL to http://NEW_DOMAIN/mediawiki.

```
sudo /opt/bitnami/apps/mediawiki/bnconfig --machine_hostname NEW_DOMAIN
```

If you already moved your application to the root URL you should include both options at the same time.

```
sudo /opt/bitnami/apps/mediawiki/bnconfig --appurl / --machine_hostname
```

If you have configured your machine to use an static domain name or IP, you **should rename or remove** the "/opt/bitnami/apps/mediawiki/bnconfig" file.

```
sudo mv /opt/bitnami/apps/mediawiki/bnconfig /opt/bitnami/apps/mediawiki
```

Native Installer

Remember to use your actual installation directory instead of installdir.

Moving the application to /

If your application is running in "/mediawiki" you can remove the prefix from the URL executing the following command:

On Linux,

```
installdir/apps/mediawiki/bnconfig --appurl /
```

On Mac OS X,

```
installdir/apps/mediawiki/bnconfig.app/Contents/MacOS/installbuilder.sh
```

On Windows,

```
installdir/apps/mediawiki/bnconfig.exe --appurl /
```

(use --help to check if that option is available for your application)

Now you will be able to access to the application at http://YOUR_DOMAIN instead of http://YOUR_DOMAIN/mediawiki.

Updating the IP or hostname

Some applications require to update the IP/domain if the machine IP/domain changes. The bnconfig tool also has an option which updates the IP , called machine_hostname (use --help to check if that option is available for your application). Note that this tool changes the URL to http://NEW_DOMAIN/mediawiki.

```
installdir/apps/mediawiki/bnconfig --machine_hostname NEW_DOMAIN
```

If you already moved your application to the root URL you should include both options at the same time.

```
installdir/apps/mediawiki/bnconfig --appurl / --machine_hostname NEW_DOM
```

Tabs end

Manual Approach

If you want to change the default URL from http://your_domain/mediawiki to http://your_domain, edit the *installdir/apps/mediawiki/conf/httpd-prefix.conf* file so that it looks like the file below:

```
DocumentRoot "/installdir/apps/mediawiki/htdocs"
# Alias /mediawiki/ "/installdir/apps/mediawiki/htdocs/"
# Alias /mediawiki "/installdir/apps/mediawiki/htdocs"

(...)
```

Remember that you must substitute *installdir* with your actual installation directory (for native installers) or */opt/bitnami* (for cloud images and virtual machines).

Some applications also require additional changes in their configuration files or in their database.

In the MediaWiki case it is also necessary to update the "LocalSettings.php" file. Replace the following to move your application to the root URL.

```
$wgScriptPath       = "/mediawiki";
```

with

```
$wgScriptPath       = "";
```

Moreover, you need to modify the httpd-app.conf file, in installdir/apps/mediawiki/httpd-app.conf, and replace these two lines

```
RewriteBase /mediawiki/
...
RewriteRule . /mediawiki/index.php [L]
```

with

```
# RewriteBase /mediawiki/
...
RewriteRule . index.php [L]
```

How to create a full backup of MediaWiki?

Backup

Bitnami stacks are self-contained and the simplest option for performing a backup is to copy or compress the Bitnami stack installation directory. To do so in a safe manner, you will need to stop all servers, so this method may not be appropriate if you have people accessing the application continuously.

Cloud Server

Follow these steps:

- Change to the directory in which you wish to save your backup.

  ```
  cd /your/directory
  ```

- Stop all servers.

  ```
  $ sudo /opt/bitnami/ctlscript.sh stop
  ```

- Create a compressed file with the stack contents.

  ```
  $ sudo tar -pczvf application-backup.tar.gz /opt/bitnami
  ```

- Restart all servers.

  ```
  $ sudo /opt/bitnami/ctlscript.sh start
  ```

You should now download or transfer the *application-backup.tar.gz* file to a safe location.

