



Ubuntu:
Windows looks increasingly bad,
archaic, and unfriendly
6 Paul Nowak

Interview:
Mike Milinkovich,
Eclipse Foundation
12 Interview by Joe Winchester



ENTERPRISE OpenSource MAGAZINE

Visit us at opensource.sys-con.com

THE LEADING MAGAZINE FOR ENTERPRISE AND IT MANAGEMENT

APRIL/MAY 2007 VOLUME 5 ISSUE 3

USING THE FORCE AND OPEN SOURCE

3 **THE LINUX DESKTOP**
MARK R. HINKLE

5 **VIRTUALIZATION CHANGES**
JAMES BOTTOMLEY

10 **PROJECT LASSO FOR
LOG MANAGEMENT**
JON WALKER

14 **THE OPEN SOLUTIONS
ALLIANCE**
STACEY SCHNEIDER

18 **IT'S NOT JUST
THE CODE**
SUJEE SAPARAMADU

20 **THE EVOLUTION OF
ENTERPRISE APPS**
STEVE YASKIN

26 **HOW TO ACQUIRE AN
OPEN SOURCE
SOFTWARE COMPANY 2.0**
DOUGLAS LEVIN



28 FRANZ GARSOMBKE

PRESORTED
STANDARD
US POSTAGE
PAID
ST. CROIX PRESS

SEE PAGE 23
End Annual
ENTERPRISE 2007
OPENSOURCE
CONFERENCE+EXPO

SEE PAGE 25
2007
VIRTUALIZATION
CONFERENCE+EXPO
www.virtualizationconference.com

June 25-27, 2007
New York City



VERIO HOSTING IS LINUX WITH A LINEAGE.

- **Root Access:** Providing the control you need.
- **Advanced FairShare Technology:** Better resource management means better performance.
- **Support That's Actually Supportive:** Award-winning support provided by system administrators.

Announcing Verio Linux® VPS.

At Verio, we have a long-standing commitment to open source, dating back to our roots in FreeBSD. Now, as the pioneer in virtual private server (VPS) technology and as a hosting provider backed by the financial resources of the world's largest telecommunications company, we bring something extra to Linux: reliability. To learn more, call 1-877-837-4654 or visit www.verio.com/linuxlineage.

There is no substitute for the right foundation.

Verio and the Verio logo are trademarks and/or service marks of Verio Inc. in the United States and other countries. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries. All other names are trademarks or registered marks of their respective owners. ©2007 Verio Inc. All rights reserved.

Build on us.
VERIO
An NTT Communications Company

EDITORIAL BOARD

Editor-in-Chief

Mark R. Hinkle mrhinkle@encoreopus.com

Contributing Editor

Ibrahim Haddad ibrahim.haddad@sys-con.com

Migration Editor

Jon Walker jwalker@sys-con.com

Commercialization Editor

Paul Sterne sterner@sys-con.com

Desktop Technology Editor

Tim Griffin tim@sys-con.com

Review Editor

Matt Frye mattfrye@sys-con.com

Editor

Philip Peake philip.peake@sys-con.com

Contributing Editors

Kevin Larue kevinl@linspireinc.com

Christopher Negus cnegus@mwmt.net

Contributor

Rob Jones rob@hotlinuxjobs.com

INTERNATIONAL ADVISORY BOARD

Wim Coekaerts; *Director of Linux Engineering, Oracle*

Brian E. Ferguson; *Partner, McDermott, Will & Emery*

John Fowler; *Executive VP, Network Systems Group, Sun Microsystems*

Gaël Duval; *Cofounder/Director of Communication, MandrakeSoft*

Samuel J. Greenblatt; *Sr VP and Chief Architect, Linux Tech. Group CA*

Scott Handy; *VP, Linux Strategy and Market Development, IBM*

Bruce Perens; *Perens, LLC*

Stacey Quandt; *Principal Analyst, Quandt Analytics*

Thomas Reardon; *VP and GM, Client Product Group Openwave Systems*

John Weathersby; *Executive Director, Open Source Software Institute*

Ranjit Nevatia; *Director of Linux Strategy, VERITAS*

Andy Astor; *co-founder and CEO, EnterpriseDB*

EDITORIAL

Executive Editor

Nancy Valentine nancy@sys-con.com

Research Editor

Bahadır Karuv, PhD bahadir@sys-con.com

OFFICES

SYS-CON MEDIA

577 Chestnut Ridge Rd. • Woodcliff Lake, NJ 07677

Telephone: 201 802-3000 • Fax: 201 782-9600

Enterprise Open Source Magazine (ISSN #PENDING)

is published monthly (12 times a year)

by SYS-CON Publications, Inc.

Postmaster send address changes to:

ENTERPRISE OPEN SOURCE MAGAZINE

SYS-CON MEDIA • 577 Chestnut Ridge Rd. • Woodcliff Lake, NJ 07677

Copyright © 2007 by SYS-CON Publications, Inc.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy or any information, storage and retrieval system, without written permission. For promotional reprints, contact reprint coordinator.SYS-CON Publications, Inc., reserves the right to revise, republish and authorize its readers to use the articles submitted for publication. All brand and product names used on these pages are trade names, service marks, or trademarks of their respective companies.

WorldWide Newsstand Distribution

Curtis Circulation Company, New Milford, NJ

FOR LIST RENTAL INFORMATION:

Kevin Collopy: 845 731-2684, kevin.collopy@edithroman.com

Frank Cipolla: 845 731-3832, frank.cipolla@epostdirect.com

The Linux Desktop Marches On



By Mark Hinkle

There was a time when you couldn't shut me up about the Linux desktop. I was a fanatic. In 2000, I made the switch to a full-time virus-free Linux desktop and weeks of crash-free computing. I was a zealot. However, I did suffer from a few of the *alternative* operating systems shortcomings. My preferred desktop vendor deemed my Linux laptop¹ unsupported, so if I ever had a problem, I had to boot into Windows to receive assistance. When someone sent me a macro-laden spreadsheet, I was forced to run Excel within a virtualized Windows instance² to read the document as intended. Finally, when it came to wireless, I suffered a multitude of connection problems. While I loved the speed, the stability, and the security, it lacked convenience.

In the fall of 2006, I sold out; I flipped to another operating system – Mac OS X. I still had a bash shell, UNIX stability, and many of the benefits of my Linux desktop, but I also got manufacturer support for peripherals (specifically my EVDO broadband card to which I have become addicted) and support for my operating system. I didn't go cold turkey; in fact I continued to run a Windows and an Ubuntu Linux virtual machine on my desktop thanks to Parallels. I justified my move to OS X as an open source-inspired operating system³ owing its lineage to BSD. At the heart of my move was a combination of a need for applications and for support for a number of peripherals required for my day job. However, I think I might have acted to quickly.



Despite my personal sellout, I have become even more bullish on the Linux desktop than ever before. You see since my acquisition of a shiny, silver MacBook Pro, the world of the Linux desktop has changed. First, and foremost, Ubuntu – the wildly popular Linux desktop – has come on like gangbusters in the last two years. In January 2006, I published a book⁴ offering advice for business users on migrating to Linux desktops. I meticulously spelled out all the advantages – stability, security, low cost – and warned of pitfalls – application availability and manufacturer hardware support. In that time I talked about Red Hat Enterprise Linux, Novell SuSE Linux, Debian, Linspire, Xandros, and a few others but made little, if any, mention of Ubuntu. Shortly thereafter I started using Ubuntu, and since then I can't think of single conversation on the Linux desktop where I haven't mentioned the community-driven, financially backed Canonical creation.

On May 1 2007, Dell⁵, through a partnership with Canonical, the company that sponsors Ubuntu, announced that they would be shipping desktop and notebook products with Ubuntu 7.04. This was a big coup for desktop Linux as this is not Dell's only desktop Linux offering, but pre-installed Linux laptops also implies support for all included hardware from the manufacturer. That's a huge step forward in desktop Linux adoption. Many organizations want to procure their desktop and laptops from a single source. Now, at least those that use Dell can procure both Windows and Linux laptops from a single source. Availability in existing supply channels, in my opinion, is critical for corporate adoption that progress has been made through this deal.

Another point of encouragement is the high-profile philanthropic desktop Linux program – One Laptop per Child (OLPC). OLPC is as much an education initiative as one for Linux desktop adoption. This project aspires to put low-cost computers into the hands of children in developing countries, giving them access to online learning. On May 12, 2007, Uruguay President Vasquez inaugurated the first laptop school in Villa Cardal where 150 children received OLPC laptops. Over

—continued on page 27

About the Author

Mark R. Hinkle, editor-in-chief of *Enterprise Open Source Magazine*, is the vice president, Community and Business Development at Zenoss Inc. He serves as a founder of the Open Management consortium and is the author "Windows to Linux Business Desktop Migration" (Charles River Media).

mrhinkle@encoreopus.com



~~I CAN'T GET MY DATA OUT OF OUR CRM SYSTEM.
CUSTOMIZING OUR CRM SYSTEM IS OFF-LIMITS.
OUR CRM SYSTEM IS SO INFLEXIBLE.~~

SugarCRM® solves the problems that proprietary CRM solutions created.
Gain control over your business critical data.

