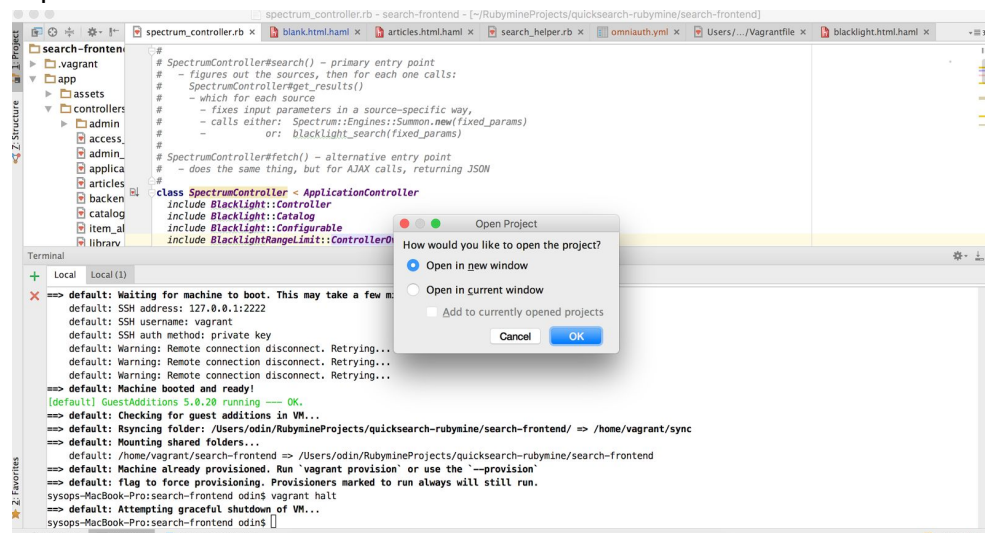# Steps for Quicksearch RubyMine (RM) Vagrant:

- Vagrant commands to learn:
    - ○ `vagrant up (to boot up the instance)`
    - ○ `vagrant reload --provision (to reload the configuration)`
    - ○ `vagrant ssh (to log in)`
    - ○ `vagrant snapshot push (to save an instance in time)`
    - ○ `vagrant global-status (to view status of all instances)`
    - ○ `vagrant status (to show status)`
    - ○ `vagrant halt (to stop Vagrant)`
    - ○ `vagrant destroy id (to remove vagrant instance)`
    - ○ `vagrant plugin install vagrant-vbguest`
- Install (1) VirtualBox; (2) Vagrant; (3) RubyMine.
    - ○ Run "`vagrant -v`" in Terminal to verify that vagrant is working.
    - ○ Install vbguest plugin:
        - ■ `vagrant plugin install vagrant-vbguest`
- Make sure that you have set up your GitHub key. The steps are:
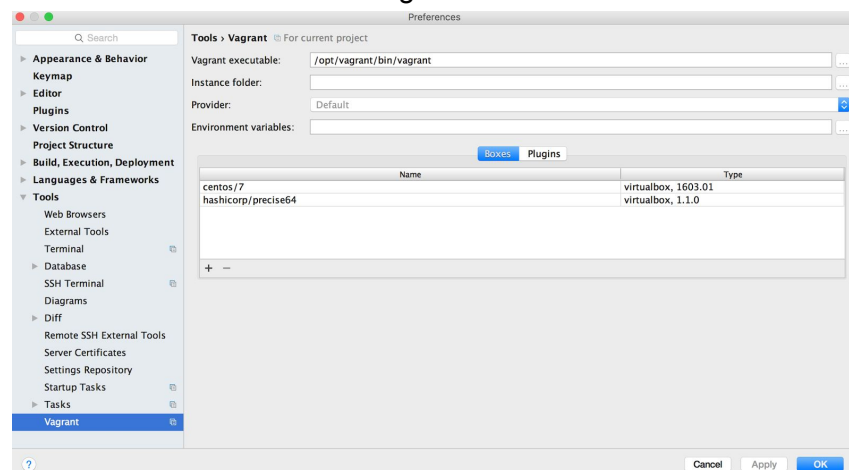
    ```
    ssh-keygen -t rsa -b 4096 -C you@email.com

    # press enter to save key to your home folder

    eval "$(ssh-agent -s)"

    # it should print "Agent pid . . ."

    ssh-add ~/.ssh/id_rsa

    # it should print "Identity added . . ."

    pbcopy < ~/.ssh/id_rsa.pub

    # now paste this key into your GitHub SSH Settings page

    ssh -T git@github.com

    # enter "yes" to recognize host
    ```

- Export your key to the folder containing the repo (and verify with `ssh-add -L`). (The steps are shown in detail in the quicksearch-vagrant repo wiki.)
    - ○ `key=~/.ssh/id_rsa`

