

# NetBeans

A Ruby IDE written in <cough>Java<cough>

and why the entire Ruby/Java thing is important.

# But who is he?

- Dj Walker-Morgan, JAFID\*
- Used Java since Java 1.0
- Currently using Ruby on Rails and Java

[www.codepope.com](http://www.codepope.com)

\* Geeky reference may contain Blue Thunder

# NetBeans for Java



- Java IDE
- Built in standard Java
- Open Sourced (GPL2 soon)
- The IDE that Sun sponsor
- Also reusable as an application platform
- Chock full o'functionality

# NetBeans is chock full o'functionality

- Version control (Subversion, CVS, others)
- Refactoring
- J2EE, JSP/JSF, Glassfish, JavaDB, UML
- Matisse Swing GUI Builder
- Profiling
- J2ME development tools

# Ruby - Java's best friend?

- Ruby is compelling
- Rails has kicked off a trend for Java framework developers to rethink their approaches....
- Why not skip that and go straight to the source?

# Java - Ruby's best friend?

- jRuby
  - Gateway to all the enterprise libraries
  - Deployability onto Java application servers
- Ruby->JVM byte code compiler
  - Offers big performance gains
  - Coming soon (<http://headius.blogspot.com/2007/09/compiler-is-complete.html>)

# NetBeans for Ruby and RoR

- Officially arrives as part of NetBeans 6
- Available now in beta (and daily) builds and standalone Ruby version

# For Ruby Applications

- Full editor support
  - Code completion with documentation
  - Refactoring
  - Shortcuts

# NetBeans knows Ruby

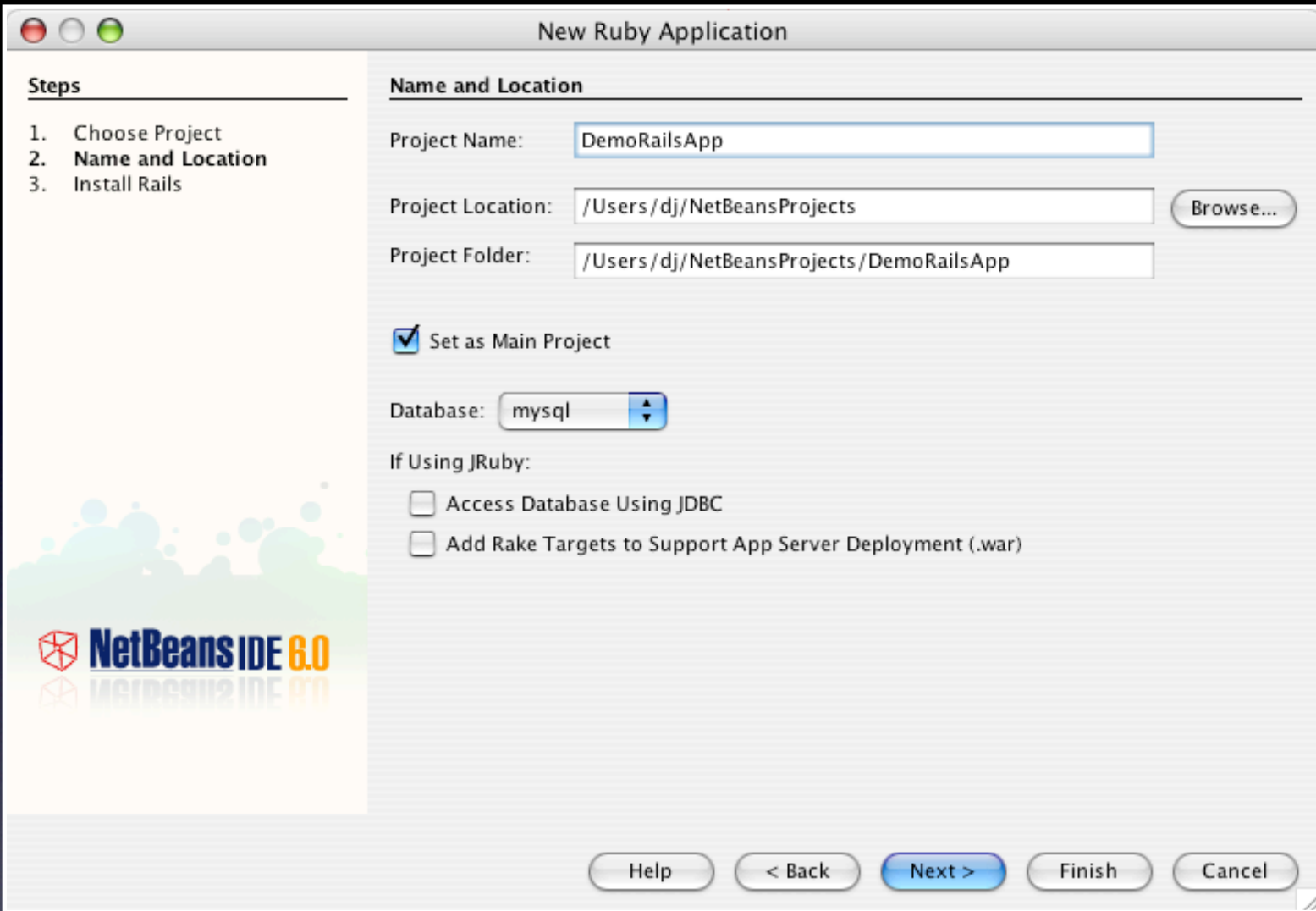
- Debugging
  - Classic or fast debugger
- Manages RubyGems
- Can use jRuby or native Ruby

