Ruby on Rails

From Wikipedia, the free encyclopedia

Ruby on Rails, often shortened to **Rails** or **RoR**, is an open source web application framework for the Ruby programming language. It is intended to be used with an Agile development methodology that is used by web developers for rapid development.[2]

Ruby on Rails

Developer(s)	Rails Core Team 🔗
Stable release	2.3.8[1]
	/ May 25, 2010; 43 days ago
Preview release	3.0.0 (Beta 4)
	/ June 8, 2010; 29 days ago
Development status	Active
Written in	Ruby
Operating system	Cross-platform
Type	Web application framework
License	MIT License
Website	http://rubyonrails.com/ 🔗

Contents   [hide]

History   [edit]

Version history

Version ⊠	Date ⊠
1.0[3]	December 13, 2005
1.2[4]	January 19, 2007
2.0[5]	December 7, 2007
2.1[6]	June 1, 2008
2.2[7]	November 21, 2008
2.3[8]	March 16, 2009

Ruby on Rails was extracted by David Heinemeier Hansson from his work on Basecamp, a project management tool by 37signals (now a web application company).[9] David Hansson first released Rails as open source in July 2004, but did not share commit rights to the project until February 2005.[10] In August 2006 the framework reached a milestone when Apple announced that it would ship Ruby on Rails with Mac OS X v10.5 "Leopard",[11] which was released in October 2007.

Rails version 2.3 was released on March 15, 2009. Major new developments in Rails includes templates, engines, Rack and nested model forms.

- Templates enable the developer to generate a skeleton application with custom gems and configurations.
- Engines let one reuse application pieces complete with routes, view paths and models.
- The Rack web server interface and Metal allow one to write optimized pieces of code that route around ActionController.[12]

On December 23, 2008, Merb, another web application framework was launched, and Rails announced a commitment to work together. The Rails team announced they would work with the Merb project to bring "the best ideas of Merb" into Rails 3, ending the "unnecessary duplication" across both communities.[13]

Technical overview   [edit]

Like many web frameworks, Rails uses the Model-View-Controller (MVC) architecture pattern to organize application programming.[14]

Ruby on Rails includes tools that make common development tasks easier "out of the box", such as scaffolding that can automatically construct some of the models and views needed for a basic website.[15] Also included are WEBrick, a simple ruby web server, and Rake, a build system. Together with Rails these tools provide a basic development environment.

Ruby on Rails relies on a web server to run it. Mongrel was generally preferred over WEBrick at the time of writing[citation needed] but it can also be run by Lighttpd, Abyss, Apache (either as a module - Passenger for example - or via CGI, FastCGI or mod_ruby), and many others. From 2008 onwards, the Passenger web server replaced Mongrel as the most used web server.[citation needed] Recently, the Unicorn web server has become a favorite of new deployments.[citation needed]

Rails is also noteworthy for its extensive use of the JavaScript libraries Prototype and Script.aculo.us for Ajax.[16] Rails initially utilized lightweight SOAP for web services; this was later replaced by RESTful web services.

Since version 2.0, Ruby on Rails by default offers both HTML and XML as output formats. The latter is the facility for RESTful web services.

Ruby on Rails relies on Ruby 1.8.6. The upcoming 3.0 version will drop support for Ruby 1.8.6 and require Ruby 1.8.7 to work[17].

Framework structure [edit]

Ruby on Rails is separated into various packages, namely ActiveRecord (an object-relational mapping system for database access), ActiveResource (provides web services), ActionPack, ActiveSupport and ActionMailer. Prior to version 2.0, Rails also included the Action Web Service package that is now replaced by Active Resource. Apart from standard packages, developers can make plugins to extend existing packages.

Deployment [edit]

Ruby on Rails is often installed using RubyGems, a package manager[18] which is included with Ruby. Many Linux distributions also support installation of Rails and its dependencies through their native package management system.

Ruby on Rails is typically integrated with a database server such as MySQL and a web server such as Apache.

Philosophy and design [edit]

Ruby on Rails is intended to emphasize *Convention over Configuration* (CoC), and the rapid development principle of *Don't Repeat Yourself* (DRY).

