

# Archimind Installation guide

---

**Author:** Klaas Andries de Graaf - <https://github.com/kadevgraaf/>

**Version:** V2

**Date:** 15-12-2016

## Introduction

This document contains installation instructions for ArchiMind. Archimind is a semantic wiki for Software Architecture (SA) documentation management and retrieval and was adapted from OntoWiki (<http://aksw.org/Projects/OntoWiki.html>) version 0.9.5. A concise overview of adaptations and their rationale is given in paper -Klaas Andries de Graaf - "[Annotating Software Documentation in Semantic Wikis](#).-

@license (remains to be:) <http://opensource.org/licenses/gpl-license.php> GNU General Public License (GPL)

<https://github.com/AKSW/OntoWiki/wiki> provides detailed installation instructions for OntoWiki. These instructions also apply to the installation of ArchiMind.

OntoWiki 1.0.0 (<https://github.com/AKSW/OntoWiki/releases>) has been release recently - this release provides a SPARQL Query endpoint and is probably better for user experience, however, the ArchiMind adaptations for Software Architecture (SA) documentation management and retrieval are not yet implemented in this version (only 0.9.5 currently).

## Installation procedure

### Creating/config database

The authors of OntoWiki advice to use Virtuoso for performance. Otherwise MySQL 5 also works. Mysql 4 is not optimized for this type of (graph) database.

Create a database name, user, and password.

The database user and password are later used to login in to Archmind/OntoWiki as admin, and is necessary to create new knowledgebases.

in {root}/config.ini you should change the following fields to match the datase name, user, and password in your hosting environment.

```
store.zenddb.dbname  = " "
```

```
store.zenddb.username = ""
```

```
store.zenddb.password = ""
```

After copying ArchiMind installation files to the webserver (see below) call index.php. An installation script will now generate a database. (you can edit what tables are created in libraries\Erfurt\Cache\Backend\QueryCache\Database.php , e.g. when you want to create your own custom version that by default also generates a table for experimentation purposes of new plugins)

#### **Pre-configured SQL (database) dump:**

The installation package contains an SQL dump named "**archimind.sql**" in the root of the GitHub repository. This file can be imported, e.g. via PHPMyAdmin to creating a database that partially pre-configures ArchiMind. Normally one has to call index.php after copying ArchiMind files to the webserver, setup users, add ontologies, and populate the ontologies.

The pre-configured database contains a populated SA ontology or "knowledge base". Populated means that the ontology classes and relationships have already been instantiated. The knowledge base consists of AK instances, such as decisions and requirements, as well as wikipages that contain SA documentation and in-page annotations that link to the AK instances.

#### **Pre-configured in-page semantic annotations SQL (database) dump:**

Add the pre-configured in-page semantic annotations SQL (database) dump (ef\_annotation.sql) to show in-page semantic annotations for software architecture documentation content in the sample knowledge base (added via **Pre-configured SQL (database) dump** above or via **Example software architecture documentation Knowledge Base (Example SA doc ontology.rdf)** below.

#### **Separate HML content storage:**

Another optimization for performance is the storage of Wikipage HTML content in separate files instead of storage in the database. This however does requires adaptation to the search engine in order for it to search the this documentation outside the database (e.g. 1) a combination with google desktop search or 2) storage of plain text in the database and HTML in separate files maybe both be a good option for such an adaptation)

## **Copying installation files**

If you copy the folder /archimind/ to domain [www.domain.com](http://www.domain.com), archimind will be accessible via URL [www.domain.com/archimind/](http://www.domain.com/archimind/) (or another folder if you want to rename this)

## **User management**

After creating the database, one can login with administrator rights by using the same credentials as the database user (database username and password).

