

# Open Source Strategies

## *The RedMonk Going Open Source Series, Part 2*

### Overview

Michael Coté

James Governor

Stephen O'Grady

This note discusses open sourcing strategies in use by organizations and companies – with the goal of providing background for companies considering open sourcing parts of their portfolio. We focus on how these entities use open source in their overall strategy, highlighting major brands and projects where appropriate. Additionally, we discuss license and governance choices. We do not focus on technological considerations, such as architectures that favor commercial open source offerings, nor do we describe the numerous tactical uses of open source in the software marketplace. While we do not discuss in depth why open sourcing is beneficial to vendors, their customers, and users, we touch briefly on the benefits.

### BEA - Working With Open Source

BEA calls their open source strategy a "blended" approach. They certify that their closed source stacks, the WebLogic application server in particular, work with select open source projects. Their strongest and highest profile relationship is with Spring; but they also "blend" with Apache Beehive, Open JPA, Struts, and the Apache JSF project. Additionally, BEA supports those projects running alongside and inside of WebLogic. Also of interest is that BEA supports the use of the open source JEE web container Tomcat and the JEE application server Geronimo – but for development use only.

In addition to certifying projects to run alongside WebLogic, BEA is involved in open source in the following ways:

- **BEA hosts an open source community site called CodeShare.** CodeShare allows community members to host open source projects related to the BEA ecosystem, for example, extra code and scripts to

configure portlets in WebLogic Portal. Projects on CodeShare must be released under the BEA Public License .

- **BEA uses Eclipse as the framework for its IDE tools.** BEA is also an Eclipse Board Member and is involved in the Web Tools Platform and AspectJ projects at Eclipse.

BEA's relation to open source licenses is primarily with Eclipse and Apache. Being an Eclipse Board Member and using Eclipse, they have a strong relationship with the Eclipse Foundation.

The governance models used by BEA depend on the outside organizations that house the projects they're involved in. A project hosted on CodeShare follows an Apache-like process wherein projects are initially put in incubation and then graduated to full public projects, with the help of an assigned Community Manager. Once a project is established, project members gate new members of the project.

### **RedMonk Red Lights**

BEA's strategy is not about open sourcing WebLogic, Tuxedo, or other major BEA software assets. Instead, it meets the demand for open source frameworks and middle-ware half-way, acknowledging and even supporting the use of open source in the development cycle and production.

Reaction to this open source strategy has been "blended" itself. On the extreme end, Marc Fluery (who had an obvious bias at the time at JBoss) labeled BEA as an "open source dumping" company who had, at the time, given over Beehive to the Apache Foundation. The majority see BEA's strategy as a pragmatic acceptance of open source. As with most middle-ware and "lower level" vendors, BEA has climbed up the value chain, re-focusing on higher-level components like SOA, portals, and Web 2.0 offerings. BEA also differentiates its commercial application server offering by marketing the production options for clustering, hot deployment, high availability, and performance. That is, they deploy the "we're better"

defense against open source offerings such as JBoss, Tomcat, and Geronimo. Unlike IBM, BEA does not offer a production open source option to its customers.

It is also worth noting that open source competitors, JBoss most notably, have had a material impact on BEA's sales prospects. JBoss sales staff, as an example, would promise WebLogic customers JBoss acquisition and licensing costs that were less than the ongoing maintenance costs for the WebLogic product.

### **IBM - All Over the Board**

Of all the large commercial software vendors with involvement in open source solutions, IBM has one of the most sterling reputations, enjoying the collective good will of open source developers across the world. This is due in part to their wide involvement and significant contributions (Eclipse, Linux, etc), but also to their unapologetic non-altruistic positioning. Their position at the heart of the SCO case has also been beneficial from a public relations perspective, as it has led to their being viewed as defenders of the Linux kernel.