Virtual Machine

Follow these steps:

- Change to the directory in which you wish to save your backup.

  ```
  cd /your/directory
  ```

- Stop all servers.

  ```
  $ sudo /opt/bitnami/ctlscript.sh stop
  ```

- Create a compressed file with the stack contents.

  ```
  $ sudo tar -pczvf application-backup.tar.gz /opt/bitnami
  ```

- Restart all servers.

  ```
  $ sudo /opt/bitnami/ctlscript.sh start
  ```

You should now download or transfer the *application-backup.tar.gz* file to a safe location.

Native Installer (Windows)

Follow these steps:

- Stop all servers using the shortcuts in the Start Menu or the graphical manager tool.
- Create a compressed file with the stack contents. You can use a graphical tool like 7-Zip or WinZip.
- Stop all servers using the shortcuts in the Start Menu or the graphical manager tool.

You should now download or transfer the *application-backup.zip* file to a safe location.

Native Installer (Linux and Mac OS X)

Follow these steps:

- Change to the directory in which you wish to save your backup.

  ```
  cd /your/directory
  ```

- Stop all servers.

  ```
  $ sudo installdir/ctlscript.sh stop
  ```

- Create a compressed file with the stack contents.

  ```
  $ sudo tar -pczvf application-backup.tar.gz installdir
  ```

- Restart all servers.

  ```
  $ sudo installdir/ctlscript.sh start
  ```

You should now download or transfer the *application-backup.tar.gz* file to a safe location.

Tabs end

Restore

Bitnami stacks are self-contained, so to restore a stack, you only need to uncompress the backup file in the same location. It is important to use the same path that was used when the stack was originally installed.

Cloud Server

Follow these steps:

- Change to the directory containing your backup.

  ```
  cd /your/directory
  ```

- Stop all servers.

  ```
  $ sudo /opt/bitnami/ctlscript.sh stop
  ```

- Rename the current directory to save it.

  ```
  $ sudo mv /opt/bitnami /opt/bitnamiBackup
  ```

- Uncompress the backup file to the original directory.

  ```
  $ sudo tar -pxzvf application-backup.tar.gz -C /
  ```

- Start all servers.

  ```
  $ sudo /opt/bitnami/ctlscript.sh start
  ```

Virtual Machine

Follow these steps:

- Change to the directory containing your backup.

  ```
  cd /your/directory
  ```

- Stop all servers.

```
$ sudo /opt/bitnami/ctlscript.sh stop
```

- Rename the current directory to save it.

```
$ sudo mv /opt/bitnami /opt/bitnamiBackup
```

- Uncompress the backup file to the original directory.

```
$ sudo tar -pxzvf application-backup.tar.gz -C /
```

- Start all servers.

```
$ sudo /opt/bitnami/ctlscript.sh start
```

Native Installer (Windows)

Follow these steps:

- Uncompress the backup file to the original directory.
- Install services by launching a new command prompt and executing the following commands. Administrator privileges are required.

```
$ cd installdir
$ serviceinstall.bat INSTALL
```

You can now start or stop servers using the graphical manager tool.

Native Installer (Linux and Mac OS X)

Follow these steps:

- Change to the directory containing your backup.

```
cd /your/directory
```

- Stop all servers.

```
$ sudo /opt/bitnami/ctlscript.sh stop
```

- Rename the current directory to save it.

```
$ sudo mv installdir installdirBackup
```

- Uncompress the backup file to the original directory.

  ```
  $ sudo tar -pxzvf application-backup.tar.gz -C /
  ```

- Start all servers.

  ```
  $ sudo installdir/ctlscript.sh start
  ```

Tabs end

IMPORTANT: When restoring, remember to maintain the original permissions for the files and folders. For example, if you originally installed the stack as 'root', make sure that the restored files are owned by 'root'.

If you want to create only a database backup, refer to these instructions for MySQL and PostgreSQL.

How to configure the email settings of MediaWiki?