**Login**

Local OpenID FOAF+SSL

Username

Password

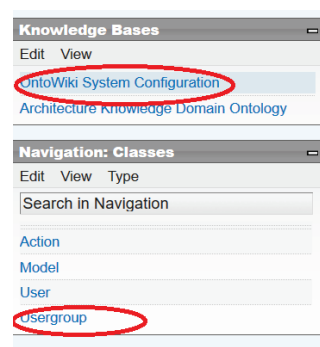
☐ Remember me

Login Register

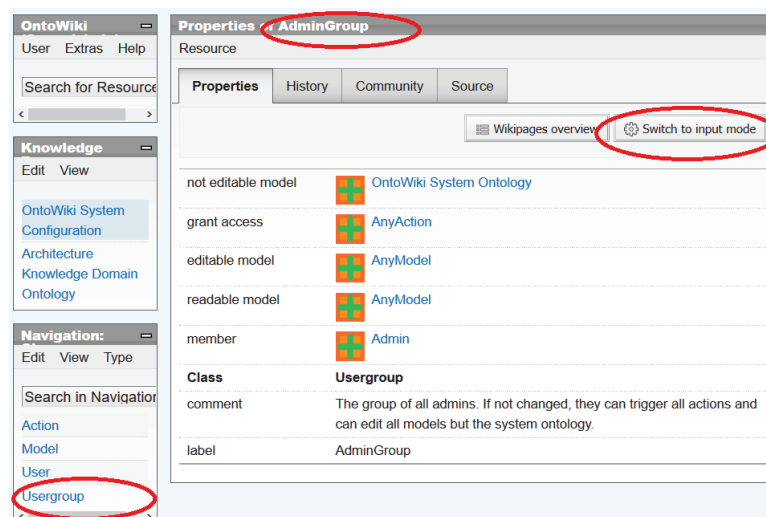
After login in as administrator, one can change the rights of existing users.

New users can be created using the top-left menu "Ontowiki" -> "user" -> "register new user".

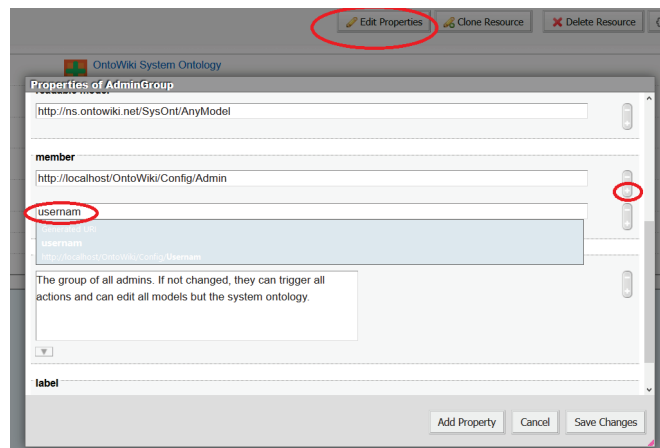
A user can be made administrator by going to top-left menu "Knowledge Bases", selecting OntoWiki System Configuration, clicking class "Usergroup",



clicking instance "admingroup", click button "switch to input mode",



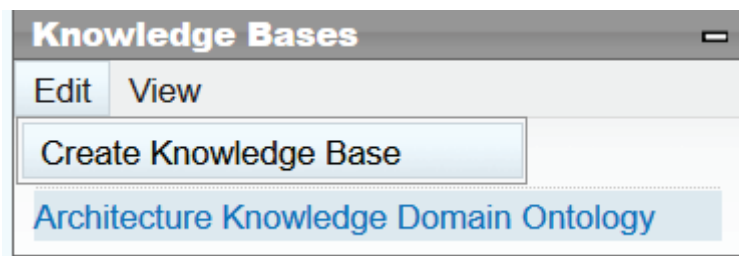
click "edit properties", and add the user to this group by adding the name of the user in property "member" (click the small + icon, type the name of the user and follow the recommended auto-suggest option that is shown).



Specific user rights can be changed by going to top-left menu "Knowledge Bases", selecting OntoWiki System Configuration, clicking class "user", clicking the specific user, click button "switch to input mode", click "edit properties", and from there add or edit properties "deny access" and "grant access" (for features login, rollback, etc..), "not readable model", readable model, and "editable model" (for seeing/editing an ontology/KB). This allows one to create a user for a specific knowledge base.