        `[[ -z $(ssh-add -L | grep $key) ]] && ssh-add $key`
- Start RubyMine.
    - ○ Git clone in RubyMine (third option in the launcher) all the QS projects:
        - ■ `search-frontend`
        - ■ `search-backend`

- ○ For the 2nd project (as shown below) you're adding, hit the checkbox, and select "Open in current window."
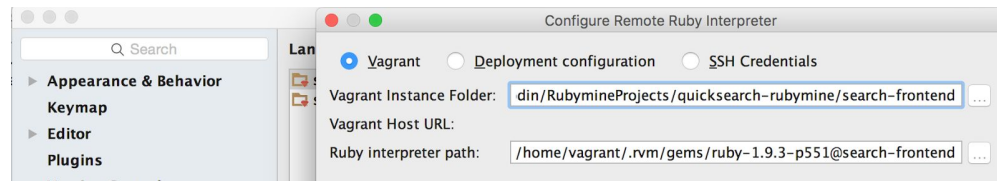

- ○ Open Terminal in RubyMine (View/Tool Windows/ Terminal)

- Run "vagrant up"
  - ○ This will install the stack. (It will take a while.)
  - ○ Install vbguest plugin (just in case you haven't already and if it complains).
  - ○ Run "`vagrant reload --provision`" if you ran into an error and had to fix something.
- Configure Vagrant and Ruby connections in RubyMine.
  - ○ Specify the path for "Vagrant executable: and verify that box "centos/7" shows up in RM/ Preferences/ Tools/ Vagrant.
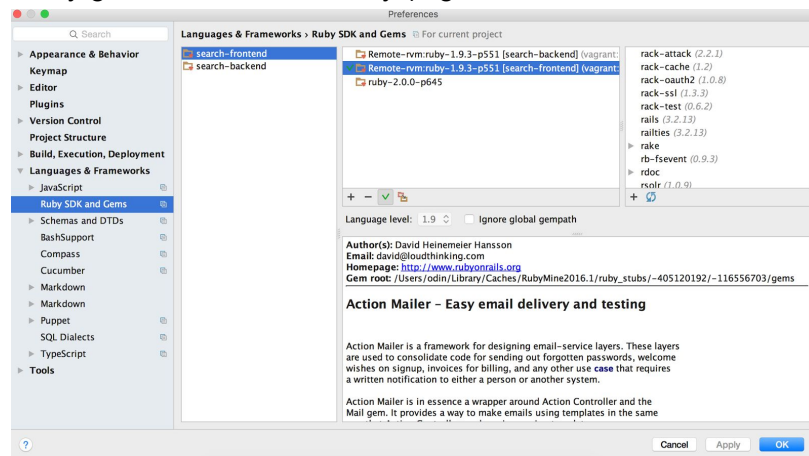

  - ○ Go to RM/Languages & Frameworks/Ruby SDKs & Gems
  - ○ Click on + icon to add remote SDK. Select Vagrant in the dialog box (as shown below). Make sure the path is to search-frontend in the first box, and the last path points to gem locations (the same location as the path vagrant loading says

"Happy Coding!")

- ○ Ensure that the correct remote appears with a green checkmark next to it.
- ○ Verify gem/bundler availability (e.g., Rails 4 for SB; Rails 3 for SF):