"Convention over Configuration" means a developer only needs to specify unconventional aspects of the application. For example, if there is a class *Sale* in the model, the corresponding table in the database is called *sales* by default. It is only if one deviates from this convention, such as calling the table "products sold", that the developer needs to write code regarding these names. Generally, this leads to less code and less repetition.

"Don't repeat yourself" means that information is located in a single, unambiguous place. For example, using the ActiveRecord module of Rails, the developer does not need to specify database column names in class definitions. Instead, Ruby on Rails can retrieve this information from the database based on the class name.

Trademarks [edit]

In March 2007 David Heinemeier Hansson filed three Rails related trademark applications to the USPTO. These applications regard the phrase "RUBY ON RAILS",[19] the word "RAILS"[20] and the official Rails logo.[21] As a consequence, in the summer of 2007 Hansson denied to Apress the permission to use the Rails logo on the cover of a new Rails book written by some authoritative community members. The episode gave rise to a polite protest in the Rails community.[22][23] In response to this criticism, Hansson made the following claims:[22] *I only grant promotional use [of the Rails logo] for products I'm directly involved with. Such as books that I've been part of the*

development process for or conferences where I have a say in the execution. I would most definitely seek to enforce all the trademarks of Rails.

Reception [edit]

Rails has been criticized for issues with scalability.[24] These critics often mentioned various Twitter outages in 2007 and 2008, which spurred Twitter's partial transition to Scala (which runs on the Java Virtual Machine) for their queuing system and other middleware.[25][26] The user interface aspects of the site continue to run Ruby on Rails.[27]

See also [edit]

- Comparison of web application frameworks
- *Agile Web Development with Rails*
- Dave Thomas

Free software portal

References [edit]

1. ^ "Ruby on Rails 2.3.8 Released" 🗗. Retrieved 25 May 2010.
2. ^ Rønn Jensen, Jesper (April 12, 2006). "Ruby on Rails as rapid prototyping tool" 🗗. *justaddwater.dk*. Denmark. Retrieved February 2, 2010.
3. ^ "Rails 1.0: Party like it's one oh oh!" 🗗. Riding Rails. Retrieved June 9, 2010.
4. ^ "Rails 1.2: REST admiration, HTTP lovefest, and UTF-8 celebrations" 🗗. Riding Rails. Retrieved June 9, 2010.
5. ^ "Rails 2.0: It's done!" 🗗. Riding Rails. Retrieved June 9, 2010.
6. ^ Rails 2.1: Time zones, dirty, caching, gem dependencies, caching, etc 🗗. Riding Rails. Retrieved June 9, 2010.
7. ^ "Rails 2.2: i18n, HTTP validators, thread safety, JRuby/1.9 compatibility, docs" 🗗. Riding Rails. Retrieved June 9, 2010.
8. ^ "Rails 2.3: Templates, Engines, Rack, Metal, much more!" 🗗. Riding Rails. Retrieved June 9, 2010.
9. ^ Grimmer, Lenz (February 2006). "Interview with David Heinemeier Hansson from Ruby on Rails" 🗗. MySQL AB. Retrieved 2008-06-08.
10. ^ "37 Signals, Rails core team profiles." 🗗. Retrieved 2008-07-15.
11. ^ Hansson, David (August 7, 2006). "Ruby on Rails will ship with OS X 10.5 (Leopard)" 🗗. Retrieved 2008-06-08.
12. ^ Hansson, David (March 16, 2009). "Rails 2.3: Templates, Engines, Rack, Metal, much more!" 🗗.
13. ^ "The day Merb joined Rails" 🗗. December 27, 2008.
14. ^ Ruby on Rails - Rails' MVC architecture 🗗
15. ^ There were quite a few changes in the 2.0 release, including the way that Rails generates scaffolding code. 🗗
16. ^ Rails includes the Prototype JavaScript framework and the Scriptaculous JavaScript controls and visual effects library. 🗗
17. ^ "Rails 3.0: Release Notes" 🗗. rubyonrails.org. Retrieved 2010-03-06. "*Rails 3.0 requires Ruby 1.8.7 or higher. Support for all of the previous Ruby versions has been dropped officially and you should upgrade as early as possible*"
18. ^ "Ruby on Rails: Download" 🗗. RubyonRails.org.
19. ^ ""Ruby on Rails" Trademark Status" 🗗. USPTO. Retrieved 2007-08-01.
20. ^ ""Rails" Trademark Status" 🗗. USPTO. Retrieved 2007-08-01.
21. ^ "Rails Logo Trademark Status" 🗗. USPTO. Retrieved 2007-08-01.
22. ^ *a* *b* Forde, Pete (2007-07-23). "Beginning Rails: From Novice to Professional" 🗗. Retrieved 2007-08-01.
23. ^ Cooper, Peter (2007-07-24). "David Heinemeier Hansson says No to Use of Rails Logo" 🗗. Retrieved 2007-08-01.
24. ^ "5 Question Interview with Twitter Developer Alex Payne" 🗗. radicalbehavior.com. 2007-03-29. Retrieved 2009-07-18.
25. ^ Steve Jenson, Alex Payne, and Robey Pointer interview (2009-04-03). "Twitter on Scala" 🗗. artima.com. Retrieved 2009-07-18. "*We had a Ruby-based queuing system that we used for communicating between the Rails front ends and the daemons, and we ended up replacing that with one written in Scala. The Ruby one actually worked pretty decently in a normal steady state, but the startup time and the crash behavior were undesirable.*"
26. ^ "Twitter jilts Ruby for Scala" 🗗. theregister.co.uk. 2009-04-01. Retrieved 2009-07-18. "*By the end of this year, Payne said, Twitter hopes to have its entire middleware infrastructure and its APIs ported to*