Start using the most powerful commercial open source CRM solution for managing sales, marketing, support and group collaboration activities today.

Get hooked on SugarCRM. Visit www.sugarcrm.com/eosm or call +1 408.454.6941.

Commercial Open Source Customer Relationship Management



SUGARCRM®

COMMERCIAL OPEN SOURCE

www.sugarcrm.com • www.sugarforge.org
1.877.SUGARCRM • +1.408.454.6941

President & CEO

Fuat Kircaali fuat@sys-con.com

Group Publisher

Roger Strukhoff roger@sys-con.com

ADVERTISING

Senior VP, Sales & Marketing

Carmen Gonzalez carmen@sys-con.com

Advertising Sales Director

Megan Mussa megan@sys-con.com

Associate Sales Manager

Corinna Melcon corinna@sys-con.com

EVENTS

Events Manager

Lauren Orsi lauren@sys-con.com

Events Associate

Sharmonique Shade sharmonique@sys-con.com

PRODUCTION

Art Director

Alex Botero alex@sys-con.com

Associate Art Directors

Abraham Addo abraham@sys-con.com

Louis F. Cuffari louis@sys-con.com

Tami Lima tami@sys-con.com

CUSTOMER RELATIONS

Circulation Service Coordinator

Edna Earle Russell edna@sys-con.com

Alicia Nolan alicia@sys-con.com

SYS-CON.COM

VP Information Systems

Bruno Y. Decaudin bruno@sys-con.com

Information Systems Consultant

Robert Diamond robert@sys-con.com

Web Designers

Stephen Kilmurray stephen@sys-con.com

Richard Walter richard@sys-con.com

ACCOUNTING

Financial Analyst

Joan LaRose joan@sys-con.com

Accounts Payable

Betty White betty@sys-con.com

SUBSCRIPTIONS

TOLL FREE 888-303-5282

201-802-3012

subscribe@sys-con.com

For subscriptions and requests for bulk orders,

please send your letters to Subscription Department

Cover Price: \$5.99/issue

Domestic: \$49.99/yr (12 issues)

Canada/Mexico: \$79.99/yr

all other countries \$99.99/yr

(U.S. Banks or Money Orders)

Back issues: \$12 U.S. \$15 all others

Virtualization Changes in 2.6.21

By James Bottomley

Back in July 2006, one of the most contentious issues at the Linux Kernel Summit was what to do about virtualization. At that time, there were three contenders: Xen, VMware, and OpenVZ (the latter being a lighter-weight container-based approach). The biggest fight was between Xen and VMware over competing approaches to running kernel operations through their respective hypervisors: Xen touting their hypercall interface and VMware touting their VMI (Virtual Machine Interface) approach. Neither of these approaches was palatable to kernel developers for a variety of reasons, most of which were technical but also because selecting either would give that contender a great public relations boost over their competitor.


The reason for the fuss? Virtualization schemes like Xen and VMware can be programmed to work correctly on unmodified operating systems. However, they do this by intercepting certain actions the operating system takes (like trying to modify the page tables) and redirecting these operations so as to give the operating system the illusion that it's running exclusively on the hardware, while in reality, it is just one among many running in a so-called "virtual machine." The problem with this interception is that it's expensive (it really uses special CPU hardware to execute a trap into the hypervisor whenever the operating system does something that the hypervisor needs to check). Interception became much cheaper with CPU versions that contain virtualization technology (Vanderpool in Intel and Pacifica in AMD); however, it would be cheaper still if, instead of having to use hardware to watch the operating system, the virtualization layer could just plug into all of the operations that it needs to intercept. The two schemes being pushed by Xen and VMware are essentially different ways of plugging into the operating system.

The germ of a compromise at the Kernel Summit was achieved using an approach that had been taken by the venerable Power PC hypervisor interface (PPC has long had a firmware hypervisor with Linux support on the IBM p series of hardware). This approach, called paravirt ops (ref: <http://lwn.net/Articles/194543>), would essentially provide a well-defined but pluggable interface that could sit underneath the competing approaches and allow either to work comfortably in the kernel – sort of like a universal socket in the kernel for the Xen and VMware plug-ins.

In the interim, in kernel 2.6.20, a new virtualization interface, called KVM, went in with comparatively little fuss, ramping up the pressure on both Xen and VMware to produce their paravirt ops patches.

Finally, in the latest kernel version (2.6.21 released on April 25, 2007), VMware finally saw their VMI patch based on paravirt ops go into the Linux kernel (patches from Xen are also being applied in this kernel, but Xen still isn't there yet in terms of being fully available inside the Linux kernel).

What will VMware users see when running kernel 2.6.21 and beyond as a guest operating system? Essentially the benefits are twofold: first, and most important, increased speed. In theory, with VMI and hardware virtualization technology, a guest operating system should be able to operate almost as fast as the corresponding operating system would have run natively on the hardware. Second, VMware will be able to exercise greater control over the virtual guest operating system: this means operations like suspend and resume (for VMotion) or administrative operations like real-locating processing or memory resources should be much easier and quicker to perform.

Will users see these benefits immediately? Unfortunately, this might depend on the version of VMware being used. VMware released Workstation version 6 recently, which is capable of fully utilizing the VMI features in Linux Kernel 2.6.21 (although the VMI utilization is listed as "experimental"). However, for the more standard (and non-free) ESX version of VMware, or for the free VMware server product, support for VMI is still only going to be delivered at some point in the future. 

About the Author

James Bottomley is CTO for SteelEye Technology.



The Changing Trajectory of Software

After Ubuntu, Windows looks increasingly bad, increasingly archaic, increasingly unfriendly

by Paul Nowak

My recent switch to a single-boot Ubuntu setup on my Thinkpad T60 simply floors me on a regular basis. Most recently it's had to do with the experience of maintaining the software. Fresh from a very long Windows 2000 experience and a four-month Windows XP experience along with a long-time Linux sys admin role puts me in a great position to assess Ubuntu. Three prior attempts over the years at using Linux as my daily desktop OS had me primed for failure. Well, Ubuntu takes Linux where I've long hoped it would go – easy to use, reliable, dependable, great applications too but more on that later. It has some elegance to it – bet you never heard that about a Linux desktop before.



There are many night-and-day differences between Windows and Ubuntu and, for a guy that does 80% standard office tasks and the rest of the time I'm doing Linux admin tasks, it was nearly all in favor of Ubuntu after the first few weeks of the transition. Overall, my productivity and the scope of things I can do with Ubuntu far exceed what I could do with Windows and just as importantly Ubuntu (like any Linux would) lets me easily create my own productivity shortcuts of a variety of sorts.

One of the things that's become clear as I've gotten used to the appliance-like experience of Ubuntu is that the future of software in an open source-dominated world is going to be significantly different than the world dominated by Microsoft. So what distant point on the horizon has Ubuntu shone a light on for me? Simple. Software will increasingly compete on ease of use in the total software experience more than on features. The future will be more about being simple than about any other dimension.

Here are some recent use cases:

- I needed to rebuild my T60 with a fresh OS. Which was easier? MS Windows with a factory install disk, separate disks for Office and for Virus protection and then a lot of hunt-and-peck downloading for various

apps like Thunderbird, Firefox, SSH, and Calendar or.... Ubuntu with one CD and an OS that includes an integrated, extensible, and slick software package manager where all the software is approved and tailored to the installation?

- I needed to rebuild a T43. I tried to use the rebuild partition included on the HD but it was corrupt. So I tried to make factory-install disks but the corrupt partition prevented it. Next option? Call Lenovo and get disks sent for \$51. That process took five days and eight CD-ROMs from start to finish. With Ubuntu, this process takes three hours max, not four days and there's no software keys or other things to track down. The labor involved is less than a fifth with Ubuntu and the delivered product is a lot more productive – for my use models anyway.
- I needed to resubscribe to Symantec on a Windows machine. Again this is a 30-60 minute timeout from production AND a \$49 charge AND a hassle with product keys and sending data about my machine and purchases around to companies that I'd choose not have it if I had a choice. But I didn't since Windows XP needs Symantec's products badly even though these scanning and cleaning products degrade machine performance badly – even with a gig of RAM.
- And I now hear that Windows Vista renames the partition it's installed on what used to be the C: partition. I need to check out this story but the very idea of automatic partition renaming is insane to even contemplate.

So my machine sings with Ubuntu. Having no virus scanning alone unleashes a responsiveness that makes the power of the T60's Intel dual-core shine. And what am I noticing most about all of this?