It is necessary to install two PEAR packages: Mail and Net_SMTP. Go to the Bitnami Console and run the following command. If you are using the Virtual appliance or the AMI you should specify the "sudo" command before:

```
$ pear install mail
$ pear install net_smtp
```

Then add the following code at the bottom of the *installdir*/apps/mediawiki/htdocs/LocalSettings.php file according to your SMTP server settings. In this example we are configuring it using a GMail account.

```
$wgSMTP = array(
        'host' => 'ssl://smtp.gmail.com',
        'IDHost' => 'gmail.com',
        'port' => 465,
        'username' => 'your_account@gmail.com',
        'password' => 'your_password',
        'auth' => true
);
```

Restart the servers and that's all.

How to enable SSL?

You can see how to configure Apache to enable SSL connections at [How to enable SSL to access through https?]()

How to debug MediaWiki errors?

Cloud Server

Once Apache starts, it will create two log files at */opt/bitnami/apache2/logs/access_log* and */opt/bitnami/apache2/logs/error_log* respectively. On Amazon Linux and Red Hat Enterprise cloud images, the log files are created at */var/log/httpd/access_log* and */var/log/httpd/error_log* instead.

- The *access_log* file is used to track client requests. When a client requests a document from the server, Apache records several parameters associated with the request in this file, such as: the IP address of the client, the document requested, the HTTP status code, and the current time.
- The *error_log* file is used to record important events. This file includes error messages, startup messages, and any other significant events in the life cycle of the server. This is the first place to look when you run into a problem when using Apache. If no error is found, you will see a message similar to:

```
Syntax OK
/installdir/ctlscript.sh : httpd started
```

Virtual Machine

Once Apache starts, it will create two log files at */opt/bitnami/apache2/logs/access_log* and */opt/bitnami/apache2/logs/error_log* respectively.

- The *access_log* file is used to track client requests. When a client requests a document from the server, Apache records several parameters associated with the request in this file, such as: the IP address of the client, the document requested, the HTTP status code, and the current time.
- The *error_log* file is used to record important events. This file includes error messages, startup messages, and any other significant events in the life cycle of the server. This is the first place to look when you run into a problem when using Apache. If no error is found, you will see a message similar to:

```
Syntax OK
/installdir/ctlscript.sh : httpd started
```

Native Installer

Once Apache starts, it will create two log files at *installdir/apache2/logs/access_log* and *installdir/apache2/logs/error_log* respectively.

- The *access_log* file is used to track client requests. When a client requests a document from the server, Apache records several parameters associated with the request in this file, such as: the IP address of the client, the document requested, the HTTP status code, and the current time.
- The *error_log* file is used to record important events. This file includes error messages, startup messages, and any other significant events in the life cycle of the server. This is the first place to look when you run into a problem when using Apache. If no error is found, you will see a message similar to:

```
Syntax OK
/installdir/ctlscript.sh : httpd started
```

Cloud Server

The main MySQL log file is created at */opt/bitnami/mysql/data/mysqld.log*.

Virtual Machine

The main MySQL log file is created at */opt/bitnami/mysql/data/mysqld.log*.

Native Installer

The main MySQL log file is created at *installdir/mysql/data/mysqld.log*.

How to increase the allowed size of the uploaded files?

You can modify the following option in the php.ini file to increase the allowed size for uploads:

```
; Maximum size of POST data that PHP will accept.
post_max_size = 16M
```

```
...

; Maximum allowed size for uploaded files.
upload_max_filesize = 16M
```

If you have enabled PHP-FPM (enabled by default in **Cloud Images** and **VMs**) you need to restart PHP-FPM running the following command:

```
sudo /opt/bitnami/ctlscript.sh restart php-fpm
```

Note: For native installers replace /opt/bitnami with your current installation directory.

Otherwise, you need to restart the Apache server:

```
sudo YOUR_INSTALLATION_DIRECTORY/ctlscript.sh restart apache
```

How to change the MediaWiki URLs to pretty URL?

