

A Business View of Ruby on Rails

You Will Hear About This

If your business uses Web-based software (also known as webware), you will hear about Ruby on Rails. This newsletter will answer three questions:

- What is Ruby on Rails?
- Why do you care?
- How can you benefit?

What is Ruby on Rails?

Ruby on Rails (“Rails” or “RoR” for short) helps developers write better webware in less time. Clients appreciate results, and Rails enables developers to deliver in a big way.

RoR consists of two components: an object oriented programming language called Ruby, and a Web development framework called Rails.

Ruby: The Language

Ruby is the object oriented language dominant in Rails applications. Many developers believe that software written in object oriented languages is easier to maintain.

Your company can benefit from the object oriented approach because easier maintenance means a longer, revenue-generating lifespan for your webware.

Wikipedia.com has a great discussion of object oriented languages for those who want more details.

Rails: The Framework

One of the best ways to define Rails is to describe the problem that it solves.

The problem: Creating a Web-based application can be very time-consuming because of all the details that must be tracked and managed.



Key Points

- Rails helps developers write better software in less time.
- Rails is to software developers as electronic spreadsheets are to accountants. Both tools make their respective professionals more productive by eliminating grunt work.

As users, we take well-built webware for granted. However, every element of every page must be clearly defined and thoroughly tested before users visit the site. Every text box, every drop-down arrow, every data field (credit card number, expiration date, etc.) must be programmed and thoroughly tested before any users visit the site.

Rails Automates Repeated Tasks

Rails automates time-consuming, repetitive tasks (things that were once done manually) so that developers can focus on solving problems for your business.

Here’s an example from another profession. Can you imagine an accountant’s life without a spreadsheet like Microsoft Excel or Lotus 1-2-3? Maybe you don’t have to imagine! You might recall the days of paper spreadsheets, pencils, and 10-key hand-cranked calculators.

Rails is to software developers as electronic spreadsheets are to accountants. Both tools make their respective professionals more productive by eliminating repetitive tasks (grunt work).

Limitations of Rails

Rails and spreadsheets both eliminate grunt work. At the same time, they both have limitations.

Neither is a replacement for rational thought. For example, knowing how to use a spreadsheet won’t make you an accountant because you still need to understand tax law and discounted cash flow analysis.

Likewise, a developer who knows Rails must also understand object oriented programming and relational database theory in order to create a workable piece of software.

A spreadsheet won’t make you an accountant, Rails won’t make you a Web developer, and a big hat in Texas won’t make you a cowboy!




Other Options

By the way, there are other frameworks. There are other languages. Rails is the latest in a long line of tools designed to make programmers more productive. History shows that the tools will continue to improve over time.

Why Do You Care?

You care because in the end, results matter. Further, if there’s a way for your developers to achieve better results with better tools, then tools matter too. Rails is not a panacea for all Web development. You’ve heard the saying “If the only tool you have is a hammer, every

Components of the Rails Framework

Component <i>What is it?</i>	Technical View <i>What does it do?</i>	Business Advantage <i>How does it help you?</i>
 Rails	Rails is a software development framework that eliminates the repetitive tasks (grunt work) of web application development.	A developer who spends less time on grunt work can spend more time listening to you, and turning your business ideas into profitable software.
 Ruby	Ruby is the dominant language of Rails applications.	When properly used, an object oriented language enables a more accurate model of your business, its challenges, and potential solutions. Some developers believe that software written with object oriented languages is easier to maintain long term.
 SQL Database	SQL (structured query language) databases are used in storing and retrieving data rapidly and reliably.	The most exciting Rails applications are database driven. When you store data in a structured way, you can retrieve it rapidly. Even better: You can spot hidden (and potentially profitable) relationships between the data elements.

problem looks like a nail.” Be sure that the developer you choose understands the tool in use and why it was chosen. This may save you headaches in the end.

Evaluating Developer Skills

One good way to measure your developer’s skills: Take a look at projects the developer has completed in the past. Were the clients satisfied? How did the developer handle problems? A good developer, like a good craftsman, will be willing to talk to you about tools and methodologies. The more comfortable you are with the developer and the tools, the greater the chances of project success.

As of this writing, Wisdom has three Rails applications under development. You can check our web site for real-time information about our Rails applications.

How Can You Benefit?

Maybe you can’t. Some business processes are best handled by off-the shelf software. Bookkeeping, for example, is one area where there’s a penalty for creativity so it’s better to do what the government and your investors want you to do.

But keep this in mind. If you’re running your most critical business processes with off-the-shelf software,

so can your competitors. A company that does not stand out from its competitors runs the risk of being commoditized. And all commodities are subject to low profit margins.

Good Ideas, Amplified

Good ideas are amplified when implemented in good software. If you have a bold & innovative business idea, if your idea can be monetized, and if your idea is so unique that it does not exist in off-the-shelf form, then custom Webware in the Rails framework might fit you.

One More Thing...

Rails is a tool for creating webware, and webware runs virtually anywhere! Rails applications are designed to run inside a simple web browser like Microsoft Internet Explorer or Apple Safari. You can even find browsers on personal digital assistants (PDAs).

Imagine the benefit of using the same business software on your PC, Macintosh, and PDA! No new interface to learn, and no special installation steps for each platform.

Advantages of Webware are discussed further in the 2Q2007 issue of *The Wisdom Report* entitled *Webware ROI*. You can download it at WisdomGroup.com.

Suggested Reading

Programming Ruby by Dave Thomas. Anyone who wants to write Rails code must know Ruby. This title is available in paper form and as a downloadable PDF from the publisher.

Agile Web Development With Rails by Dave Thomas, Mike Clark, and several other authors. This title builds upon the Ruby book above. Also available in paper and PDF formats.

About the Author

Raymond T. Hightower is president of WisdomGroup, creators of outstanding web-based software. WisdomGroup leads the open source user group ChicagoRuby.org, and we created the WindyCityRails conference. For more information, visit WisdomGroup.com.