# For Ruby On Rails

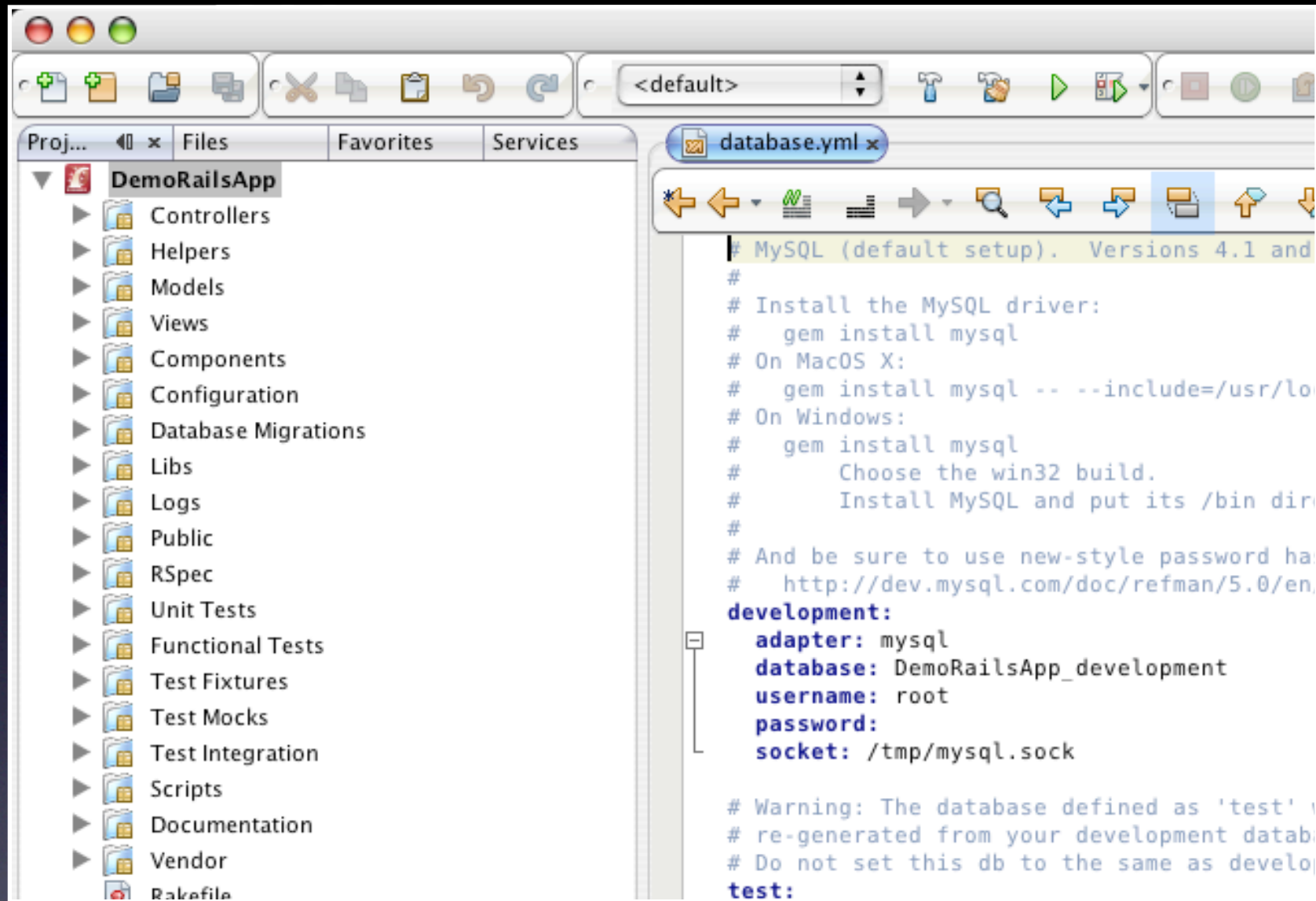
- Project creation
- Database migration
- RHTML editing with code completion

# NetBeans knows RoR

- Integrates with the underlying command line tools including Rake
- Shows an organised project view
- Knows about the relations of controllers and views, controllers/models and tests



Let's Make A Rails App...