> Pretty links have been included by default in Bitnami MediaWiki for Unix systems since version 1.22.2-1. It is not possible to include them in in Windows systems, due to the colon (:) in the Special pages URL.
>
> https://issues.apache.org/bugzilla/show_bug.cgi?id=41441

The following steps can be done to convert MediaWiki URLs to pretty URLs

Add the following lines to the /installation_directory/apps/mediawiki/conf/httpd-app.conf file, inside the Directory directive

```
<Directory "/installation_directory/apps/mediawiki/htdocs">
....
RewriteEngine On
RewriteBase /mediawiki/
RewriteRule ^index\.php$ - [S=1]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . /mediawiki/index.php [L]
...
</Directory>
```

Add the following lines at the end of the /installation_directory/apps/mediawiki/htdocs/LocalSettings.php

```
$wgArticlePath = "/mediawiki/$1";
$wgUsePathInfo = true;
```

After that, restart the server and all the links will be pretty links.

How to install ParserFunctions extension on MediaWiki?

If you want to install the ParserFunctions extension on MediaWiki, please follow this steps:

- Go to the MediaWiki extensions folder:

```
cd /opt/bitnami/apps/mediawiki/htdocs/extensions
```

- Create the ParserFunctions directory:

```
mkdir ParserFunctions
```

- Go to the new directory:

```
cd ParserFunctions
```

- Download the extension from http://www.mediawiki.org/wiki/Extension:VisualEditor

```
wget -O parserfunctions.zip https://git.wikimedia.org/zip/?r=mediawiki/e
```

- Edit */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* and add this code at the end of the file

```
require_once( "$IP/extensions/ParserFunctions/ParserFunctions.php" );
```

- If you want to use the integrated string funciont functionality, add just after that line:

```
$wgPFEnableStringFunctions = true;
```

- Restart the Apache server

```
/opt/bitnami/ctlscript.sh restart apache
```

How to install VisualEditor extension on MediaWiki?

If you want to install the VisualEditor extension on MediaWiki, please follow this steps:

- Go to the MediaWiki extensions folder:

```
cd /opt/bitnami/apps/mediawiki/htdocs/extensions
```

- Create the VisualEditor directory:

```
mkdir VisualEditor
```

- Go to the new directory:

```
cd VisualEditor
```

- Download the extension from http://www.mediawiki.org/wiki/Extension:VisualEditor:

```
wget -O visualeditor.zip https://git.wikimedia.org/zip/?r=mediawiki/exte
```

- Edit *`/opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php`* and add this code at the end of the file:

```
require_once("$IP/extensions/VisualEditor/VisualEditor.php");

// OPTIONAL: Enable VisualEditor in other namespaces
// By default, VE is only enabled in NS_MAIN
//$wgVisualEditorNamespaces[] = NS_PROJECT;

// Enable by default for everybody
$wgDefaultUserOptions['visualeditor-enable'] = 1;

// Don't allow users to disable it
$wgHiddenPrefs[] = 'visualeditor-enable';

// OPTIONAL: Enable VisualEditor's experimental code features
//$wgVisualEditorEnableExperimentalCode = true;
```

- Restart the Apache server:

```
/opt/bitnami/ctlscript.sh restart apache
```

Note that you will be able to use VisualEditor to create new Wiki Pages, but not for edit them. If you want to use VisualEditor to edit them, you will need to follow the official Parsoid installation guide: http://www.mediawiki.org/wiki/Extension:VisualEditor#Setup_a_Parsoid_service

How to install Parsoid extension on MediaWiki?

