



Bringing Reactive Applications to the Java Virtual Machine

Improving development velocity at Janrain

Consolidating technologies and simplifying the development process were key goals for Anthony Dupré when he assumed the engineering leadership role at Janrain. By leveraging his experience with tools and best practices learned through many years of building mission-critical applications, he quickly enabled his engineering organization to realize dramatic improvements.

About Janrain

Like so many other great companies, Janrain was started from a simple observation - that the number of username and password combinations required of consumers on the web was reaching an untenable state. In late 2004, Founder Larry Drebes and a few engineers began working closely with the early proponents of the OpenID protocol to provide a simple way for companies to solve this problem. Their work laid the foundation for Janrain and what is known today as social login. As they began helping organizations implement the technology, they quickly realized the solutions could be used to create richer, more meaningful relationships between website visitors and customers. Today, the Janrain User Management Platform (JUMP) makes it easy for clients to acquire, engage and learn more about their consumers.

The Business Problem

Janrain had chosen Ruby and Haskell as the tools they would use for the first iteration of their platform. Ruby provides development agility and productivity, while Haskell provides the benefits instilled by a functional programming language - expressive and compact code.

Companies often start with a variety of disparate tools that fit their solution architecture, only to consolidate them as their long term requirements become more tangible. Anthony Dupré, Vice President of Engineering decided that building the next generation of products at Janrain required standardizing around a cohesive toolset that embodied their core requirements of performance, scalability, agility and productivity.

The Platform of Choice

Anthony and Thomas Lockney, a Software Architect, had both used Scala for a number of years prior to joining Janrain and knew that they could benefit from using it again. Scala is a general purpose programming language designed to express common programming patterns in a concise, elegant, and type-safe way. It smoothly integrates features of object-oriented and functional languages, enabling Java and other programmers to be more productive. As it turns out, Scala had quietly made its way into the company months before as a way to enhance a Ruby and Java based service. Scala was a natural choice given that it was appealing to the engineering teams, and would help Janrain in consolidating around a toolset that most developers already knew well, and had already been at least somewhat established in the Janrain environment.

Building with Scala

With many developers already exposed to functional programming with Haskell and even some existing Scala use, extending the commitment to Scala was easy. For the non-Haskell programmers, the biggest challenge for them seemed to be moving from dynamically typed languages to a statically typed one more than anything else.

The fact that we have a very mixed, diverse set of environments in-house, and a large portion of the engineering team here has been exposed to a number of different projects, made it very easy to commit to Scala at Janrain.

Thomas Lockney, Software Architect, Janrain

With Scala as the language, Janrain has two standard “stacks” that they utilize: Akka with Spray for Web Services, and Play Web Framework for user interface development. Akka is a toolkit and runtime for building highly concurrent, distributed, and fault tolerant event-driven applications. Spray is a toolkit for building REST/HTTP-based integration layers on top of Scala and Akka. Being asynchronous, actor-based, fast, lightweight, and modular it's a great way to connect your Scala applications to the outside world. Play Framework is an MVC based web framework that focuses on developer productivity, modern web and mobile applications, and predictable, minimal resource consumption resulting in highly performant, highly scalable applications. Scala, Akka, Play and Spray are all components of the Typesafe Reactive Platform.

An unexpected benefit of adopting the Typesafe Reactive Platform is that it certainly attracts top engineering talent to the company.

Our social login platform is about 70% Scala and approximately two-thirds of our fifty-person development team is working primarily in Scala.

Anthony Dupré, Vice President, Engineering, Janrain

Deployment

Janrain's products are primarily deployed on the Amazon Cloud. The value that they see in the Typesafe Reactive Platform is that it is enabling them to move towards elastic sizing in the Cloud. It's going to definitely reduce costs moving forward, compared to the previous way that they had deployed, which was using JRuby running within a Tomcat container.

Akka, in particular - especially with Akka 2.2 and the clustering capabilities – has enabled Janrain to very quickly spin up a new approach that allows a lot of flexibility in how they deploy applications

Thomas notes that utilizing the Typesafe Reactive Platform has allowed Janrain to free themselves from the technical debt that was built up by being constrained by a container based environment, thus resulting in far more flexibility from an engineering standpoint.

From a DevOps standpoint, the team already had a multi-language, multi-environment set up that was built with the assumption that it would have to interact with a lot of different types of services, types of metrics, and logging. Indeed, the monitoring and operational environment has largely been based on the more typical modern stack, like Sensu/Graphite and log stacks with Elasticsearch. Consequently it has been very easy to quickly plug into that new environment, and this new development approach has significantly simplified Janrain's overall environment.

Moving Forward

Janrain is moving from a very product focused, vertical structure approach to a more "service oriented" architecture. "Scala and Akka fit just perfectly with Janrain's small, lightweight, service component-based development modules," said Thomas.

Undoubtedly, the biggest benefit seen is the velocity of development. Teams are seeing much quicker turn around on functionality and developer productivity. Being able to share knowledge across teams and using a platform that is targeted towards a very fast, iterative development model really has improved things dramatically.

*The **Typesafe Reactive Platform** is a modern software platform that makes it easy for developers to build scalable software applications. It combines **Play Framework**, **Akka** runtime, the **Scala** programming language, and robust developer tools in a simple package that integrates seamlessly with existing Java infrastructure. Commercial support and maintenance is available for the Typesafe Reactive Platform through the **Typesafe Subscription**.*