While IBM is heavily involved in open source, the majority of their software revenue still derives from sales of closed source products – proving that open source is not necessarily inimical to proprietary revenues. Generalizing IBM's open source strategy is imprecise, but they could be said to use four strategies:

- **Amortizing the cost of software development across multiple organizations.** Rather than bear singularly the costs of developing web servers (Apache), application servers (Geronimo), operating systems (Linux) and development tools (Eclipse), IBM instead is able to lessen their development costs by relying on collaborative open source development costs. This is particularly evident in releases such as Lotus Connections, which ships as infrastructure dozens of different open source projects.

- **Supporting and servicing open source projects.** With a services arm that comprising hundreds of thousands, it's no surprise that IBM realizes significant and ongoing revenue opportunities in selling and supporting open source projects. The highest profile example of this is perhaps WebSphere CE, a free-to-use application server with the option for commercial support. Following the acquisition of Gluecode in 2005, IBM now offers an Apache Geronimo- based open source alternative that competes at least indirectly with its closed source cousin.
- **Helping to drive non-software businesses.** The opportunities afforded its services business have already been discussed, but the impact of open source on its hardware businesses – from the x86 to the POWER to the Z platforms – is likewise considerable. Linux with the application availability it affords has opened significant doors for IBM within its customer base and without.
- **Being a patron for industry-wide open source efforts.** IBM spun off the Eclipse Foundation (now an independent powerhouse in the open source world), and works on several Apache projects and many other open source projects.

The wide range of IBM involvement in open source means that IBM is associated with numerous open source licenses – from GPLv2 to BSD. IBM has strong alliances with Eclipse, Apache, and the Linux world. Its strong commitment to Linux notwithstanding, IBM's stated preference is for permissive licenses (e.g. Apache) over reciprocal licenses (e.g. GPL).

The governance model IBM uses for open source efforts is characteristically external; that is to say that IBM does not, typically, maintain and run projects themselves, perhaps realizing that independent projects are more likely to attract outside participation. Examples here include Apache Derby and Eclipse.

### **RedMonk Red Lights**

Not being a software company, IBM is able to rely on hardware, services (IGS), and even managed hosting offerings to drive revenues. This is not to say that IBM software sales are lacking, of course. Rather, the its wide spectrum of revenue streams allows IBM to diversifies risk with respect to any single product by carefully investing in a variety of software areas and amortizing the costs of development across multiple organizations. The result is that IBM is able to offer customers open and closed source options for customer problems - and dollars - rather than having to drive customers to primarily closed source offerings.

### **Sun - Pragmatically Going Open Source**

Like IBM, Sun is involved in open source in many different ways. Sun's stated vision is that its entire software portfolio will become open source; this promise has been supported in recent years by the release of its flagship Java and Solaris offerings under open source licenses – the GPLv2 and the CDDL, respectively. Apart from those two major projects, Sun is open sourcing several "smaller" projects such as OpenLDAP and OpenSSO. Sun, like IBM, is also a frequent patron of open source projects by means of employing key developers on open source projects such as Apache Roller and JRuby.

### **Open JDK**

Java is dual-licensed – with the same commercial license it always has had, and the GPLv2+Classpath Exception license. Contributions are currently restricted to patches, meaning that Sun will accept patches but does not have outsider committers and team members. These patch contributions are submitted under a common JCA, or Joint Copyright Agreement. This grants copyrights to both Sun and the original contributor; JCA's are typically employed so that material changes to the project – with respect to licensing, as an example – do not require the parent project to seek the individual permission of each and every contributor.

Although open sourcing Java was long in coming, when it finally happened, reaction from the open source community was overwhelmingly positive. It in fact led certain projects to reconsider Java where it had not previously been an option; the Mono project lead, Miguel de Icaza, admitted in an interview that had Java been open source all along Mono might well have been unnecessary. The positive reactions to the licensing were due, at least in part, to the specific selection of the GPLv2+Classpath Exception license, which is popular among open source developers (nearly 70% of projects on Sourceforge employ some iteration of the GPL). Of course, that also meant that others with interests in Java, such as the Apache developers or IBM, who favored a more permissive license, reacted coolly to Open JDK.