## Importing an ontology

An ontology can be imported via panel "knowledge bases", menu "Edit" and "Create Knowledge Base".



When uploading a file, the ontology (e.g. \*.rdf, \*.owl) should have a unique URI (e.g. <http://www.semanticweb.org/ontologies/2011/5/Ontology1308582601209.owl>) . Importing two ontologies with the same URI will generate an error.

### Example software architecture documentation Knowledge Base (Example SA doc ontology.rdf):

The root of the GitHub repository contains an example knowledge based (ontology + instances) for software architecture documentation named " Example SA doc ontology.rdf" which can be uploaded and used to illustrate and get to know ArchiMind. Add the pre-configured in-page semantic annotations SQL (database) dump (ef\_annotation.sql) to show in-page semantic annotations for software architecture documentation content in this knowledge base.

In my experience the most practical method of populating and creating an ontology is by using Protogé or any other offline tool for ontology construction. This since ArchiMind is not very practical for creating and populating an ontology from scratch.

It is also good to regularly backup the ontologies you are working on, as OntoWiki/ArchiMind is not bug free, and finding any errors/corruptions in an ontology you are working on is not trivial.

### Managing in-page annotations

The table EF\_annotation (Also see **Pre-configured SQL (database) dump ef\_annotation.sql**) contains all in-page annotations done on wikipages. If you incorrectly annotate something you can remove the annotation (in a single row) from this table in you database administration tool, e.g. phpmyadmin. currently there is no other way of doing this.

## References

### OntoWiki:

<http://aksw.org/Projects/OntoWiki.html>





<http://ontowiki.net/>

<https://github.com/AKSW/OntoWiki>

<https://github.com/AKSW/OntoWiki/releases>

### ArchiMind:

<https://github.com/kadevgraaf/> - <http://kadevgraaf.nl/>

- Klaas Andries de Graaf - "[Annotating Software Documentation in Semantic Wikis](#)"  - In *Proceedings of the fourth workshop on Exploiting semantic annotations in information retrieval (ESAIR '11)*, pages 5-6., ACM, 2011. [\[View paper on ACM Digital Library\]](#)
- 
- My PhD Thesis - "[Ontology-based Software Architecture Documentation](#)"  (cover)
- Klaas Andries de Graaf, Peng Liang, Antony Tang, Hans van Vliet - "[How Organisation of Architecture Documentation Affects Architectural Knowledge Retrieval](#)"  - *Science of Computer Programming*, Volume 121, Pages 75-99 Elsevier, June 2016. [\[View paper on ScienceDirect\]](#)
- Klaas Andries de Graaf, Peng Liang, Antony Tang, Hans van Vliet - "[Supporting Architecture Documentation: A Comparison of Two Ontologies for Knowledge Retrieval](#)"  - In *International Conference on Evaluation and Assessment in Software Engineering (EASE)*, pages 3:1--3:10, ACM, 2015. [\[View paper on ACM Digital Library\]](#)

- Klaas Andries de Graaf, Peng Liang, Antony Tang, Hans van Vliet - "[The Impact of Prior Knowledge on Searching in Software Documentation](#)"  - In *Proceedings of the 2014 ACM symposium on Document engineering (DocEng)*, pages 189-198, ACM, 2014. [\[See paper on ACM Digital Library\]](#)
- Klaas Andries de Graaf, Peng Liang, Antony Tang, Willem van Hage, Hans van Vliet - "[An Exploratory Study on Ontology Engineering for Software Architecture Documentation](#)"  - *Computers in Industry*, Vol. 65, nr. 7, pages 1053-1064, Elsevier, 2014. [\[View paper on ScienceDirect\]](#)
- Klaas Andries de Graaf, Antony Tang, Peng Liang, Hans van Vliet - "[Ontology-based Software Architecture Documentation](#)"  - In *Proceedings of the Joint 10th Working IEEE/IFIP Conference on Software Architecture & 6th European Conference on Software Architecture (WICSA/ECSA)*, pages 121-130, IEEE Computer Society, 2012. [\[View paper on IEEE Xplore\]](#)