Well, first off Ubuntu is good as a productivity platform. Without that, the rest wouldn't matter a bit. But since Ubuntu is not only good on features but reliability then at least some

About the Author

Paul Nowak first used Linux in 1995 while migrating from Sun to Linux at the University of Michigan. He used Linux in subsequent IT projects including web, telecom, telemetry and embedded projects and is currently CIO of a small professional association based in Washington D.C.
paul@naaee.org

Gemini

2U

– Ultra Dense Series



Up to
**40% Less Power
Consumption**

Up to
40% Less Heat

- Two fully independent systems in a 2U
- Ability to run two discrete operating systems in one box
- Up to 8 CPU cores (forward compatible to 16)
- Up to 12 hot-swappable SATA, SCSI, or SAS hard drives
- RAID 0, 1, 5, 6, 10, 50 available on both systems
- Opteron™ or Xeon™ multi-core processors
- Up to 64GB memory per motherboard
- One available PCI-X or PCI-E slot per motherboard



Patent Pending

2 Systems in One

Built on open standards, the Gemini 2U elegantly accommodates two discrete motherboards in a 25" chassis uniquely designed for easy access from the rear.

Gemini 2U represents the realization of intoxicating power and superior environmental specifications, with up to 40% less power consumption, 40% less heat and 30% less noise. Remarkably, it all fits nicely into any standard rack. Front to back, Gemini 2U is both powerful and efficient.

At Open Source Systems we understand you need practical, customizable, and affordable solutions that are easy to manage and maintain.

For more information and to request your evaluation unit today, visit us at www.OpenSourceSystems.com, or call direct at 866.664.7867.



of us would crawl over broken glass to get it installed.

But, in fact, there's no broken glass in the picture. It's the opposite. Ubuntu's installation is so easy, and maintaining it once it's installed is so simple that Ubuntu nearly falls into your machine like a ying to the hardware's yang. Once there, Ubuntu happily makes a home in your head with hardly a blip. I think Ubuntu actually dropped my blood pressure. Not something you typically find when switching ALL your software for something that's about as alien to Windows as it possibly can be.

Once that major hurdle is cleared, then the other big issues come into focus. Ease of install, easy updates, easy software maintenance, easy data backups. After experiencing Ubuntu, the world of Windows looks increasingly bad, increasingly archaic, increasingly like a neighborhood that makes life hard. Why should I put up with what Windows makes me go through if I don't have to?

I've used rsync for backups for years. I back up my mail, my Thunderbird data, and "my document" directory (i.e., /home/xxxx/). One of these backup commands looks like this and sits in a single shell script and runs from cron once a day (I've already sent the ssh key to the backup target server so no need to manually login to the backup server for this command to run):

```
rsync -avgz /home/xxxx/.mozilla-thunderbird/
root@mycomcastipnumber:/hdb/ibmt60-ubuntu-mozilla-tbird/ >>
/home/xxxx/backup-.txt
```

That little command executes in a few seconds to a few minutes no matter where I am on the Internet and even if I've added some decently sized files to my computer. I've got my home router set up to pass the ssh port 22 through to a Linux server sitting in my attic. Quick and painless backups run without a hitch. It's a thing of beauty. I use the same solution for my servers so having a single platform from server to desktop has benefits and this is but one of them. I used to sweat about my Windows backups in the old days – if I did them every two weeks, I was happy. Ubuntu dropped my blood pressure on backups alone by 10% and now I have to decide how often is too often to do a backup. Also, I'm up on the MIRRA product but, trust me, you don't want to forget a password there.

Through a similar setup, I can also print to my home printer from any Internet connection. This is not a Windows- or Linux-specific feature but it's nice to have and I use it more than I expected. This is just good fun but it may also drop my blood pressure a point or two.

So far, none of this is news to those in the know about Ubuntu. It's not news but it is a big deal. A very big deal. Ubuntu is getting rave reviews: it's a productive platform, it's a reliable platform, it's a durable platform, it's an upgradeable platform, it's an easy-to-install platform, and adoption is through the roof.

What's changing in all this?

In my view, once you realize the platform is viable from a daily productivity standpoint (exceedingly so), the #1 thing that Ubuntu is then changing is ease of access to software. If I had decided to rebuild my PC with Windows XP – we won't even talk about Vista – this is what I was looking at:

- 1) Buy OEM Install disks from Lenovo because my rebuild partition was corrupt – \$51.
- 2) Buy a Symantec subscription because I was done with the 90-day free trial – \$49.
- 3) Buy an extra 512MB of RAM because XP couldn't run Firefox, Thunderbird, MS Word, MS Excel, and SSH all at once with 512MB of installed RAM – \$104.
- 4) Install all of the above with product keys along the way – four hours? Maybe six? Maybe more because the tools for getting 2GB-3GB of mail data back into Thunderbird in Windows aren't nearly as good as the same tools in Linux.

That's \$204 just to get me back to where I thought I was two months back – i.e., a machine with XP and Office on it. Symantec alone is going to want to pick my pocket again at some point.

Ubuntu releases me from these costs and from these long-term headaches:

- 1) **Viruses** – I no longer worry and I no longer need to check my PC – that's a relief. You can pick nits here about security but the bottom line is Ubuntu is orders of magnitude better.
- 2) **Vulnerabilities** – Windows is like Swiss cheese with so many vulnerabilities that it's sick – you can't connect XP to a public Internet connection (i.e., behind a router is OK but direct to the net isn't). Ubuntu? It's Linux – no worries.
- 3) Thanks to #1 and #2, I'm free from products like Symantec and Norton and the dollar expense, the complexity of administering them (those pop-ups are annoying and a productivity hit), and wondering when they expire next.
- 4) Software updates for the entire collection of software on the machine are simple in Ubuntu.
- 5) Backups are automatic.


That's batting for the cycle. Am I missing anything? Anything at all? Yes. Printing is

easier in Ubuntu for older printers like the HP Laserjet 4 on a D-Link print server in the office and the HP 6L on an SMC print server in the home office. Multifunction printers are more of a challenge. A little care in printer purchases going forward takes this issue off the plate and I'm fine with the printer solution in place that has largely been stumbled upon.

The one bit of software that was Windows-related was a QuickBooks Timer. I haven't needed it because I began editing the output of that program in Excel six months ago because the QuickBooks Timer was too much of a clod interface to be productive. When I switched from Microsoft Office to OpenOffice on Windows XP, I continued not using the QuickBooks Timer. Doing the same manual editing of these QuickBooks Timer output files in OpenOffice Calc on Linux is a breeze. If there were a QuickBooks Timer for Linux, I wouldn't use it so I haven't checked for it.

In sum, what's changing about software? The installation, maintenance, and use of software in Windows have become a burden. A huge burden. And I don't think the average Windows user realizes how much out of their way they are going to keep their Windows PCs working. Windows challenges users and makes for a very expensive user experience in time and dollars if users follow the book and use the latest virus protection, keep that protection updated, and avoid the pitfalls that are squarely on the path that normal users use. In the best case, you end up with a machine that has a lot of crapware installed on it and is slow and clunky to use. In the more typical case, you end up with a machine that spirals to a grinding halt over six to 12 months – like the T43 I'm working on right now. A machine that has trouble opening an Excel file in three minutes because it has so much software competing for disk access and CPU cycles.

My experience with Linux on the server with its multi-hundred day uptimes broken by hardware upgrades, not software reboots, and with no performance degradation even at high disk utilizations tells me Ubuntu isn't taking me down with it. My blood pressure is truly low now.

I'm literally running out the door to get the word rolling on this changing dynamic. It's that big. And a word to IBM and Lenovo: if you're listening, Ubuntu as an OEM install on your Thinkpad T and X series would be a huge win for you and for the the OSS adoption curve. This is a classic case of experience changing perception and it's got me to thinking about a seamless platform from server to desktop to phone – think about it. 

IT'S THE END OF THE LINE FOR SPAM.



PUT IT IN THE CAN.

Ads, fraud schemes, porn, viruses, phishing attempts...
if it's offending e-mail, Roaring Penguin Software's CanIt™
solutions will stop it before it reaches your mail server.



- Simple administration
- Automatic spam and virus updates
- Compatible with all e-mail servers
- Available as software or hardware appliance
- Flexible end-user controls

Throw the book at spam today!
Request a free 20-day CanIt evaluation
and stop spam on YOUR terms!

ROARING 
PENGUIN
SOFTWARE INC.

Project LASSO for Log Management

Open source software for collecting Windows event logs

by Jon Walker



Recently, I had the pleasure of speaking with Anton Chauvkin, Director of Product Management at LogLogic. We had an interesting discussion about log management and the open source project he's involved in that collects Windows event logs. Here's an overview of our chat.

Drowning in logs is all too common nowadays when organizations are struggling with a combination of operational, security, and compliance requirements. A typical organization will have logs from a wide array of log sources such as server operating systems (Unix and Windows), desktops, mainframes, network gear such as routers and switches, web proxies, security gear such as network IDS, IPS or anti-virus tools, Web, e-mail, and DNS server software as well as enterprise applications.

Large organizations typically have tens of thousands of servers generating log files. The challenge for IT is how an enterprise can efficiently collect logs from all these servers without losing any data. In fact, almost 30% of all enterprise data is log data. Owing to compliance requirements from regulations like Sarbanes-Oxley and PCI, the archived log data must be stored. A single organization can easily be required to store hundreds of terabytes of log data. How IT manages this large set of data continues to be a challenge for enterprises, regardless of size.