### **Open Solaris**

Solaris was probably hurt more severely than any other commercial operating system by the rise of Linux. Non-Windows customers – large and small – increasingly selected the younger Linux over the one-time default choice for scalable architectures. This was due in part to technical issues with Solaris – particularly its performance – and in part to barriers to entry: notably the fact that, unlike Linux, Solaris was not open source and freely available. Sun has sought to address both of those concerns by remedying the technical problems – as well as introducing technically differentiating features such as Dtrace and ZFS – and open sourcing the project.

While the open sourcing effort is still underway, as Sun tries to work around encumbrances and integrate external developers into its development process, the initial returns on the significant investment are almost universally positive. While Linux continues to perform well, Solaris appears to have regained its footing and is even making gains in certain spaces. The decision to open source the project appears to have played a significant role in that success.

## RedMonk Red Lights

Sun's vision for a go-to-market with open source is more purely open source than IBM in that Sun hopes to eventually open source all of its closed source software – an admittedly simpler task because Sun's portfolio does not command comparable revenue to that of IBM's. Sun's vision can essentially be distilled down to the monetization of ubiquity and volume, relying on support, service, managed services (such as SaaS), and hardware sales for revenue. So the current actions of Sun in the open source world can be seen as continually figuring out the best process for open sourcing its wider portfolio. Sun has exhibited the willingness to take the time to learn what the best options are and evolve their general strategy on a case-by-case basis, as needs dictate. As with any large, mature company, making the widespread change to open source will be a long process, largely one of changing culture and significant legal due diligence rather than simply jumping technological hurdles.

## Actuate

Actuate provides several products in the business intelligence space. In the context of open source strategies, their heavy involvement in and use of Eclipse BIRT is relevant to the discussion. Reporting and business intelligence have been large, established markets for some time, but ones that were somewhat stagnant and limited in the small and medium business markets. By committing to an open source offering, Actuate hoped to expand their overall addressable market; and they've been successful to some degree. Actuate is part of the BIRT project at Eclipse and uses BIRT as the base for products and services it sells. For example, Actuate recently entered into a service agreement with GroundWork Open Source to help GroundWork developers use BIRT to create IT management reports and business intelligence views. Partnerships such as this between open source companies are increasingly common.

Being housed at Eclipse, BIRT follows the Eclipse governance process and is licensed under the EPL.

### **RedMonk Red Lights**

Like other companies profiled in this paper, Actuate has realized the need to embrace open source as a tools provider. Providing Eclipse BIRT as a fully functional business intelligence tool allows Actuate to lower barriers to entry to the BI space for developers and others. When companies using BIRT need assistance or higher level products and services, Actuate is available to help. This strategy allows Actuate to monetize at the point of value by helping to solve complex, rather than simple, problems.

### **Alfresco - A Dual License Example**

Alfresco offers an enterprise content management system. The company started with the Mozilla Public License, but has recently moved to using the GPL+FLOSS Exception. Currently, this means their offerings are covered under the GPLv2 (though it contains the "future versions of GPL may be used" clause) with the hope that other OSI licensed open source projects will not be incompatible with Alfresco.

As with Java, Alfresco offers its product under a dual license scheme. In simple terms, the dual license model – of which MySQL is probably the most notable adherent – offers potential Alfresco users two options: a GPL licensed version, or a commercially licensed version. Users and developers may use Alfresco under a commercial license or the GPL. Along with the FLOSS Exception, this scheme gives Alfresco's customers maximal latitude in how they account for the use of Alfresco. The commercial, or "Enterprise Edition," of Alfresco is billed as having more rigorous testing, support, faster bug fixes, and integration with a wider range of 3rd party and sub-systems, such as databases.

Under dual licensing schemes, the company typically maintains control over all IP contributed to the project. Because of this, the governance process requires any participants to at least share the copyright on code they contribute, if not give over all rights.



