

REALNETWORKS DRIVES INNOVATION AND SAVES TIME AND MONEY WITH SPRING

RealNetworks, Inc. delivers digital entertainment services to companies and individuals, allowing them to create, send and receive audio, video and other multimedia services over the Internet.

RealNetworks introduced the RealPlayer, the first mainstream media player to enable one-click downloading and recording of Internet video, as well as the Rhapsody digital music service, which provides more than 1 billion songs per year to RealNetworks customers. Rhapsody enables users to listen to millions of songs, through a subscription service, without paying per track. Rhapsody users can search by song, artist, title or genre; tune into Rhapsody Channels for continuous, ad-free music; and create and share playlists.

The applications designed by the RealNetworks development team are the core business of the company. The functionality, performance and ease of use of the applications are the reasons customers choose RealNetworks products and services such as Rhapsody. Consequently, application development is a mission-critical operation that directly impacts RealNetworks' ability to retain its leadership position in the marketplace.

CHALLENGE

"Performance of the application is critical to keeping RealNetworks customers happy," explains Paul Davis, Lead Software Development Engineer at RealNetworks, who is currently working on a customer-facing application with an embedded website for Rhapsody. "This is a subscription business, and if our customers are not happy, they go away."

Prior to using Spring, however, Davis relied on an in-house application development framework built and maintained by the RealNetworks development team. This presented several challenges to RealNetworks, hindering the development of top-quality applications.

"The old framework involved a huge amount of development effort," Davis recalls. "It was really showing its age. We had problems maintaining, supporting and updating it. It was getting to the point where we were spending more time trying to maintain the application development platform than building applications on top of it."

"Because of developer turnover, there were so many hands working on the framework, and it became convoluted," he continues. "It got out of control. It was really hard to manage, and side effects from changes were unpredictable."

Prior to the introduction of Spring, RealNetworks had 20 different teams building applications on the old platform, and each team was on their own development cycle, and consequently remained dependent on different versions, further complicating the situation.

"The complexity trickled down to the people building applications," Davis adds. "Developers had to spend a lot of time setting up complex configurations before they could start working on an application."



SPRINGSOURCE RESULTS

SpringSource and Spring delivered the following business results to RealNetworks:

- Time Savings — reduced time to test applications from a month to a couple days
- Increased Productivity — due to simple easy-to-write code
- Rapid Problem Solving — easily accessible support and documentation
- Better Quality Applications — due to better code and automated testing
- More Innovation — increased experimentation generates more innovative new features and capabilities, with less risk
- Reduced Training and Ramp Up — cut time to developer productivity from one month down to one day
- Cost Savings — reduced costs on platform and application development, testing, maintenance and support; and developer training

REALNETWORKS DRIVES INNOVATION AND SAVES TIME AND MONEY WITH SPRING

Application performance was unreliable due to the inconsistencies of the development model. Furthermore, it was difficult to maintain the quality of applications because of the turnover of developers and the challenges of the proprietary development environment.

Developer turnover also impeded team productivity, because of the framework's complexity. RealNetworks was continuously hiring new developers who would initially be consumed by trying to learn the in-house framework. Davis says it would take a month or more of learning curve before a new developer was productive. Because of turnover, a substantial investment of time and money in developer training was lost every time a developer would leave the company. A replacement developer would be brought in and the training and ramp up cycle would start all over again.

Support for the RealNetworks internal framework was also a significant challenge. A key support responsibility was maintaining reference documentation, as well as training new developers. Because of the complex nature of the environment and the multiple versions, it was very difficult for RealNetworks to keep this documentation organized. Consequently, developers had to rely on support information passed along by word of mouth, or random documents kept in unspecified locations.

Application testing was an additional challenge that exhausted time and resources. Platform and application complexity resulted in serious testing backlogs. Teams of four or five developers would test an application for weeks, at the risk of producing new bugs as they made changes throughout the lengthy testing and debug process.

Davis and the RealNetworks development team were facing continuously increasing pressure to get more applications out—"that is the nature of the business," says Davis. Consequently, the problems they were experiencing were only going to get worse. Finally, some of the team leaders like Davis decided all the time and effort devoted to the platform was not worth the trouble, and they then switched to Spring.

SOLUTION

Now, several development teams at RealNetworks, including the team run by Davis, build all applications on Spring, the world's leading platform to develop, run and manage enterprise Java applications. For the web application server, the teams utilize Apache Tomcat, another open source platform supported by SpringSource.

"Broad global use is the major reason we chose Spring over other options," says Davis. "The wide adoption gives us great confidence in Spring, and the applications we build on top of it."

Davis considers Spring more reliable because the prominent open source platform has been tested and proven around the globe by top development teams.

"Everything just works," he says. "We don't spend a lot of time struggling with the system. Now, instead of spending all our time maintaining the old infrastructure, we can focus on building applications."

Davis also finds that Spring empowers developers to write cleaner, more maintainable code, which results in superior application performance and less problems.

"Spring promotes a better development style," he continues. "Spring's dependency injection makes it easier to develop interface-driven code, and the code is in small modular pieces which is easier to test."

Some development teams at RealNetworks are still on the proprietary platform, and Davis feels that the differences between those teams versus the teams using Spring are very clear.

