WikiMap

# Release Notes Alana Killeen, Dylan Horkin, Gordon Liu, Kimberly Koenig, Liem Dinh, Michael Amorozo, Robert Chu, Steven Kwan

## Draft 1.4

## 3/9/2011

## CSRocks, Inc. CSE 403 2011

Table of Contents

[1. Revision History 3](#_Toc287375814)

[2. Introduction 4](#_Toc287375815)

[3. Supported Browsers 4](#_Toc287375816)

[4. Live URL 4](#_Toc287375817)

[5. Required Tools 4](#_Toc287375818)

[6. Feature List 4](#_Toc287375819)

[7. Known Issues 5](#_Toc287375820)

[8. Install/Run Instructions 5](#_Toc287375821)

[8.1. Package Overview 5](#_Toc287375822)

[8.2. Basic Installation 5](#_Toc287375823)

[8.3. Building/Compiling Components 6](#_Toc287375824)

[8.4. Connecting to Your Own Database 6](#_Toc287375825)

[8.5. Generating Your Own Data 7](#_Toc287375826)

[8.6. Setting up the database 7](#_Toc287375827)

[9. Database Schema 8](#_Toc287375828)

[9.1. Database Location & Credentials 8](#_Toc287375829)

[9.2. Accessing the Database 8](#_Toc287375830)

[9.2.1. Using MySQL Workbench 8](#_Toc287375831)

[9.2.2. Running MySQL Queries in PHP 11](#_Toc287375832)

[9.3. Sample Queries 11](#_Toc287375833)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Primary Author(s) | Description | Date |
| 1.0 | Kimberly Koenig | Set up initial document | 2/19/2011 |
| 1.1 | Kimberly Koenig | Added install/run instructions, changelog access instructions | 2/20/2011 |
| 1.2 | Kimberly Koenig | Updated database connection instructions and schema. | 2/22/2011 |
| 1.3 | Kimberly Koenig | Updating for final release. | 3/8/2011 |
| 1.4 | Kimberly Koenig | Updated per team feedback. | 3/9/2011 |

# Introduction

This binary release documentation is for anyone who wants to run their own installation of WikiMap.   
  
We assume that you have a database [set up with the appropriate schema](#_Overview_&_Schema). Explaining the database setup beyond providing a schema and basic instructions is outside the scope of this document.

# Supported Browsers

WikiMap is supported in Chrome 9+, Firefox 3.6+ , and Safari 5+. We recommend using Chrome, Safari, or Firefox 4 for optimal handling of HTML5 and Canvas.

# Live URL

Coming soon…

# Required Tools

For the purposes of running WikiMap we assume you have:

1. **Basic Install –** You use our database and do not want to make any changes to the back-end
   1. Firefox 3.6+, Chrome 9+, or Safari 5+ browser (Chrome recommended)
   2. Linux web server
2. **Custom Install –** You want use your own database and/or want to generate your own relationship data. Requires all of the above, plus:
   1. [Eclipse IDE for Java Developers](http://www.eclipse.org/downloads/moreinfo/java.php)
   2. MySQL database with write access
   3. [MySQL Workbench](http://wb.mysql.com/)
   4. [Ant 1.8 or higher](http://ant.apache.org/)

# Feature List

1. UI
   * 1. Searching for an article
     2. Clicking and dragging the map
     3. Hovering over nodes displays a preview (image and summary text)
     4. Single-clicking on an article node builds a map for it
     5. Swapping between “Full Map” view and “Full Article” view
     6. Clicking on an article link (in “Full Article” view) fetches the article and mini-map for it
     7. Hovering on article links (in “Full Article” view) displays a preview
     8. Clicking on sidebar links (in “Full Map” view) builds a map for it
2. API
   * 1. Relevancy Tree Builder
     2. Relevancy Tree Caching
     3. Image and Summary Tree Caching
3. Logic
   * 1. Relevancy algorithm
     2. Dump downloader
     3. Wikipedia XML Parser

# Known Issues

**Bug database:** <http://code.google.com/p/cse403-wi11-wikimap/issues/list>

**Bug list coming soon…**

# Install/Run Instructions

## Package Overview

1. **Front-End – PHP, HTML, CSS, JavaScript (web server components)**
   1. **UI** – Everything that should be placed on your web server. These are all the front-end components. They do not require compilation due to being JavaScript, PHP, and CSS files.
2. **Back-End – PHP, Java**
   1. **Logic –** Components to parse Wikipedia data, build relationships for the database, and detect updates to the Wikipedia XML dump. Requires compilation.
   2. **Communication –** Components for connecting to and updating the database. Requires compilation.
   3. **DB –** Database setup scripts.

## Basic Installation

This is a basic, out-of-box installation. It connects to our database.