# Taa...Daa....

TODO: Add Glamourous Assistant



# Let's Make A Model

Generate: scaffold Install Generators...

Model Name: Demonstration

Controller Name: Event

Actions:

When file already exists:  Skip  Overwrite

Preview Changes Only

Description:

The scaffold generator creates a controller to interact with a model. If the model does not exist, it creates the model as well. The generated code is equivalent to the "scaffold :model" declaration, making it easy to migrate when you wish to customize your controller and views.

The generator takes a model name, an optional controller name, and a list of views as arguments. Scaffolded actions and views are created automatically. Any views left over generate empty stubs.

The scaffolded actions and views are:  
index, list, show, new, create, edit, update, destroy

If a controller name is not given, the plural form of the model name

# Lets Make a Controller

```
def index
  redirect_to :action=>:list
end
```

●	redirect_back_or_default(default)	Authent>
●	redirect_to(options, *parameters_for_meth...	ActionC>
●	redirect_to_path(path)	ActionControlle>
●	redirect_to_url(url, permanently)	ActionC>

```
redirect_to :action => "index"          rea
redirect_to :action => "show", :id => @item  reai
redirect_to :controller => "items"         rec
redirect_to :controller => "items", :action =>... reca
redirect_to :controller => "items", :action =>... reca
```

---

**Base.redirect\_to(options, \*parameters\_for\_method\_reference)**

Redirects the browser to the target specified in options. This parameter can take one of three forms:

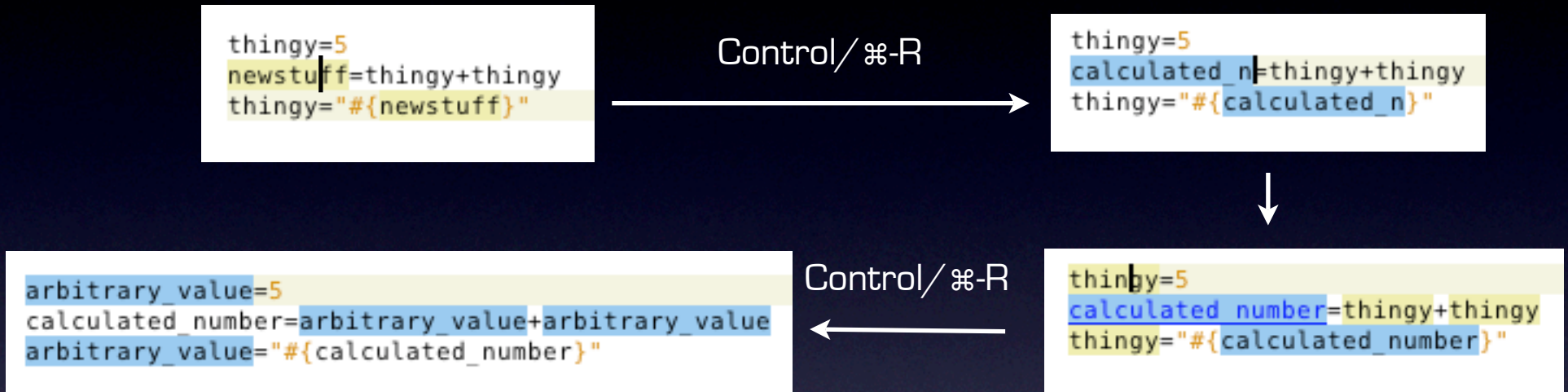
- Hash: The URL will be generated by calling url\_for with the options.
- String starting with protocol:// (like <http://>): Is passed straight through as the target for redirection.
- String not containing a protocol: The current protocol and host is prepended to the string.
- :back: Back to the page that issued the request. Useful for forms that are triggered from multiple places. Short-hand for redirect\_to(request.env["HTTP\_REFERER"])

Examples:

```
redirect_to :action => "show", :id => 5
redirect_to "http://www.rubyonrails.org"
redirect_to "/images/screenshot.jpg"
```

# Code completion and live documentation

# Renamfactoring



- Inline renaming
  - A quicker way to tidy up localised code
  - Switches to general refactoring if not local
  - with a **very very very important** preview

# Navigation on Rails

- Context sensitive
  - From View to Action
  - From Controller or Model to Test
  - From partial reference to partial
  - Through any symbol
    - with documentation preview

# Wrapping Up

- NetBeans rocks with Java
- Rocks even more with Ruby on Rails
- It's cheaper than TextMate :-)
- It runs on Windows too
  - And Linux and Solaris and....
- <http://www.netbeans.org/>