Installing Visual Query on Linux

Dependencies

Visual Query has five dependencies:

1) Ruby (programming language)
2) RubyGems
3) Rails (does database access, etc)
4) DBMS (MySQL, Oracle, etc – we use MySQL)
5) Required gems and plugins

They need to be installed in this order, although 4) and 5) can be installed in either order.

The exact steps necessary to install any package on Linux depends on the distribution, so some of this will be distribution-dependent. If the PackageManager on your Linux Distribution has the ability to install Ruby, RubyGems, Rails, etc then by all means, use that to install it. Unfortunately, not all distributions include Ruby and Rails in their packages, or may only include an older version.

Ruby

It is very possible that Ruby is already installed on your Linux system. Run the command "which ruby" to check is one is installed and then "ruby –v" to check on the version. If version 1.8.6 or similar is installed, then go to the next step.

Note: *You should install Ruby 1.8.6, not Ruby 1.9*. Ruby 1.9 has some problems with Rails 2.3.

Good instructions for installing ruby are at *http://ruby.about.com/od/tutorials/ht/installrubylin.htm*

RubyGems

RubyGems is the equivalent of the PackageManager for programs that run under Ruby.

Good instructions for installing RubyGems are at *http://docs.rubygems.org/read/chapter/3*

Rails

Once RubyGems is installed installing Rails is simple. You can use RubyGems to install Rails:

sudo gem install –v=2.3.6 rails

You should install Ruby on Rails 2.3.2 or above, but NOT Rails 3.0.x. The latest version of Rails 2 is 2.3.6 but we use 2.3.2 internally. "**gem install -v=2.3.6 rails**" will force version 2.3.6. If you leave that off you will get Version 3.0.3 and Visual Query is not currently compatible with Rails 3.

Database

After rails is set up you will need to installer your database if you don't already have it. Do a search on MySQL if you want to use that, or follow the DB instructions for the DBMS that you want to use. You will need to create a user account for Rails to use.

Once you have VisualQuery set up you will need to modify the "database.yml" file in the config directory of the rails installation to define the proper DB and user information. The changes will look something like:

> **username: cadviewer**
> **password: olympics**
> **host: mysql.vizquery.com**

Testing

Once you have everything set up go to the directory where you want to put the rails project and type "rails visquery" (or whatever you want to name it). This will create a dummy rails project. If you want to use your web server, then you will need to point your server to the "visquery/public" directory as the root directory for the website on port 3000 (by default). If you want a "quick and dirty" webserver, just go to the visquery directory and type "ruby script/server" and that will start the Mongrel webserver that comes with rails.

You should now go to the webpage, http://localhost:3000" if you use Mongrel. It should display the Ruby on Rails "Welcome Aboard" message. If you then click the "About Your Applications Environment" button you will get a "We're sorry, but something went wrong." message. Don't worry, that is correct at this point because nothing will have been installed yet.

Copy Visual Query

You should now copy the supplied version of Visual Query over the sample rails project you created above. This will install Visual Query.

Required Gems

Visual Query requires several gems to run correctly. Thankfully these gems will be installed in one step using the following command:

rake gems:install

To see a list of required gems prior to installation run:

rake gems

They can be installed manually using:

prawn	sudo gem install prawn

Required Plugins

Visual Query requires several plugins to run correctly. These plugins will be delivered already installed in the public/vendor/plugins directory. If they need to be reinstalled for any reason the current plugins can be found at:

auto_complete	https://github.com/rails/auto_complete.git
prawn_to	git://github.com/thorny-sun/prawnto.git
paperclip	https://github.com/thoughtbot/paperclip.git

Oracle Adapter

JPMC uses Oracle as the database management system. For Rails to work with a given database system the Database Adapter for ActiveRecord must be installed and configured. For Oracle this consists of two parts: the Ruby-OCI8 gem and the ActiveRecord Oracle Enhanced Adapter gem. Thankfully these are easy to install:

gem install ruby-oci8
gem install activerecord-oracle_enhanced-adapter

There is a good article on this subject at *http://www.skorks.com/2010/02/using-ruby-on-rails-with-oracle-and-deploying-it-all-to-tomcat/*

More Testing

It is now time to test Visual Query. You can start Visual Query by accessing proper page under your web server. If you are using localhost as your webserver (with either the web server supplied with Rails, "Webrick", or a normal web server such as Apache, you should configure the "public" directory as the home directory for port 3000 on the web server. You would then start Visual Query like http://localhost:3000/rooms. This will start the Rooms controller under Visual Query. You should see the default drawing on the left side and list of rooms on the right side of the screen.

If you see "Trial Only" over the drawing that means the CadViewer license is not installed properly. Please open the Java Console (accessed from the "Tools" menu on most web browsers) and send the output to Tailor Made Software so we can examine the URL details and correct any problems.