1. Download the distribution to your web server’s **root** directory using wget:  
     
   **wget** [**http://www.kimberlykoenig.com/files/wikimap/wikimap\_binary.zip**](http://www.kimberlykoenig.com/files/wikimap/wikimap_binary.zip)
2. Download our installation script to the same **root** directoryusing wget:  
     
   **wget** [**http://www.kimberlykoenig.com/files/wikimap/wikimap\_install.sh**](http://www.kimberlykoenig.com/files/wikimap/wikimap_install.sh)
3. Change permissions on the shell script to allow execution:  
     
   **chmod +x wikimap\_install.sh**
4. Install WikiMap by running the shell script:  
     
   **./wikimap\_install.sh**

|  |
| --- |
| **Note:** this script assumes that your web server uses a **www** directory to deliver web content. If a **www** folder already exists, WikiMap files and folders will be copied into it. Otherwise, a new **www** directory will be created. If your web server uses a different schema, such as **public\_html**, just rename the **www** directory. |

This installs front-end components to **/www/wikimap\_ui/** and back-end components to a **/wikimap/** directory.

1. WikiMap will now be installed and available at: http://<your\_web\_server>/wikimap\_ui/

|  |
| --- |
| **Security Note:** On your web server, PHP files should be prevented from being directly accessed, otherwise you will expose your database credentials.  We do not provide instructions for doing so because the method may vary, but it typically involves modifying your .htaccess to prevent access to certain filetypes (some tips available here: <http://www.ducea.com/2006/07/21/apache-tips-tricks-deny-access-to-certain-file-types/>) |

## Building/Compiling Components

If any Java components in the back-end are modified, the WikiMap package will need to be recompiled. These instructions will guide you through that process.

1. Navigate to the back-end components in the **wikimap** directory. Ensure that there is a file called **“build.xml”** in the folder.
2. Clean and build the WikiMap package by running our ANT build script:

**ant clean all**

Where targets include: clean, init, all, communication, parser, relbuilder, dumpupdater

## Connecting to Your Own Database

If you want to use your own database, some components will have to be reconfigured. Note that you will have to recompile any Java files that are altered.

1. Coming soon…

## Generating Your Own Data

These instructions will guide you through the process of generating your own relationship data to be inserted into your database. This data is takes the form of a .sql file containing a batch of INSERT statements generated using our parsing algorithm and the Wikipedia XML dump.

|  |
| --- |
| **Note**: This is a very time, processor, and memory-intensive process. We recommend designating a server specifically to perform data generation, or using AWS or similar to run this algorithm and to host your database. |

1. In the same directory as the <build file name>, run the following command to download the Wikipedia XML (in Linux):  
     
   **wget** [**http://download.wikimedia.org/enwiki/latest/enwiki-latest-pages-articles.xml.bz2**](http://download.wikimedia.org/enwiki/latest/enwiki-latest-pages-articles.xml.bz2)
2. Then untar it:

**tar -xjvf enwiki-latest-pages-articles.xml.bz2**

Run the command:  
 **ant clean all**

Where “filename” is the name of the file that you unpacked in the previous step.

1. Coming soon…

## Setting up the database

Database setup is currently outside the scope of this document. We assume that your database has been set up with the appropriate schema, permissions, etc. The details of our database setup are outlined in the [Database access section](#_Database_access).

# Database Schema

Our database is a MySQL server running on AWS. The database has the following schema/tables:  


## Database Location & Credentials

The WikiMap MySQL database is located on AWS and is read-only accessible.

1. Customers/staff may access the database read-only using the **MySQL Database Username** and **Password** we provide:

**Username:** wikiread

**Password:** WikipediaMaps123

**Server:** cse403.cdvko2p8yz0c.us-east-1.rds.amazonaws.com

1. The WikiMaps database table is called **wikimapsDB**.

More detailed access instructions (including using MySQL Workbench and PHP) are provided below.

## Accessing the Database

### **Using MySQL Workbench**

To query the database, view its tables, or otherwise access it from a front-end without having to write PHP code, use MySQL workbench as follows:

1. Download Workbench from this link: <http://www.mysql.com/downloads/workbench/>
2. After installing, click on **Open Connection to Start Querying**