Log Management and Intelligence is an approach to dealing with large volumes of computer-generated log messages (also known as audit records, audit trails, event logs, etc.), which consists of log collection, centralized aggregation, long-term retention, log analysis (in real-time and in bulk after storage) as well as sharing the information with the relevant parties across the organization. Such analysis is usually done for security, operational (such as system or network administration), or regulatory compliance.

Effectively analyzing large volumes of diverse logs is a challenge. From huge log volumes – often reaching hundreds of gigabytes of data a day for a large organization – to log format diversity, obstacles in dealing with log data confounds IT daily. Couple that

with undocumented proprietary log formats that often hinder analysis and the presence of false log records found in certain logs, such as intrusion detection logs, and the situation becomes more complex.

To unravel the complexity, tools to handle log collection and analysis are sometimes built by users, assembled from various open source components or acquired from commercial vendors in the form of LMI or Log Management and Intelligence solutions. So far, the open source community hasn't been able to come up with a single tool to deal with most log challenges that confront IT. But there are some promising contenders.

Moreover, the open source community has been pretty effective in building pieces of log management infrastructure. Syslog-NG enables log collection from Unix servers and network devices, serving as a better replacement for standard syslog daemons than is typically provided by operating system vendors as a primary example of open source excellence. There are also a huge number of simple scripts and small programs such as logwatch, logsentry, and fwanalog that were written by the open source community over the years to handle specific logs or a particular slice of a log puzzle. At times it seems that it was easier for some people to create their own script instead of looking for one online. However, most of these tools focused on Unix and Linux platforms and largely ignored Windows-based systems.

One of the recent open source solutions that enables a critical part of log management is Project LASSO, a Windows-based open source software designed to collect Windows event logs, including custom application logs, and provide for the central collection and transport of Windows log data via TCP syslog to any syslog-NG compatible log receivers. Before Project LASSO incorporating Windows

About the Author

Jon Walker serves as CTO of Versora, an ISV providing Microsoft to Linux migration software. Mr. Walker recently has co-authored 2 whitepapers with Novell titled *Migrating from IS Web Servers to Apache SUSE LINUX Enterprise Server 9.0* and *Migrating File and Print Servers from Windows to SUSE LINUX Enterprise Server 9*. Prior to Versora, Mr. Walker was CTO/VP of Engineering for Miramar Systems. Software developed under his direction at Miramar has been deployed to over 20 million computers worldwide. Mr. Walker has also served as senior technologist for Nortel and Xing Technology (now Real Networks).
jwalker@versora.com

server and workstation logs in an overall log management process was extremely onerous. One had to use agents installed on every single Windows system to collect the logs or be stuck with super-expensive proprietary solutions. And deploying agents on every system is one of the most dreaded tasks in all of enterprise IT.

Open source tools such as syslog-ng existed for years to simplify log management for Unix and Linux as well as network devices that support syslog (such as Cisco routers and firewalls), but the Windows part of the world was largely excluded because binary Windows event logs aren't syslog. Project LASSO bridges this gap, allowing remote Windows log collection without agents, as well being deployed as an agent on each server, if needed. LASSO enables the inclusion of logs in log management systems, such as the one by LASSO's sponsor LogLogic or other companies.

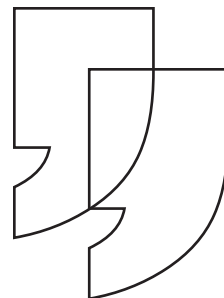
Overall, Project LASSO enables users to connect the dots by allowing central collection and analysis of Windows event logs with the same ease that they are used to with Unix and Linux. After the data is collected by LASSO, users can use report and search features to review and analyze logs across all the systems in an enterprise: Windows, Unix, network

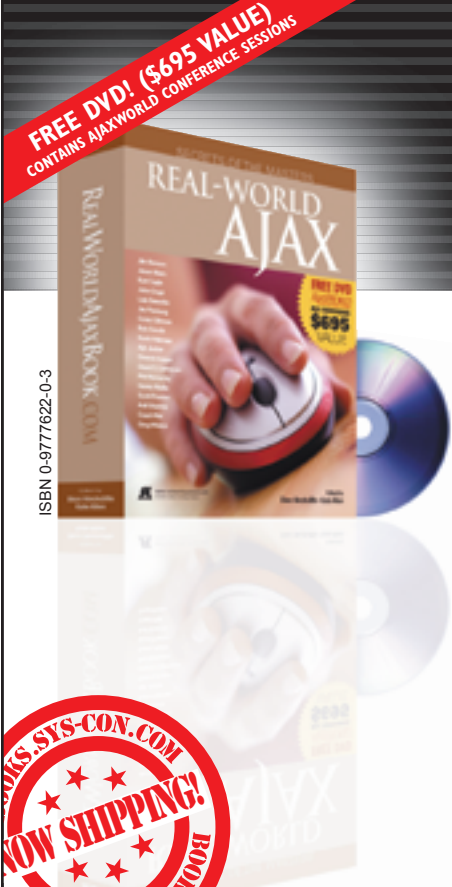
systems, applications, etc. Moreover, LASSO greatly reduces the impact on monitored servers in terms of storage and processing, as well being able to capture application-specific and custom Windows event logs.

Using LASSO, IT can gain invaluable insight into its network. For example, a query for an account holder can be run across all the systems in an enterprise, identifying the files or applications that he or she touched. Such capability is critical for compliance, as well as for incident response and forensics.

Log Management is increasingly making its way onto the IT agenda. Today, a simple Google search of "log management" drives this home with over 240 million hits – and it's growing daily. As more organizations move toward implementing policies for compliance, log management systems have taken on a vital role. LMI's greatest value lies not only in the improvements it creates in automating compliance and providing forensics, but there are great benefits to be found in ensuring operational efficiency by giving IT visibility into the details of what has happened on every system in its network. As Log Management and Intelligence matures, the open source tools that intersect with log files will surely continue to evolve and mature. 

Before LASSO incorporating Windows server and workstation logs in an overall log management process was extremely onerous





ISBN 0-9777622-0-3

JOIN THE AJAX REVOLUTION!

Secrets of the Masters: Real-World AJAX


Edited by Dion Hinchcliffe & Kate Allen


"If you're looking for a one-stop shop for an AJAX book...you would have a long search to find a better overall resource than what you find in the chapters of this book."

Order Online at RealWorldAJAXBook.com and get

40% OFF

Regular Bookstore Price!

 books.sys-con.com
from the World's Leading i-Technology Publisher © COPYRIGHT 2007 SYS-CON MEDIA



The Vision for Eclipse

Interview with Mike Milinkovich

Interview by Joe Winchester

Mike Milinkovich, executive director of the Eclipse Foundation, has been kind enough to answer some questions for Enterprise Open Source Magazine. Rather than rattle off the usual ones about the name, about why Swing wasn't used, or how much influence IBM still has, Mike has fielded questions on some more current and topical subjects, as well as given us his insights onto the future. Thanks for taking the time to talk to us Mike.



Mike Milinkovich
Eclipse Foundation

EOSM: The Eclipse Foundation recently joined the Java Community Process. Can you tell us how this is going and what you expect to get out of this, as well as give to the JCP?

Mike Milinkovich: Yes, we recently joined the JCP, as we also joined the OSGi Alliance and OMG. The reason for joining these organizations is that the Eclipse community relies heavily on the standards that are produced by these standards organization, so we wanted to show our support.

As for the JCP specifically, we are planning to contribute in a couple of different areas, the most immediate example being the use of the Eclipse Equinox code as the reference implementation for JSR291.

EOSM: At this year's EclipseCon I felt that the amount of interest in RCP had surpassed the amount of interest in the actual IDE. Do you think this is the case, and if so does this change the dynamics of Eclipse's strategy and direction to become more of a general-purpose application platform and less of a development environment?

Milinkovich: Yes, I agree the Eclipse community and the industry as a whole has moved toward viewing Eclipse as an application platform. We are seeing a lot of interest and adoption of RCP and also projects such as Equinox, RAP, and Higgins. However, this is not new as we have had a conscious strategy to move Eclipse beyond just being a Java IDE for several years now. I believe what we are seeing is quite simply that a number of the newer projects within the Eclipse community are becoming more mature and are enjoying greater interest and adoption as a result. The

vision for Eclipse has always been about being a complete platform for software development and I think we are well on the way.

However, I do continue to see a lot of interest in Eclipse as a tools platform. We have new projects for providing IDEs for dynamic languages such as Ruby and PHP. The AJAX Toolkit Framework (ATF) is attracting a lot of interest as a tooling platform for AJAX developers. CDT, our C/C++ IDE, has great momentum as being the C/C++ IDE for embedded and Linux development. In my opinion, Mylar is one of the most innovative new developer technologies to come about in a long time.

Therefore, I don't really see a large change in strategy or direction; I see it more as a journey and evolution. This is what makes Eclipse such a vibrant and interesting community.