"The difference in morale is night and day," Davis says. "Everyone on our team's morale is much higher, productivity is greater, and the quality of the applications we are building is much better—because we are using Spring."

"On our team, everyone's morale is much higher, productivity is greater, and the quality of the applications we are building is much better—because we are using Spring."

REALNETWORKS DRIVES INNOVATION AND SAVES TIME AND MONEY WITH SPRING

"Other teams see our successes and want to follow that," Davis notes, pointing out that Spring is used by many teams throughout the company, and is growing in popularity as more developers migrate from the previous environment.

BENEFITS

Spring has delivered the following business results to RealNetworks:

Time Savings

Spring saves RealNetworks time on multiple levels. First, simply by eliminating the efforts required for development, maintenance, support, documentation and upgrades on the proprietary RealNetworks framework, Spring saves the company an enormous amount of time.

"The overall development time is much quicker because we don't have to worry about dealing with infrastructure issues," Davis adds, explaining that his team can get more applications out the door faster.

According to Davis, Spring facilitates automated testing, which accelerates time to market on new applications. Previously, application testing could take as long as month, but with Spring it only takes a few days.

"We used to have four or five people testing the application for several weeks," Davis says. "With Spring, we just push a button."

Increased Productivity

Davis maintains that because the code is easier to develop in Spring, developer productivity is higher.

"Spring makes interface driven design easier," explains Davis. "Parts of the application can be built in parallel, and the application is easier to test. The 'pluggability' of various components and the ease that Aspects can be applied allow many capabilities—such as transactions and caching—to be transparent to the application code. This means the code is cleaner, simpler, and easier to maintain."

Rapid Problem Solving

Time savings is also a direct result of the fact that Spring is more reliable than RealNetwork's previous proprietary platform, and problems are solved quickly because of the myriad of resources available for Spring.

Better Quality Applications

Spring enables the team to be more productive, which not only means the developers can produce more applications, but they can also put more effort into the quality of the applications.

"A side effect of using Spring is that people tend to write better code," Davis says. "A good development platform is going to influence the way you write code, and because Spring makes it easier to use small modular chunks and plug-ins, our developers tend to write that style of code. Smaller and simpler code means better quality applications that are easier to maintain."

Spring also makes automated testing easier, according to Davis, because the applications are more modular. Fast, repeatable tests simplify and accelerate the debugging of the application prior to production—which not only saves time but also ensures application quality. Davis notes that if it takes a month to test the application, there is a chance of actually producing additional bugs during the long, drawn out debugging process—while rapid testing enabled by Spring preserves the integrity of the application.

Application quality and performance is critical to customer satisfaction, especially in a subscription business, and Spring supports this vital objective.

"Prior to using Spring, we averaged from three weeks to a month for a new developer to ramp up. Now, a new developer usually gets started on application development on the first or second day."

REALNETWORKS DRIVES INNOVATION AND SAVES TIME AND MONEY WITH SPRING

More Innovation

More than just the quality of the application, the functionality is a critical selling point. The features of RealNetworks celebrated customer-facing applications are the reasons customers sign up in the first place.

"Innovation of new capabilities is critical to the success of the company," Davis states. "Because Spring saves us time, it enables us to develop more features for our applications. In addition, the modular way applications are built in Spring makes innovation easier and gives us more freedom and independence to experiment because the risk of causing side effects is much lower. If someone has an idea, we can just plug it in and see how it goes. If it works, we leave it in. If it doesn't work, we simply remove it. We do not have to worry about any negative impacts to the application."

Reduced Training and Ramp Up

Spring enables new RealNetworks developers to become productive almost immediately, because training and ramp up on Spring is not as time-consuming and challenging as with the proprietary development platform. Davis says he can simply hire developers that know Spring. This is a dramatic change compared with the previous ramp up method that required in-depth training on a confusing, constantly changing proprietary platform. For job postings, Davis lists Spring as a job requirement. Because it is so ubiquitous, he can easily find new developers with Spring experience.

"Prior to using Spring, we averaged from three weeks to a month for a new developer to ramp up," Davis remembers. "Now, a new developer usually gets started on application development on the first or second day. For the Rhapsody application we are currently working on, we brought in a new front-end developer and she was able to start writing code her first day on the job."

Davis notes that this Spring advantage not only applies to new developers, but any developer starting on any new project, because there is no need to become familiar with a different version of the development platform or complex configurations unique to the application. Developer portability with Spring enables RealNetworks to easily reallocate resources from one project to another without sacrificing any productivity.

Cost Savings

Many of the Spring advantages cited by Davis also result in the additional bottom-line benefit of saving money for RealNetworks. For example, Spring saves RealNetworks on costs related to: developing, testing, maintaining, supporting and upgrading the platform; developing, testing and debugging applications; application maintenance, due to higher application quality and reliability; and training and ramp up of new developers.

"Spring requires less manpower on projects, and manpower translates into money," Davis concludes.

"Innovation of new capabilities is critical to the success of the company . . . The modular way applications are built in Spring makes innovation easier and gives us more freedom and independence to experiment."



SpringSource, Inc.

411 Borel Avenue
Suite 101
San Mateo, CA 94402
USA
+1 800-444-1935