A history of Ruby

Ruby is an object-oriented, interpreted programming language. Interpreted programming languages are read line by line instead of by compiling the code into an executable that is unreadable to a human being (but is much quicker to process by a computer.) Other interpreted languages include JavaScript and BASIC. If you open up your web browser on a page that uses JavaScript, you can read the source code by using your browser's window. Try opening up an application like your system's Calculator in a text editor. All you see is some garbled text because the Calculator application was written in a compiled language.

As Figure 1-1 shows, Ruby was developed in 1993 by Yukihiro Matsumoto, but first released to the public in 1995. Matsumoto designed Ruby primarily to reduce the workload of developers by following the principle of least surprise, meaning that the language typically behaves as the programmer expects: methods are named using common English terms that appropriately define the action being performed. For example, Ruby has actions called strip, split, delete, and upcase to perform actions on strings of text. Each of those names intuitively explains the action they perform.

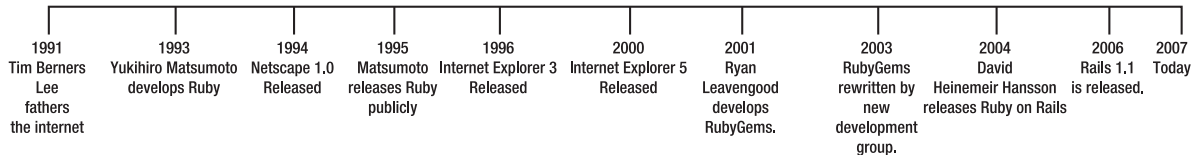| 1991	1993	1994	1995	1996	2000	2001	2003	2004	2006	2007
Tim Berners Lee fathers the internet	Yukihiro Matsumoto develops Ruby	Netscape 1.0 Released	Matsumoto releases Ruby publicly	Internet Explorer 3 Released	Internet Explorer 5 Released	Ryan Leavengood develops RubyGems.	RubyGems rewritten by new development group.	David Heinemeir Hansson releases Ruby on Rails	Rails 1.1 is released.	Today

Figure 1-1. The Ruby timeline

Ruby started to gain popularity in 2001 with the commencement of Ryan Leavengood's work on RubyGems, which is an easy way to package and distribute applications and libraries. RubyGems' development stalled for several years because Leavengood left the project at version 0.4.0. In 2003, a group of developers reincarnated the RubyGems project and released a totally rewritten version under the same name. While they didn't share any of the same code, they shared the same principle: simple software distribution for Ruby.