EOSM: JSR 291 ratified the OSGi Java module mechanism used by Eclipse to become part of the Java language specification. Can you see the same occurring with SWT?

Milinkovich: I haven't seen any interest from the community in putting SWT into the JCP process. It is not that I don't think it would be an interesting idea but someone would have to step up to spend the considerable amount of time required to take it through the process.

EOSM: How do you manage the relationship between the Eclipse board members, some of whom are fierce competitors in the commercial marketplace, yet need to collaborate for the good of the platform?

Milinkovich: Interestingly, this has been mostly a non-issue to date. I've been really very happy with how collegial and effective the

About the Author

Joe Winchester, JDJ's Desktop Technologies Editor, is a software developer working on development tools for IBM in Hursley, UK.

joewinchester@sys-con.com

Board of the Eclipse Foundation has been. That said, I think you have already identified the answer; there is a common need for a strong, stable platform for building all sorts of different software. Eclipse is providing this platform and it really becomes a unifying force at the Board and through-out the community. The other thing is that the Eclipse governance model is proving to be very good at managing the different interests that participate in Eclipse. All the organizations have an equal say at the Board level and the principles of meritocracy and transparency help resolve a lot of the issues within the projects.

EOSM: What excites you most about what is going on with Eclipse at the moment?

Milinkovich: Europa, our next release, is going to be pretty exciting. We have over 22 projects lined up for the annual release train, so a lot of new stuff is getting ready for release. Our annual release trains are very important for the entire Eclipse community. First of all, they are a real testament to the committer and project community's ability to deliver on a predictable schedule. Second, they are hugely important to our adopter community as they use the projects to deliver their commercial products or open source projects.

I think the growing involvement of Eclipse in the world of Equinox-based OSGi runtimes – such as the EclipseLink persistence project recently proposed by Oracle – is cool as well. Over the next 12–18 months we are going to see a lot of new stuff being built with Equinox.

EOSM: What worries you most about what is going on with Eclipse at the moment?

Milinkovich: I'm not sure that "worried" is the right word, but I would like to see vertical market frameworks (e.g., banking frameworks, health care frameworks) being developed as Eclipse open source projects. This will require enterprises to become more involved with contributing to open source projects and it is something that I think will take time. We have a great start with the Eclipse Open Healthcare Framework (OHF) but we need to do more to encourage large enterprises to begin collaborating in open source projects.

EOSM: NetBeans seems to be gaining a lot of traction at the moment, especially with some of the emerging and BRIC markets. What is Eclipse doing in this space to keep up?

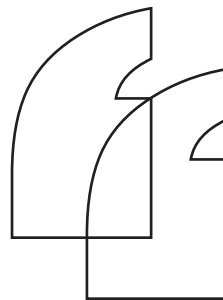
Milinkovich: This is a funny question since we are finding the BRIC countries to be our highest growth areas. Our downloads from China are just exploding; Evans Data recently

reported our usage grew 30% in India and 20% in Brazil. It is nice to hear that NetBeans is doing well but I certainly don't feel like we need to keep up. In fact, I believe we are leading in the BRIC countries.

EOSM: There is talk that Eclipse 4.0 will be a major rethink of the platform and how it is put together. Can you share with us what's going on here?

Milinkovich: To be honest, everything about Eclipse 4.0 is conjecture at this point. EclipseCon was the first time the idea had even been discussed by the community and not only are there no concrete plans yet for 4.0, there isn't even yet a firm commitment that it is going to happen. Of course, I think it will, but it is definitely too early for me to speculate what may or may not be in it.

What is going on at this point is the beginning of a process – a community process to decide whether doing a major new release – quite possibly with some API breakages – is the right thing to do for our broad ecosystem.



It is my belief Eclipse has become a place where many organizations can **collaborate** and **innovate** on interesting new technology.

EOSM: If you could do Eclipse all over again, what would you do differently and why?

Milinkovich: First, please remember, I did not "do" Eclipse. A lot of people spent a considerable amount of time and energy establishing Eclipse long before I appeared on the scene in 2004. I have just been lucky enough to be involved in the implementation of the vision they set out.

Things are working remarkably well, so I am not sure I would change anything. It is my belief Eclipse has become a place where many organizations can collaborate and innovate on interesting new technology. I truly believe Eclipse has the right technology, architecture, and governance to make it an important platform for future software innovation for many years to come. 

Open Source, Open Integration, and the Open Solutions Alliance

Hyperic & JasperSoft Present a Case Study-in-the-Making

by Stacey Schneider

When the Open Solutions Alliance (OSA) launched in February, there was a lot of interest. But some of that interest has been industry head-scratching, wondering about the results and the intentions of the OSA (www.opensolutionsalliance.org). Noted open source business blogger Matt Asay was among the skeptics in his post "Yet Another Alliance (OSA)" (http://weblog.infoworld.com/openresource/archives/2007/02/yet_another_all.html). He asks three pertinent questions, mostly about the OSA's purpose.



1. Is the alliance about getting people to buy into open source?
2. Is the alliance about getting quick marketing or paper partnerships for vendors?
3. What does this have to do for customers?

Matt's first observation is emphatically correct; there's no denying that open source is booming. Open source business models are changing and adapting and customers are flocking to it for reasons other than those that make it good reading on a balance sheet. The software is quickly accessible with demonstrable results for the business often in the same timeframe it used to take to go through the arduous process of buying the proprietary closed source software. Projects are open and flexible, typically bringing to market solutions faster than their proprietary alternatives. Much faster, in fact. This is partly because of the community, which helps to focus the company on market priorities through active communication and collaboration. It also helps deliver the results. In general, using and extending that software is done through the community, which brings a powerful knowledge base and easy access for all documentation in the community forums and wikis.

Working directly with an open source project is easy. Working with multiple vendors at the same time, however, is not. When embarking on a new business project that re-

quires software, the people involved typically ask three key questions:

- Has anyone done this before?
- Does it work with our current systems?
- How should we go about this?

Answering these questions quickly and correctly is critical to customers AND vendors. This is the founding principle of OSA that Hyperic bought into. As a company, Hyperic writes software that manages other software. By nature, everything we do has some level of integration. It's not always easy. Through the proliferation of open source and technologies in general, there's no standard "stack" on which to build a software solution. You're free to choose J2EE, one of the 20 or so permutations of LAMP, commercial closed source alternatives, or a hybrid mix of all of them! Whatever you do, you're going to want to integrate and manage all of that software.

A Case Study Evolving

Hyperic (<http://www.hyperic.com>) meets the demands of the marketplace by providing the visibility and tools needed for operational IT workers to diagnose and correct problems quickly, improving performance and availability. It does this so well that the business folks in the glass offices are paying attention. They want to see summary reports detailing the overall availability or downtime of the systems, the average outage or performance episodes, and the percentage of downtime planned versus unplanned. They want operational performance reports on the level of optimization in their virtualization deployment. In short, they want comprehensive reporting.

When reviewing the capabilities of open source projects, we decided that JasperServer from JasperSoft (<http://www.jaspersoft.com>) not only met but exceeded our reporting requirements. As a bonus, they're also part of OSA. But that's not the reason why we elected

About the Author

Stacey Schneider is senior director at Hyperic and Member of the Open Solutions Alliance.

ENTERPRISECLASS

Robust security. Fully scalable.
Java/J2EE environment.
Supports leading SQL databases.
In use today by Fortune 500.

ADVANCEDDESIGN

Best-of-breed CRM modules.
Runs on all standard-based platforms.
VoIP integration.
Customizable workflows.

OPENSOURCE

Source code freely available.
Flexible license.
Freely extend product capabilities.
Integrate with existing systems.

CUSTOMERMANAGEMENT

Manage sales pipeline end-to-end.
Leads management. Sophisticated
Help Desk. Custom surveys and
marketing communication.

No Matter the Weather, The Weather Channel® Is Open



The Weather Channel is open for business 24/7. When they decided to make their IT systems open as well, they turned to **Centric CRM**, the most advanced Open Source CRM system available. **Centric CRM** gave The Weather Channel's sales and support teams access to data from their patented technologies housed in 10,000 cable headends across the country. The result? Reduced total cost of ownership (TCO), improved customer view, and the elimination of multiple proprietary applications. Regardless of the conditions outside, **Centric CRM** helps The Weather Channel stay open.

Find out how Centric CRM can help your business get open:

Visit www.centriccrm.com or Call **757.627.8977**

Set up a free demo system / Become a Centric CRM Partner



Brian Shield
CTO, The Weather Channel

“Centric CRM is a key part of The Weather Channel's move to Open Source software, allowing us to mothball expensive, proprietary applications. Centric CRM is reliable, low cost and easy to use, adding feature rich functionality and flexibility to the way critical data is managed.”

Because Great Minds Are Open

 **CENTRICCRM**
Built for the Enterprise...Open to All

to work with them. That said we're each motivated to leverage OSA to speed our integration and invite the public to participate.