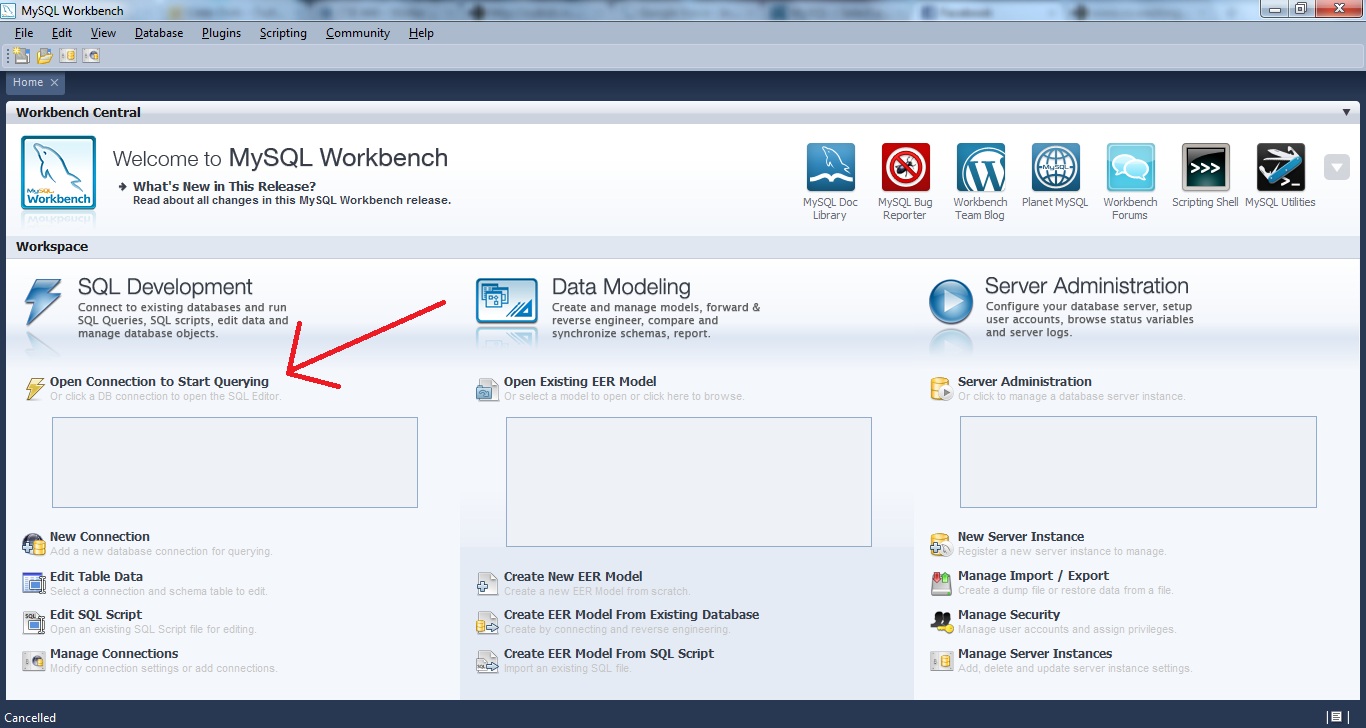
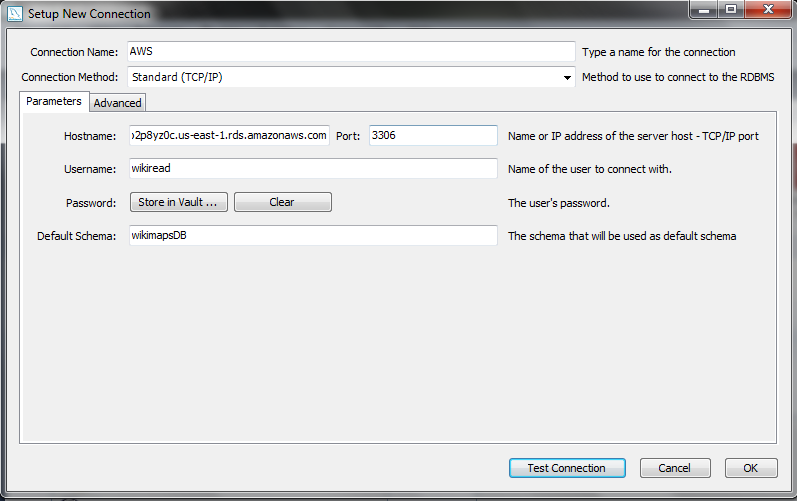


Figure : Open Connection to Start Querying

1. Enter the following settings (see next page for a sample settings screen):
   1. For **Connection Type** select **Standard TCP/IP**
   2. Change the **MySQL Hostname** to *cse403.cdvko2p8yz0c.us-east-1.rds.amazonaws.com*
   3. Change the **MySQL Username** to *wikiread*.
   4. Enter your **MySQL password** by clicking **Store in Vault**, entering the password *WikipediaMaps123*, and saving it.