- Log in ("`vagrant ssh`" into) each box through RM Terminal. If the search-frontend gemset is not selected as default by rvm, select it.
  - ○ `cd search-frontend; rvm gemset list; …`
  - ○ Same for SB.
- Verify database and tables exist for both SF and SB. For example, just for SF:
  - ○ `cd quicksearch-vagrant/db; sqllite3 development.sqllite3`
  - ○ `rake db:migrate` (in case the database is missing. A sign is getting Voyager error from search-backend.)
- Start Solr on SF box (if it's not running):
  - ○ Inside the blacklight-jetty-4.10.4 folder do: `java -jar start.jar &`
  - ○ Verify by running `ps -A | grep java`.
- Ensure `/bin` folder exists in both search-frontend and search-backend. Otherwise, RubyMine will be unable to run the project.
- For Debug, RubyMine might prompt for installation of debugging gems. This might change in future.
- Select RM/Run to run both programs (or select each project and click on Run -- it should automatically pick up the project runtime info, but if it doesn't click on Edit Configuration). The Edit Configuration should look like the following.
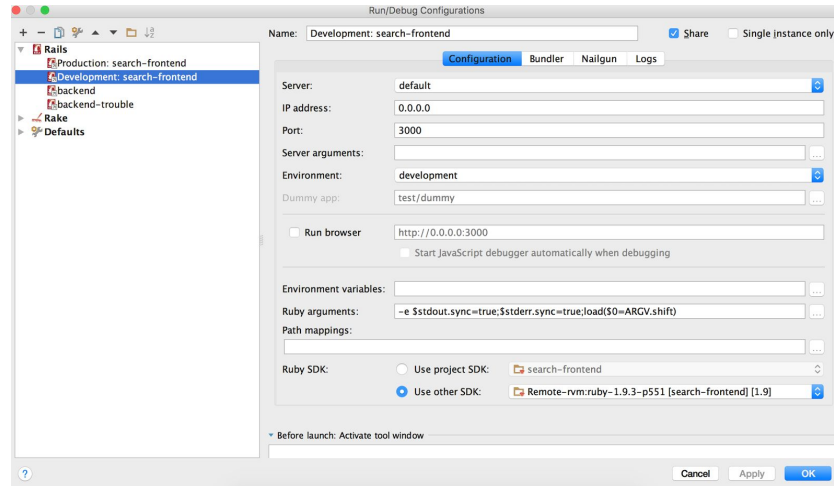 - ○ *Note the different IP and port for search-backend*. See Figure 5.
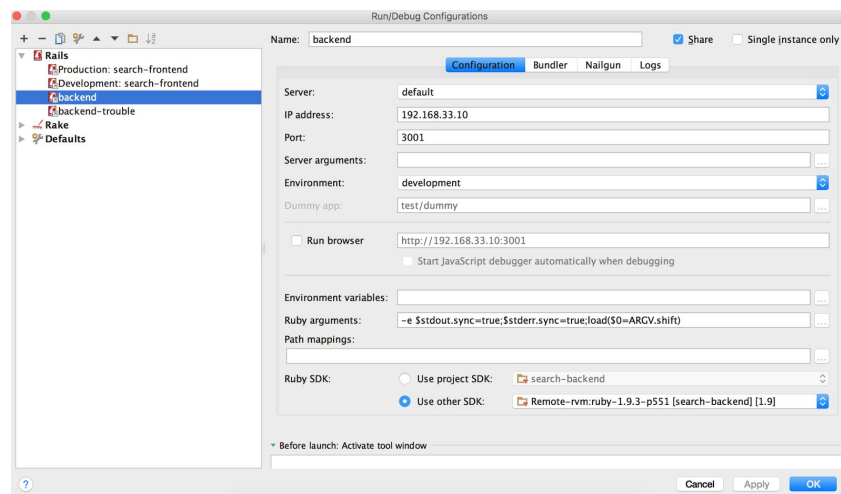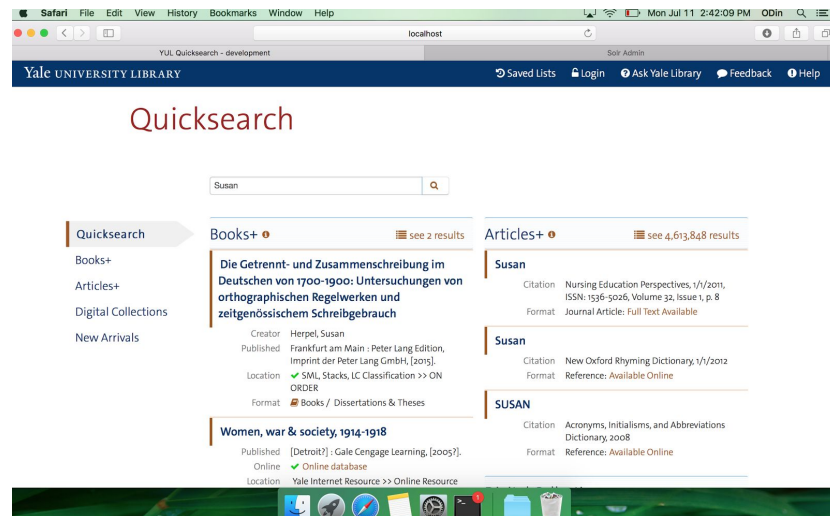
Fig 4: SF Run/Debug Configurations

Fig 5: SB Run/Debug Configurations

- Finally, browse to
  - `localhost:3000` for search-frontend
  - `localhost:8983` for Solr

RubyMine Tips

- View/Tools Windows/Terminal to view Terminal
- Command-4 to open Run Console
- Command-T to open a new Terminal window

Troubleshooting

- Each project needs a `bin` folder. When running a project, if `Run Configurations` keeps pointing to the wrong folder (e.g., search-frontend for search-backend), there's a trick to fool RubyMine: point to a local SDK, click Apply/Close, and then point to the remote SDK. This might happen if you accidentally deleted the `bin` directory.
- Sometimes Vagrant can keep displaying "remote connection error". Do a `vagrant halt` and then a `vagrant up`.
- Sometimes RM might have problem connecting to the box. Try again.

Consolidated VM Instructions:

This VM contains the full stack.

- Git clone:
  - https://github.com/yalelibrary/quicksearch-vagrant
- Follow the steps in the repo wiki.