To start the project, we did our own vendor selection. Perhaps in time, this process will be good to open an integration-savvy community. But as OSA is just getting started, we went through that process privately; however, this is exactly the kind of effort OSA will help facilitate and foster for the benefit of business users in the future.

Based on community requests, we decided that, for Hyperic users, availability reports were most important. We started there. We met with the JasperSoft team and completed intensive reviews of our respective infrastructures, data schemas, and APIs. We discovered in the meeting that the JasperSoft community had been asking for better manageability.

A perfect match. And so two projects were born:

1. Embedding JasperReports in Hyperic HQ and
2. Building a Hyperic HQ plug-in for JasperReports

We published the requirements and announced the projects to our communities offering to include anyone who wanted to help. We also agreed that we would publish the lessons we learned and insights we had for standards or best practices on OSA's forums, and asked our communities to review the integrations between our projects on OSA directly. We identified three key areas to review:

- **UI Integration** – Embedding the JasperReports UI into Hyperic's to become a seamless experience and maintain a level of interactivity will require a significant amount of work. Aside from fusing our UI frameworks and widgets there needs to be a standardization of style sheets that maintain a consistent look and feel. There are also localization requirements as we try to isolate the application architecture for quick translation into other languages.
- **Single Sign-On** – Fusing user navigation into a single experience and dictating permissions collectively is a challenge for any software integration solution. The requirements for our projects will be bidirectional. Hyperic open source users will only need a single login to both Hyperic and JasperSoft

solutions. Since our enterprise extensions are extensions to the open source project, Enterprise customers have requirements for permissions for information they can see, change, or act on. Single sign-on needs to be solved for universal access, but must accommodate permissions as well. Hyperic HQ will also need universal access to the complete JasperSoft infrastructure, log, and configuration files to provide management.

- **Management APIs** – Reporting applications aren't known for management standards, since they're not generally considered paramount for transactional systems. After all, the transactions may experience errors or problems, but the reports generated may just be delayed or skipped. However, these reports are increasingly offering serious analysis into general transaction performance and can affect the overall result. Just think of Hyperic's use case, where we'd use these reports to substantiate the required availability SLA of a virtualized server farm. Important insight into the transactional system could improve reporting system availability and performance. Requirements for database performance including query optimization and ETL transformations. Data export/import will be essential.

Not a Corner Case

Most, if not all, of these requirements aren't unique to a systems management/reporting software integration story. In fact, they're very common. We're not going to be the first to discuss these issues. We are, however, going to be among the first projects to discuss our open issues, best practices, and lessons learned in the OSA's open forum created exclusively to speed the adoption and integration of software solutions with open source components. We believe that the best route to foster adoption is by centralizing a knowledge repository on the subject to assist the community at large in their efforts to do similar integrations.

So, to answer Matt's questions more formally:

1. Is the alliance about getting people to buy into open source?

No, it won't drive them to Open Source; that hasn't been a problem for a while. It may however, help prospective adopters

make informed choices about how open solutions will "plug and play" in the context of their current IT ecosystem. Mostly, it will help accelerate the implementation and time-to-value of their open solutions choices by offering help on common integration challenges.


2. Is the alliance about getting quick marketing or paper partnerships for vendors?

No, although there's some level of extra brand awareness as a by-product of participating. But frankly, participating in any open discussion gets you the same result. JasperSoft and Hyperic as companies didn't have any extra incentive to embark on this project together via membership in the OSA. We decided to do this based on regular due diligence of available projects and a best-fit analysis. Membership in OSA gives us no business or legal incentive to work together. If anything that comes from a common use of the GPL and has nothing to do with OSA membership or provisions. It does mean though that we're both motivated to contribute to this new community. Contributions are an important facet of any open source project since they keep the community alive and build mind share and credibility.

3. What does this do for customers?

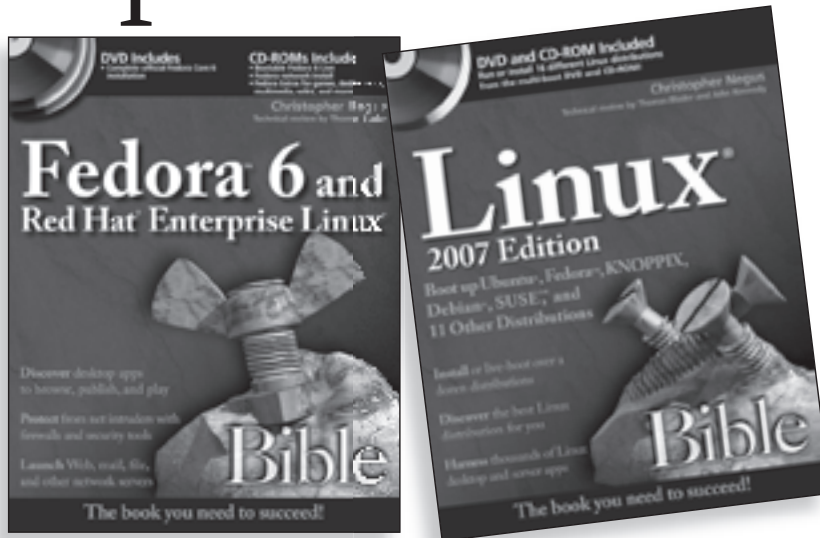
It will help provide a common ground for integration discussions and solutions that will help improve and speed integration efforts essential to every IT project out there.

The age of computing has evolved into an age of integration – integrating technology not only with itself but with business and everyday life. With the sponsorship of vendors and the involvement of the community at large we believe that the OSA will go a long way toward increasing the speed of integration, adoption, and deployment of open source technology, and we hope, ultimately increase the pace of innovation a little bit more.

Two months into this alliance and we're making advances toward our goal – and not just Hyperic and JasperSoft. For more information on all the projects under development on the OSA forum and to watch the developments of this one, see <http://www.opensolutionsalliance.org>. 

When Chris Negus speaks, people learn Linux!

Wiley and the Wiley logo are registered trademarks of John Wiley & Sons, Inc. Linux is a registered trademark of Linus Torvalds.



978-0-470-08278-2

978-0-470-08279-9

Get the latest Linux Bibles by Christopher Negus.
The books you need to succeed.



Looking to Stay Ahead of the *i*-Technology Curve?

Subscribe to these
FREE Newsletters >

Get the latest information on the most innovative products, new releases, interviews, industry developments, and *i*-technology news

Targeted to meet your professional needs, each newsletter is informative, insightful, and to the point.

And best of all – they're FREE!

Your subscription is just a mouse-click away at www.sys-con.com

ITsolutions

NET JOURNAL

JDJ

WebServices

Information
STORAGE+
SECURITY
journal

LinuxWorld
MAGAZINE

wireless

LINUX BUSINESS
WEEK

XML JOURNAL

MX
developer's journal

WebSphere
JOURNAL

wldj

ColdFusion

SYS-CON
MEDIA

The World's Leading *i*-Technology Publisher

It's Not Just the Code

What open source has to offer human resource management

by Sujee Saparamadu

After infiltrating corporate operating systems and middleware products, open source software has in recent years moved into enterprise applications with great success.



Within Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM), many companies have enjoyed a blossoming of open source software. By harnessing the collective wisdom, experiences, expertise, and requirements, the model has ensured that needs have been rapidly met while the open source code has provided the wanted flexibility and low cost. Everything made possible through the Internet, which has enabled easy zero-cost distribution and improved global collaboration. By leveraging large user communities the software products have grown fast and made it possible for open source companies to challenge major proprietary vendors.

Up till now Human Resource Management (HRM) has been an area of development and use of open source software overlooked by most. This seems due to existing barriers and special aspects that have made entering this area troublesome and have kept open source HRM systems from penetrating mainstream IT. However, if these barriers are passed and the special aspects are kept in mind when developing a system, open source software and HRM is a great match and the benefits for HRM are numerous.

Let's first briefly look at the main factors that have kept open source software from breaking through to HRM. What has kept many companies from looking the way of open source software has often been the fear of troublesome installation, user unfriendliness, and lack of support.

The users of Human Resource Management are typically non-IT professionals with little or no knowledge of the more technical standards and specifications of the software. They simply demand a well-featured product, easy to install and use. A focus on user friendliness is therefore crucial when developing an open source HRM system. So is community

management. Having a large and active community shows the sceptical user that he/she can get support if needed and that an effort is being made to improve the product. Last but not least, when developing HR software, supreme programming skills alone are not enough. A clear understanding of the human resource field is essential.

It will be a constant challenge, but if these things are managed properly, HR professionals will benefit from open source just like other individuals and companies are benefiting because of successfully handled projects.

The Benefits

So exactly how can HRM benefit from open source? Many of the benefits are closely related to the user community and collaboration around an open source HRM system. Developing and managing this community is crucial as I said. This continuously evolving collective knowledge base holds a positive consequence: speed of innovation. When delivering free open source software the number of users can grow fast. And because of the speed with which users, developers, and companies can post bugs, feature requests, documentation, and patches the product life-cycle will be shortened considerably. Through this model functionality is constantly being improved and product stagnation avoided. And because money isn't wasted on expensive sales and marketing campaigns, more money and resources can be put into product development.