### **RedMonk Red Lights**

Alfresco is one good representative of the dual licensing model that many new "open source companies" are using. Under this approach, companies depend on the revenue streams from closed source customers who want to avoid the reciprocal open sourcing that the GPL may require. Additionally, dual license offerings typically add additional services with the commercial offering as Alfresco does. An interesting additional component that dual licensing companies such as MuleSource and MySQL offer with their commercial versions are management tools – in both of those cases, OEM'ed from Hyperic.

### **RedMonk Take**

As the above discussion shows, there are many ways to go about the business of open source software, and a variety of revenue strategies. Several large companies prefer to take their efforts to open source organizations such as the Eclipse and Apache foundations. When the technology is a long-time core asset, as with Java, companies tend to in-house their open sourcing efforts. On the extreme end, as with IBM and Eclipse, a company will create a new, external organization.

The governance processes differ widely, but largely follow a rule of meritocracy when admitting new project members. Choosing a custom or unique license – which would have to be approved by the OSI for maximal acceptance – is not recommended. Selecting an existing license is the preferred path, for both PR and educational reasons. Additionally, a dual licensing approach is often attractive when open sourcing previously closed source projects: many existing customers are happy to continue the exact same commercial relationship with the company, regardless of open source options.

The selection of an open source license may be dictated by preferences in several dimensions :

1. **Control:** How much control is needed over the code and how is it to be used? Dual licensing offers the most control, but the least benefit in amortizing the cost of development. Similarly, the reciprocal provisions of the GPL, which require that any downstream distributions of the codebase carry the exact same license, act as a deterrent to forks and the co-opting of the project. It also can act to restrict commercial involvement, however. In addition to other intellectual property requirements, the GPL requires that modifications to the original codebase be made available, while more permissive licenses such as the Apache License make no claims on derivative works or fixes to the code.
2. **Revenue:** What are the monetization requirements and expectations for the project? Will the primary financial benefit be money earned (support, service, licensing, etc.) or money saved (collaborative development, distributed QA and bug reporting, etc.)? Offering commercial support works for any license, while the stipulations for re-use can create revenue opportunities by allowing customers to buy their way out of complying with those stipulations.
3. **Culture:** Which open source philosophy will be optimal for your project's success? The choice of license, governance, contribution model, and even source code management technologies, can all play a strong role in defining the culture of a given project – either existing or new. Each license type will have distinct implications – positive and negative – for adoption, commercial investment, distribution, and so on. Another important factor to consider is the perception that existing customers and share-holders may have of the license you choose.

The decision of how and when to open source software, particularly in the context of commercial vendors considering the open sourcing of previously closed source software, should be driven by clear high-level goals. The relative importance of volume, revenue, community contributions, and so on will inform the decision-making process, as they will imply certain choices.

While the decision to open source takes some effort and planning, RedMonk strongly recommends that companies look towards open source as a means of realizing significant and sustainable benefits in distribution, code quality, and more. Open source as a method of software delivery, use, and development is here to stay. The success of the companies profiled above among others shows that open source is ultimately beneficial not only to customers and users, but to software vendors as well.

## **About the Creative Commons License**

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs License. To view a copy of this license, visit

<http://creativecommons.org/licenses/by-nc-nd/2.5/>

or send a letter to

Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

## **About RedMonk**

RedMonk is a research and advisory services firm that assists enterprises, vendors, systems integrators and corporate finance analysts in the decision making process around today's enterprise software stacks. We cover the industry by looking at integrated software stacks, focusing on business and operational context rather than speeds and feeds and feature tick-lists.

Founded by James Governor and Stephen O'Grady, and headquartered in Denver, Colorado, RedMonk is on the web at [www.redmonk.com](http://www.redmonk.com).

If you would like to discuss this report email Michael Coté ([cote@redmonk.com](mailto:cote@redmonk.com)), James Governor ([jgovernor@redmonk.com](mailto:jgovernor@redmonk.com)) or Stephen O'Grady ([sograd@redmonk.com](mailto:sograd@redmonk.com)).