If you want to install the Parsoid extension on MediaWiki, please follow this steps:

- Go to the MediaWiki extensions folder:

```
cd /opt/bitnami/apps/mediawiki/htdocs/extensions
```

- Download the extension from https://www.mediawiki.org/wiki/Extension:Parsoid. For Mediawiki version 1.25.1:

```
wget https://extdist.wmflabs.org/dist/extensions/Parsoid-REL1_25-f5500c1
```

- Extract the code:

```
tar -xzf Parsoid-REL1_25-f5500c1.tar.gz
```

- Edit */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* and add this code at the end of the file

```
wfLoadExtension( 'Parsoid' );
```

- Restart the Apache server

```
/opt/bitnami/ctlscript.sh restart apache
```

- Check that Parsoid extension is installed visiting page Special:Version in your MediaWiki page. You will see something like

Note: if you are using MediaWiki 1.24 or earlier, you will need to add line

```
require_once "$IP/extensions/Parsoid/Parsoid.php";
```

to your LocalSettings.php file instead.

How to install Math extension on MediaWiki?

If you want to install the Math extension on MediaWiki, please follow this steps:

- Go to the MediaWiki extensions folder:

```
cd /opt/bitnami/apps/mediawiki/htdocs/extensions
```

- Download the extension from https://www.mediawiki.org/wiki/Extension:Math. For Mediawiki version 1.25.1:

```
wget https://extdist.wmflabs.org/dist/extensions/Math-REL1_25-2998273.ta
```

- Extract the code:

```
tar -xzf Math-REL1_25-2998273.tar.gz
```

- Edit the permissions

```
sudo chmod 775 Math/
sudo chown -R bitnami:daemon Math/
```

- Edit */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* and add this code at the end of the file

```
require_once ( "$IP/extensions/Math/Math.php" );
```

- Run the update script which will automatically create the necessary database tables that this extension needs.

```
sudo /opt/bitnami/use_mediawiki
cd /opt/bitnami/apps/mediawiki/htdocs
php maintenance/update.php
```

- Check that Math extension is installed visiting page Special:Version in your MediaWiki page. You will see something like

Screen Shot 2015-08-11 at 12.01.35.png

How to upgrade MediaWiki?Edit section

It is strongly recommended that you create a backup before starting the update process. If you have important data, it is advisable that you create and try to restore a backup to ensure that everything works properly. You can get more info about how to create a full backup here.

Of course, before executing any command, you have to start the bitnami_console script or the shortcut on Windows before typing the following commands.

There are two different ways to upgrade your application.

1. If you want to upgrade the application and all Stack components PHP, MySQL, Apache… You can follow the steps described at How to upgrade the full Stack migrating the data?
2. In case you only want to upgrade the application code without modifying any other Stack components. You can find below the steps to upgrade only the MediaWiki application.

In the MediaWiki case, these are the steps to **migrate the database** from an old version to a new one. You can launch a new MediaWiki version instance or download the latest MediWiki installer from http://bitnami.com/stack/mediawiki/. Note this guide uses "/opt/bitnami" as the default installation directory, you can use a different one if you have installed it in a different folder.

- Copy the database backup to the new Bitnami version server.

- Stop all servers and start only MySQL. Note that the installation directory could be different. This is the default installation directory for Virtual Machines and Cloud images.

  ```
  $ sudo /opt/bitnami/ctlscript.sh stop
  $ sudo /opt/bitnami/ctlscript.sh start mysql
  ```

- Remove the previous database and create the new one. You can configure the database user password with a secure password.

  ```
  $ mysql -u root -p
  Password: ****
  mysql> drop database bitnami_mediawiki;
  mysql> create database bitnami_mediawiki;
  mysql> grant all privileges on bitnami_mediawiki.* to 'bn_mediawiki
  ```

- Restore the new database:

  ```
  $ mysql -u root -p bitnami_mediawiki < backup.sql
  ```

- Edit the MediWiki configuration file to update the database user password (the same that you set previously) "/opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php":

```
$wgDBpassword = "DATABASE_PASSWORD";
```

- Restart the servers and you can access to your new MediaWiki.
- Copy the ""/opt/bitnami/apps/mediawiki/htdocs/files" folder from the old installation to the new one.
- If you have installed plugins in the previous version, copy the folders from "vendor/plugins" directory into new installation directory, in "plugins" folder.

```
$ sudo /opt/bitnami/ctlscript.sh restart
```

How to upgrade MediaWiki database?