Evaluating products is easier with open source. Many proprietary vendors make the user sign agreements that prevent him/her from publishing benchmarks on their products. No such requirement will be imposed by open source vendors. There's no interest in suing the community of users, stakeholders, and partners who can help improve the product.

About the Author

Sujee Saparamadu is CEO of OrangeHRM, Inc., a leading provider of open source human resource management software used by small and mid-size enterprises worldwide. The highly flexible OrangeHRM system is backed by a fast-growing and knowledgeable open source community and is available with support and services under the professional open source model. OrangeHRM on-demand can be used under a 30-day free trial. An open source version can be downloaded at www.orangehrm.com.


Another benefit is improved flexibility and possible customisation. Because no two HR departments are created equal, the consensus is that HR software must be customisable. Companies and HR departments throughout the world have different standards and procedures and the open source code approach makes the system more flexible. Anyone is allowed to modify the code for a system to match the specific needs of the company. Furthermore the approach gives control over the HRM implementation that proprietary closed source HRM restricts and the user avoids vendor lock-in.

Open source also enables better possibilities for support services. Any software or consultancy with the expertise, the resources, and the will can elect to provide whatever level of support it feels the market will bear. With complete access to the source code nobody has the edge that was previously only available to the owners of the proprietary software. Support is now open to competition and the quality of support is open to market forces. This will also be a benefit in a way that it makes space for local people with clear

knowledge of local business standards and culture to take the task on providing support for the products.

It's been said many times before, but it's worth mentioning again. A major benefit of open source software is the financial impact. Many SMEs simply can't afford the price of proprietary HRM software. Open source gives every company the right and possibility to have its own HRM system.

The financial benefits are often what begin the exploration of open source software. After adoption the primary benefit might shift to flexibility or simply the acknowledgement of the products' high usability and superior quality. In the near future this will also be the case for open source HRM systems. Open source software is neither a religion nor an ideology. It is however a very pragmatic way of developing software in today's rapidly changing environment. An important question often asked by HR professionals and their managers about a new system is: "Will this software make or save us money?" Developed and managed properly, open source HRM software will do both. 



ISBN 0-9777622-2-X

BUILD RICH INTERNET APPS!

Rich Internet Applications with Adobe Flex & Java

Written by Yakov Fain, Dr. Victor Rasputnis and Anatole Tartakovsky


"The authors have been key contributors to Flex's success via their participation in our beta programs, posts to community forums, public presentations, and blog postings... There's a lot to learn, but Yakov, Victor, and Anatole have done an excellent job introducing you to everything you need to know to build a robust application."


— Matt Cbotin, Adobe, Product Manager

Order Online at TheRIABook.com and get

40% OFF

Regular Bookstore Price!

 **SYS-CON BOOKS** books.sys-con.com
from the World's Leading i-Technology Publisher © COPYRIGHT 2007 SYS-CON MEDIA



The Evolution of Enterprise Applications: Open Source, Common Platform

Moving at the speed of business

by Steve Yaskin

According to study by the Burton Group, the next disruptive technology in the enterprise business solutions will come from the Google Enterprise Applications portal (Burton Group: *The Disrupter: Google Enterprise Apps* by Guy Creese, Senior Analyst, Collaboration and Content Strategies, March 19, 2007).

Global companies, as well as individual users, will be able to shop online a-la-carte style and choose the applications they need. Integration will be a thing of the past, as all of the applications will run on the same Google GWT platform. This is what makes software giants like Microsoft and SAP so worried about the future of enterprise software development. Everyone is afraid to miss the next big thing and is trying to capitalize on emerging trends. When enterprises do start leveraging their new technology and paradigms, large vendors want to be ready. Microsoft has made advances toward Yahoo! while SAP is investing heavily into portals. But Google is gaining steam, and other “traditional” software powerhouses are forced to play a catch-up game.

Microsoft, Oracle, and SAP are old-school companies based on a traditional, closed-core and proprietary technology. As with any new and disruptive technologies, industry giants seem to be all over the open source phenomena. However, this time it might be impossible for large vendors to adopt the new thinking and embrace it in order to capitalize on it and turn it around into the same old philosophy. While traditional vendors invest heavily into “embracing” open source, their essence remains the same: none of these companies will be willing to release their source, which contains millions of lines of legacy code that for many years has been dragged from one release to the next. They will also risk making their enterprise customers (who paid hefty for the product licenses) majorly upset. The open source philosophy and approach to the software overall is prov-



ing to be the next big thing, which will shake off older vendors.

A new cluster of startups is emerging in the open source arena that offers business applications based on common and expandable open source platforms. Google recently announced plans for the development of Google Enterprise portal and a new development framework on which such applications can be built – Google Web Toolkit (GWT). Some open source companies immediately took note of this and recognized the potential in partnering with Google on the open source front, and in delivering enterprise-grade applications under Google’s umbrella. These companies are the future of enterprise solutions that one day, in the not-so-distant future, will dominate the market place.

Open for Business

Open source is not a new term and has been around for years. Companies like Red Hat, JBoss, and MySQL have been successful in providing open source back-end and middleware solutions and making money by providing services and support for their products. The benefit to the consumer has been realized in savings on license fees and the ability to maintain applications’ infrastructure on their own. Most of the time, companies running their infrastructure on open source platforms have been more efficient with less downtime, even though it requires more technical expertise from the end customer.

In recent years, however, the open source movement concept has experienced tremendous growth and adaption in the enterprise. It expanded from middleware and databases

About the Author

Steven Yaskin is CTO of Queplix Corporation, which offers a suite of software tools and professional services to help companies deliver customer care with contact centers, help desks, and eService. Its flagship product, QueWeb, is a commercial open source solution used worldwide by leading corporations such as John Deere, Sony Ericsson, Hewlett Packard, Telcordia, and Norfolk Southern. (www.queplix.com)

to companies offering open source business applications. Examples are numerous: SugarCRM offers sales force solutions; Compiere offers a broad range of open source applications from ERP to virus protection software; Alfresco offers content management solutions; and Quepox offers a wide range of open source solutions for the customer care industry.

Licenses, Liberty, and the Pursuit of Openness

Contrary to popular belief, the biggest benefit of open source is not that it is a “free-for-all” solution. While some solutions are indeed offered to the public for free with no strings attached, these solutions may bring limited value to the final consumer (since the developers working on them probably have full-time paying jobs that take priority over these side projects).

Many open source companies are trying to figure out how to combine the community aspect of open source with the need to make a profit. Some have been experimenting with numerous open source licenses, from BSD to GPL, to many variations in between. It's important here to recognize that perhaps the biggest benefit of open source is not that it's “free,” but that it shows a customer what he or she is buying. When you purchase a closed-core proprietary solution, what you see is what you get. While you can estimate that the solution will do about 60–80% of what you need, you won't have any idea of what it is really capable of until you install it and run it for several months. Will it integrate with your other business systems? Will it synchronize with your billing, accounting, analytics, and knowledge management solutions? Will it be able to work with numerous databases? Will it talk to your middleware infrastructure? Will it be able to sustain successful upgrades of numerous components that it depends on? Will you be able to customize it and make sure that it stays on top of your constantly changing business? When you buy proprietary software, the answers to these questions are as “black box” as the software itself, no matter what the vendor is trying to make you believe.

But is the situation any different when you buy an open source solution? In order to


make sure you can resolve these questions if you decide to go the open source route, it's important that your vendor can answer “yes” to the following key questions before you write a check:

- Are there any large corporations and enterprises that run your software to support their critical business functions? (Ask for references. Talking to actual users is the most reliable way to know how dependable, scalable, and robust the application really is.)
- Is your software based on standard enterprise technology (such as J2EE)?
- Is detailed technical information and documentation available?
- Is there a large community of developers that supports this application?
- Is there extensive training available?
- Does the open source vendor provide support for the solution? What are the support and maintenance structure and costs?
- Are new versions of software included with the support and maintenance?

Subject to many variations, licensing terms are particularly important when dealing with open source utilities – so be sure to study the licensing agreement and learn what kinds of things you are and are not allowed to do with the software you choose. Make sure that the license allows you to customize, support, and use the software internally without limitations, so you have the freedom to disassociate yourself from the vendor at any point in the future. After all, not being locked in with the vendor is one of the greatest benefits of open source software.

The Need for Speed

If you are looking to deploy an enterprise application – whether it's for customer care, content management, or a range of other capabilities – open source solutions should definitely be on your short list. A look at the industry shows that these solutions are quickly outpacing traditional software applications, offering significantly lower TCO and the ability to adapt (among many other benefits) – ultimately allowing your company to move at the speed of business.

The final question is: How fast do you want to go? 

The Freedom to Innovate

Interview with Emma McGrattan

by SYS-CON.TV

This article is an excerpt from a SYS-CON.TV Webcast with Emma McGrattan from Ingres Corporation.