the new language. Ruby will remain, but only on the front end. "We're still happy with Rails for building user facing features... performance-wise, it's fine for people clicking around web pages. It's the heavy lifting, asynchronous processing type of stuff that we've moved away from."'

27. ^ ryan king (2009-09-25). "Twitter on Ruby" 🔗. evan weaver. Retrieved 2009-09-29. "*We use Scala for a few things at Twitter, but the majority of the site is Ruby.*"

Bibliography [edit]

- Ruby, Sam; Thomas, Dave; Hansson, David (March 28, 2009), *Agile Web Development with Rails* (Third ed.), Pragmatic Bookshelf, pp. 850, ISBN 1934356166
- Laurent, Simon St.; Dumbill, Edd (November 28, 2008), *Learning Rails* 🔗 (First ed.), O'Reilly Media, pp. 442, ISBN 0596518773
- Lenz, Patrick (May 1, 2008), *Simply Rails 2* 🔗 (Second ed.), SitePoint, pp. 450, ISBN 0980455200
- Tate, Bruce; Hibbs, Curt (August 22, 2006), *Ruby on Rails: Up and Running* 🔗 (First ed.), O'Reilly Media, pp. 182, ISBN 0596101325
- Holzner Ph.D., Steve (November 29, 2006), *Beginning Ruby on Rails* 🔗 (First ed.), Wrox, pp. 380, ISBN 0470069155
- Allan Hardy, Jeffrey; Carneiro Jr, Cloves; Catlin, Hampton (July 20, 2007), *Beginning Ruby on Rails E-Commerce: From Novice to Professional* 🔗 (First ed.), Wrox, pp. 361, ISBN 1590596862
- Clark, Mike (May 15, 2008), *Advanced Rails Recipes* 🔗 (First ed.), Pragmatic Bookshelf, pp. 464, ISBN 0978739221

External links [edit]

- Ruby on Rails official website 🔗, rubyonrails.com
- RubyForge.org 🔗
- Railscasts.com 🔗, These screencasts are short and focus on Ruby on Rails technique

> Wikibooks has a book on the topic of
> **Ruby on Rails**

v · d · e	**Ruby programming language**	[show]

v · d · e	**Web Application Frameworks**	[show]

Categories: 2004 introductions | Free development toolkits and libraries | Free software programmed in Ruby | Web 2.0 | Web application frameworks | Software using the MIT license