You can upgrade the MediaWiki database in two ways: Either from the command line or from the web browser (since version 1.17). Upgrading from the command line is recommended, since this reduces the risk of the upgrade process being interrupted by a timeout or connection reset.

Command line

From the command line execute:

```
$ cd installdir/apps/mediawiki/htdocs
$ php maintenance/update.php
```

First make sure that your web server (such as Apache) and your database (such as MySQL) are running.

MediaWiki will inspect the existing schema and update it to work with the new code, adding tables and columns as needed.

If you use a shared_database, you should pass the `--doshared` parameter if you want the shared tables to be updated. Otherwise they won't be touched by the update script.

Web browser

If your database is already big and in high production usage, then you should not be using the Web updater, e.g. because the update process will time out when the *maximum_execution_time* is reached. In that case you should use update.php from the command-line interface (not from the web). .

1. Access the web-updater.Always backup before performing database maintenance.

2. Navigate your webbrowser to **`/mw-config/`**. For example, if your wiki is at mydomain.com/mediawiki/Main_Page, then navigate to mydomain.com/mediawiki/mw-config/.

3. Select your language and click continue.

4. The existing installation should be detected. Follow the instructions on the screen to upgrade it.
   If asked for the "upgrade key", open apps/mediawiki/htdocs/LocalSettings.php file and look for the key assigned to $wgUpgradeKey.

How to install Semantic Bundle extension?

Semantic Bundle (abbreviated as SB) is a pre-packaged bundle of extensions for use with wikis that are based around the Semantic MediaWiki extension. It is a suitable alternative if you cannot run Composer to install or upgrade SMW (1.9+) and a number of other extensions that require Composer, such as Maps (3.0+). Do not run Composer if you want to install Semantic Bundle.

Follow the steps for your platform.

Bitnami Cloud Hosting

Follow these steps:

1. Download the ZIP or TGZ archive and unzip it to */opt/bitnami/apps/mediawiki/htdocs/extensions/*. For example, assuming you downloaded the archive in */tmp*:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/
$ unzip /tmp/SemanticBundle-20140514.zip
```

2. Change to the extension directory and create the configuration file:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle
$ cp SemanticBundleSettings.sample.php SemanticBundleSettings.php
```

3. Include the *SemanticBundleSettings.php* and *SemanticBundle.php* files in the */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* file:

```
require_once( "$IP/extensions/SemanticBundle/SemanticBundleSettings.php"
require_once( "$IP/extensions/SemanticBundle/SemanticBundle.php" );
```

4. Log in to MediaWiki as an administrator.

5. Navigate to the page at *http://[your-server]/mediawiki/Special:SMWAdmin* and click the "Initialize or upgrade tables" button.

6. After successful initialization, edit the */opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle/SemanticBundleSettings.php* file and uncomment the extensions you would like to use.

Microsoft Azure

Follow these steps:

1. Download the ZIP or TGZ archive and unzip it to */opt/bitnami/apps/mediawiki/htdocs/extensions/*. For example, assuming you downloaded the archive in */tmp*:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/
$ unzip /tmp/SemanticBundle-20140514.zip
```

2. Change to the extension directory and create the configuration file:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle
$ cp SemanticBundleSettings.sample.php SemanticBundleSettings.php
```

3. Include the *SemanticBundleSettings.php* and *SemanticBundle.php* files in the */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* file:

```
require_once( "$IP/extensions/SemanticBundle/SemanticBundleSettings.php"
require_once( "$IP/extensions/SemanticBundle/SemanticBundle.php" );
```