 Figure : Sample settings

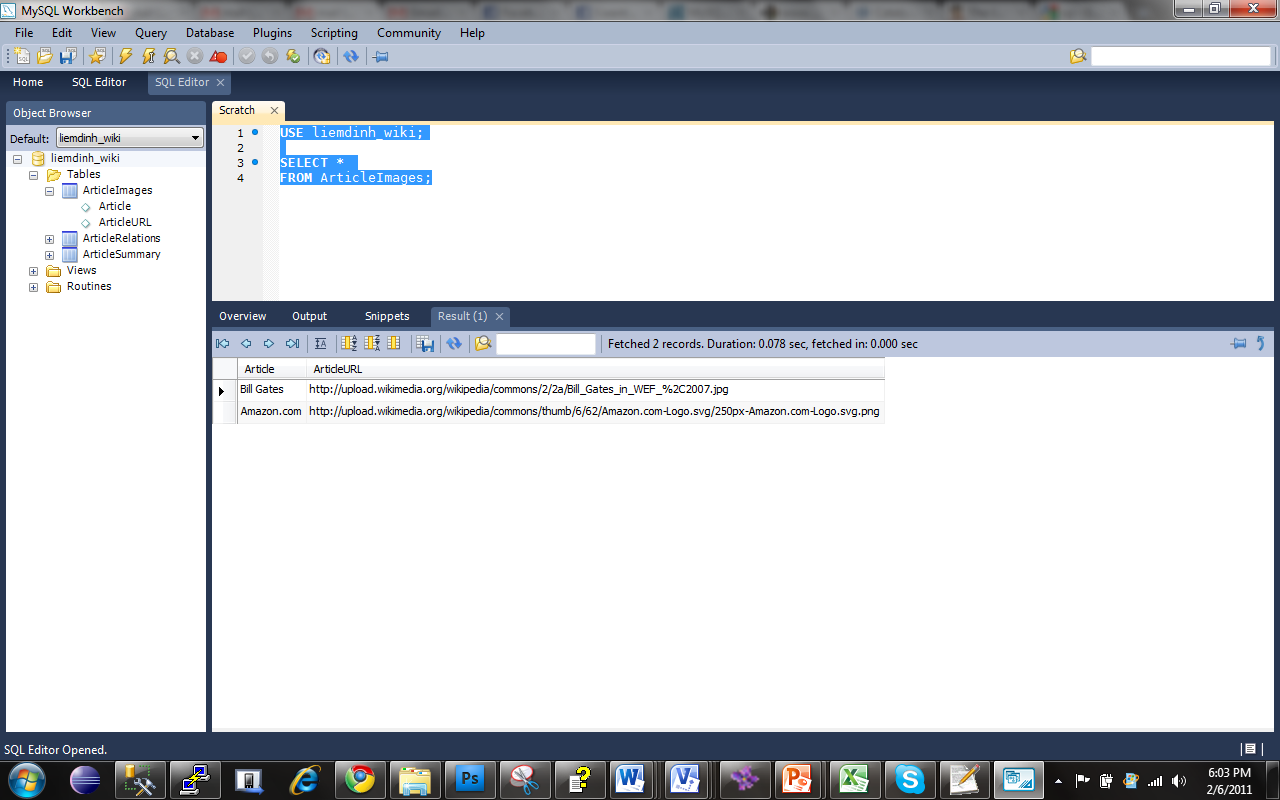
1. Now you can run a new query from the **Scratch** area. See [Sample Queries](#_Sample_Queries) for some example queries that can be run on our database. 

Figure : “Scratch” Area

### Running MySQL Queries in PHP

Here is a sample MySQL PHP query that queries the “wikimapsDB” database, retrieves all the data from the “ArticleImages” table, and prints out each row.

**<?php**

**// Make a MySQL Connection  
mysql\_connect("**cse403.cdvko2p8yz0c.us-east-1.rds.amazonaws.com**", "wikiread", "WikipediaMaps123*"*) or die(mysql\_error());  
mysql\_select\_db("wikimapsDB") or die(mysql\_error());**

**// Retrieve all the data from the "ArticleImages" table  
$result = mysql\_query("SELECT \* FROM ArticleImages") or die(mysql\_error());  
while($row = mysql\_fetch\_array( $result )) {  
// Print out each row**

**echo $row['Article'];  
echo " ";   
echo $row['ArticleURL']."</br>";}**

**?>**

## Sample Queries

1. Select all rows from the “ArticleImages” table:

**USE wikimapsDB;  
SELECT \*   
FROM ArticleImages;**

1. Select all rows from the “ArticleSummary” table:

**USE wikimapsDB;  
SELECT \*   
FROM ArticleSummary;**

1. Select all rows from the “ArticleRelations” table:

**USE wikimapsDB;**

**SELECT \***

**FROM ArticleRelations;**

1. Select the Article Summary for “Bill Gates”:

**USE wikimapsDB;**

**SELECT s.Summary**

**FROM ArticleSummary s**

**WHERE s.Article = 'Bill Gates';**

1. Return the summaries for all articles related to “Amazon.com”:

**USE wikimapsDB;**

**SELECT s.Article, s.Summary**

**FROM ArticleRelations r, ArticleSummary s**

**WHERE r.Article = 'Amazon.com' AND r.RelatedArticle = s.Article;**