4. Log in to MediaWiki as an administrator.

5. Navigate to the page at *http://[your-server]/Special:SMWAdmin* and click the "Initialize or upgrade tables" button.

6. After successful initialization, edit the */opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle/SemanticBundleSettings.php* file and uncomment the extensions you would like to use.
Google Cloud Platform

Follow these steps:

1. Download the ZIP or TGZ archive and unzip it to */opt/bitnami/apps/mediawiki/htdocs/extensions/*. For example, assuming you downloaded the archive in */tmp*:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/
$ unzip /tmp/SemanticBundle-20140514.zip
```

2. Change to the extension directory and create the configuration file:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle
$ cp SemanticBundleSettings.sample.php SemanticBundleSettings.php
```

3. Include the *SemanticBundleSettings.php* and *SemanticBundle.php* files in the */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* file:

```
require_once( "$IP/extensions/SemanticBundle/SemanticBundleSettings.php"
require_once( "$IP/extensions/SemanticBundle/SemanticBundle.php" );
```

4. Log in to MediaWiki as an administrator.

5. Navigate to the page at *http://[your-server]/Special:SMWAdmin* and click the "Initialize or upgrade tables" button.

6. After successful initialization, edit the */opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle/SemanticBundleSettings.php* file and uncomment the extensions you would like to use.

Virtual Machine

Follow these steps:

1. Download the ZIP or TGZ archive and unzip it to */opt/bitnami/apps/mediawiki/htdocs/extensions/*. For example, assuming you downloaded the archive in */tmp*:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/
$ unzip /tmp/SemanticBundle-20140514.zip
```

2. Change to the extension directory and create the configuration file:

```
$ cd /opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle
$ cp SemanticBundleSettings.sample.php SemanticBundleSettings.php
```

3. Include the *SemanticBundleSettings.php* and *SemanticBundle.php* files in the */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php* file:

```
require_once( "$IP/extensions/SemanticBundle/SemanticBundleSettings.php"
require_once( "$IP/extensions/SemanticBundle/SemanticBundle.php" );
```

4. Log in to MediaWiki as an administrator.

5. Navigate to the page at *http://[your-server]/Special:SMWAdmin* and click the "Initialize or upgrade tables" button.

6. After successful initialization, edit the */opt/bitnami/apps/mediawiki/htdocs/extensions/SemanticBundle/SemanticBundleSettings.php* file and uncomment the extensions you would like to use.

Native Installer

Follow these steps:

1. Download the ZIP or TGZ archive and unzip it to *installdir/apps/mediawiki/htdocs/extensions/*. For example, assuming you downloaded the archive in */tmp*:

```
$ cd installdir/apps/mediawiki/htdocs/extensions/
$ unzip /tmp/SemanticBundle-20140514.zip
```

2. Change to the extension directory and create the configuration file:

```
$ cd installdir/apps/mediawiki/htdocs/extensions/SemanticBundle
$ cp SemanticBundleSettings.sample.php SemanticBundleSettings.php
```

3. Include the *SemanticBundleSettings.php* and *SemanticBundle.php* files in the *installdir/apps/mediawiki/htdocs/LocalSettings.php* file:

```
require_once( "$IP/extensions/SemanticBundle/SemanticBundleSettings.php"
require_once( "$IP/extensions/SemanticBundle/SemanticBundle.php" );
```

4. Log in to MediaWiki as an administrator.

5. Navigate to the page at *http://[your-server]/mediawiki/Special:SMWAdmin* and click the "Initialize or upgrade tables" button.

6. After successful initialization, edit the *installdir/apps/mediawiki/htdocs/extensions/SemanticBundle/SemanticBundleSettings.php* file and uncomment the extensions you would like to use.

Tabs end

NOTE: It's strongly recommended to initialize Semantic MediaWiki before you include (uncomment) any of the other SemanticBundle extensions, to avoid potential errors from extensions that require Semantic MediaWiki .

How to install Semantic MediaWiki extension on MediaWiki with Composer?

You can install Semantic MediaWiki using composer. To do so, please run the following commands:

```
cd  /opt/bitnami/apps/mediawiki/htdocs
php /opt/bitnami/php/bin/composer.phar update
php /opt/bitnami/php/bin/composer.phar require mediawiki/semantic-media-w
php maintenance/update.php
```

Note: For native installers you will need to replace /opt/bitnami with your MediaWiki installation directory.

If the last command (update.php) wasn't successful, please try updating from the browser following these steps: https://wiki.bitnami.com/Applications/BitNami_MediaWiki#Web_browser

Finally, enable semantics for your domain, adding this line at the end of your /opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php. You need to use your own MediaWiki domain instead of example.com:

```
enableSemantics( 'example.com' );
```

If you get a warning like this:

> Warning: This development build of composer is over 30 days old. It is recommended to update it by running "/opt/bitnami/php/bin/composer.phar self-update" to get the latest version.

Execute the suggested command and try the steps again.

For more information, please refer to: http://www.semantic-mediawiki.org/wi...n#Installation

How to install Collection extension on MediaWiki?

The **Collection** extension allows you to organize personal selections of pages in a collection. To do it, the extension uses a public render server where the jobs are sent and processed.

Please, keep in mind that if your Mediawiki installation is not public (it is a local installation or it can't be accessed from Internet), you will need to install your own Render Server. The Collection developers have some suggestions about it: http://www.mediawiki.org/wiki/Extension:Collection#Set_up_your_own_render_server

To install the Collection plugin, please follow the steps below:

- Download the latest Collection version from the official Mediawiki extension page: http://www.mediawiki.org/wiki/Special:ExtensionDistributor/Collection
- Uncompress the downloaded file in the *extensions* directory of Mediawiki. The command below should create the *Collection* directory in the Mediawiki *extensions* directory:

```
tar -xzf Collection-*.tar.gz -C /opt/bitnami/apps/mediawiki/htdocs/extens
```

- Add the *Collection* plugin to the Mediawiki *LocalSettings.php* file:

```
echo 'require_once "$IP/extensions/Collection/Collection.php";' >> /opt/b
```

Once you have done all the steps above, you should see new options in the Mediawiki menu:

Collection_extension_mediawiki_2.png

If you already have your own Renderserver, remember to add the url in */opt/bitnami/apps/mediawiki/htdocs/LocalSettings.php*:

```
$wgCollectionMWServeURL = "http://RENDERSERVER_URL:PORT";
```

How to install MultimediaViewer extension on MediaWiki?

The **MultimediaViewer** extension gives the user of a wiki a different interface for viewing full-size, or nearly full-size, images in their browser without extraneous page loads or confusing interstitial pages.

To install the MultimediaViewer plugin, please follow the steps below:

- Download the latest MultimediaViewer version from the official Mediawiki extension page: http://www.mediawiki.org/wiki/Extension:MultimediaViewer
- Uncompress the downloaded file in the *extensions* directory of Mediawiki. The command below should create the *Collection* directory in the Mediawiki *extensions* directory:

```
tar -xzf MultimediaViewer-*.tar.gz -C /opt/bitnami/apps/mediawiki/htdocs/
```

- Add the MultimediaViewer plugin to the Mediawiki *LocalSettings.php* file:

```
require_once "$IP/extensions/MultimediaViewer/MultimediaViewer.php";
```

If you have enabled PageSpeed you will need to disable it for the images folder. To do that you need to edit the file /opt/bitnami/apps/mediawiki/conf/htaccess.conf and modify the images Directory. The Directory should look like this:

```
<Directory "/opt/bitnami/apps/mediawiki/htdocs/images">
# Protect against bug 28235
<IfModule rewrite_module>
        RewriteEngine On
        RewriteCond %{QUERY_STRING} \.[^\\/:*?\x22<>|%]+(#|\?|$) [nocase]
        RewriteRule . - [forbidden]
</IfModule>
ModPagespeed off
</Directory>
```