SYS-CON.TV: I'm very pleased to introduce Emma K. McGrattan from Ingres Corporation. Emma is responsible for the development and integration of the Ingres database and associated products and technologies; a leading authority in DBMS and open source technologies, Emma's been instrumental in the ongoing success of the Ingres product line. Born in Ireland, she earned a bachelor's degree in electronic engineering from Dublin City University.

McGrattan: For those who aren't familiar with Ingres, let me outline where Ingres came from and what the technology is all about. The Ingres technology has been around for quite some time. It dates back to the early '70s at UC Berkeley, and was a project started by Michael Stonebraker, Eugene Wong, and a number of other very talented folks at Cal Berkeley. They are essentially perceived as the fathers of relational database technology. In the '80s, Ingres went from being an open source project, available under the Berkeley license to anybody who had the money for a source code tape, to a commercial entity, a company called Relational Technology, Inc. The name was later changed to Ingres Corporation. It was thought it was easier to market the product and the company under a single brand. In 1990 it was acquired by a company called ASK Group and in '94 by Computer Associates, so it has had a number of owners over the years. In 2005, we took Ingres out of Computer Associates and set up a company that we also called Ingres Corporation and we're focused on providing support and services around the technology. It's an open source technology, and it's been very well proven in the field. It has over 10,000 customers in 58 countries who, on a daily basis, put the product through its paces in mission-critical and business-critical deployments.

Since then we've been trying to grow a development community around Ingres. We're very focused on the application development community because that's where we traditionally had

a very strong presence. People have for decades been building applications to deploy against Ingres and the enterprise, and what they find is that the same Ingres you put on your desktop to build an application will scale all the way up to powering the data servers in the data center.

Up until now we have supported a wide variety of application development environments; now we're focusing on the ones we view as very important and that provide a lot of productivity and preferably are open source solutions that can be developed by the community. There's a lot of passion built up in development communities, and certainly a development community that's targeted at building development tools is ideal. Eclipse is a huge community looking to build an application development environment that allows you to be very productive, to innovate, and to forget about what's happening behind the scenes and focus on what's important, which is developing the business logic code.

Ingres has become a part of the Eclipse community and is looking to join the Eclipse Foundation and to become committers on various Eclipse projects related to database platforms, to modeling, and things of that ilk. Eclipse is a perfect match for Ingres Icebreaker, which is our latest innovation in terms of delivering a database appliance that's a combination of the Ingres database and the R Patch Linux operating system. A single unit, it contains both the operating system components required to support the database and the database itself. The beauty of this is we provide an integrated maintenance stream, so you don't need to think about which operating system patches are a prerequisite for installing database patches, and so on. The entire maintenance stream is integrated and can be automated to a level you're comfortable with.

The following is some background as to what Ingres views as driving the software market dynamics today. Back in the '70s and the early '80s, a lot of companies were building their own data platforms, a lot of companies built their own

“Businesses that ignore the potential of SOA will find themselves outpaced by rivals who improve their agility and transform themselves into new kinds of enterprises

— Yafim Natis, Gartner Analyst

3-DAY EVENT!

SOAWorld

Plus **2007**

Enterprise OpenSource Conference & Expo 2007

TOPICS INCLUDE:

SOA Web Services

- > AJAX and SOA
- > Web 2.0
- > Universal SOA
- > Protecting Web Services
- > Troubleshooting SOA
- > Governance
- > Open-Source SOA
- > XBRL
- > Service Virtualization

Open Source

- > Open Source Business Models
- > Open Source ESB
- > OpenAjax Alliance
- > SaaS and Open Source
- > Spring, Hibernate and Eclipse
- > Seam
- > Open Source Penetration
- > Monetizing Open Source
- > Open Source Databases
- > AMQP
- > Open Source Middleware

June 25-27, 2007

Roosevelt Hotel / New York City

Register Online! www.SOAWorld2007.com

11th International
SOAWorld
CONFERENCE & EXPO

2007 is to many industry insiders shaping up to be a major inflection point in software development and deployment, with SOA, Web Services, Open Source, and AJAX all converging as cross-platform and cross-browser apps become the rule rather than the exception.

Accordingly the 11th International SOA Web Services Edge 2007 again seeks to offer comprehensive coverage and actionable insights to the developers, architects, IT managers, CXOs, analysts, VCs, and journalists who'll be assembling as delegates and VIP guests in The Roosevelt Hotel in downtown Manhattan, June 25-27, 2007

Co-located with the 2nd Annual Enterprise Open Source Conference & Expo, the event will deliver the #1 i-technology educational and networking opportunity of the year. These two conference programs between them will present a comprehensive view of all the development and management aspects of integrating a SOA strategy and an Open Source philosophy into your enterprise. Our organizing principle is that delegates will go away from the intense two-day program replete with why-to and how-to knowledge delivered first-hand by industry experts.

Visit soaeosconference.sys-con.com for the most up-to-the-minute information including... Keynotes, Sessions, Speakers, Sponsors, Exhibitors, Schedule, etc.

2nd Annual
**ENTERPRISE > 2007
OPENSOURCE**
CONFERENCE+EXPO

SOAEOSCONFERENCE.SYS-CON.COM

REGISTER ONLINE TODAY

SAVE \$200!

(HURRY FOR EARLY-BIRD DISCOUNT)

BROUGHT TO YOU BY:



» **SOA World Magazine**
focuses on the business and technology of Service-Oriented Architectures and Web Services. It targets enterprise application development and management, in all its aspects.



» **Enterprise Open Source Magazine**
EOS is the world's leading publication showcasing every aspect of profitable Open Source solutions in business and consumer contexts.

SYS-CON EVENTS For more great events visit www.EVENTS.SYS-CON.COM

COPYRIGHT ©2007 SYS-CON MEDIA ALL RIGHTS RESERVED

Exhibit and Sponsorship Info:

Call 201-802-3020 or email events@sys-con.com

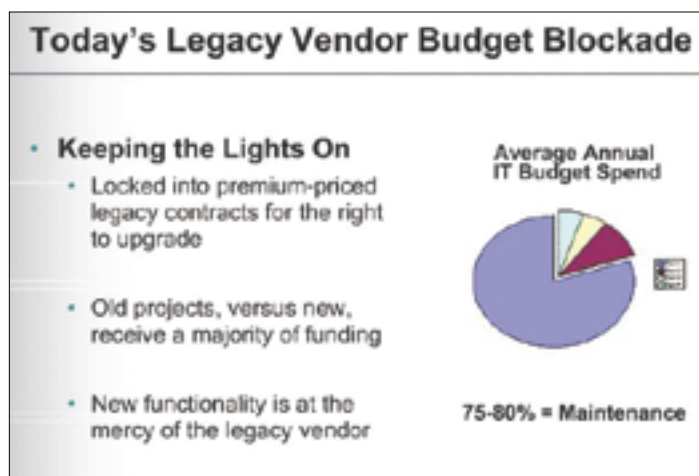


Figure 1

databases or built data integration technologies that allowed them to serve up the data they needed in the enterprise.

It became very expensive to do this and a lot of companies felt they were undergoing projects that were identical to projects already underway and they were competitors to other companies in their market. A number of database companies sprang up in the '70s and early '80s: Relational Technology, Oracle, Symation, and so on, that focused on delivering relational database solutions. If the customers could free up those resources, which had previously been focused on delivering relational database solutions, to focus on the business, then they were quite willing to pay large sums of money for the technology they were getting back as there was a lot of value in it.

What happened in the mid-'80s through to fairly recent times is that people have felt they were paying ever-increasing prices for database technologies and are not getting the value they perhaps received in the earlier days. They're getting new features that are somewhat niche and perhaps not needed in their business. They're being asked to fund these new developments and perhaps aren't making use of them and certainly don't perceive it as being a good value. This is what's driving the open source movement: people feel they're not getting value for the money they're spending today, which has led them to look at open source solutions where the perception is that what you pay for is what you're using. In the case of Ingres we have no upfront licensing fee for the product; what you pay for is the knowledge or the insurance policy that is essentially a support subscription.

The market will to be driven to equilibrium again and open source is going to drive that.

Figure 1 shows the breakdown of the annual IT budget. At the beginning of the fiscal year, 75–80% of the budget is already spoken for. Typically this is to continue paying license fees for co-source solutions, for support subscriptions,

maintenance agreements, and just keeping the currently deployed technology in place. As a result, there's very little money left at the beginning of the year to invest in new projects. This is where we believe we play an important part, because we want to allow you to derive maximum value from the money that you have to spend on innovation. It's a low-cost, high-value platform, and we want to support innovation.

Looking at the money you're spending today for a database, you could probably divide the database world in half. The first half is the database-centric environment. Here the database is the center of the universe but your applications are designed around the capabilities that the database affords. The database really drives the business logic that you can deliver to the business and oftentimes you're paying very high premiums for databases in this type of environment.

There seems to be a bit of resentment about that because the databases are perceived as delivering a lot of value. On the right-hand side of Figure 2 is this second camp, the application-centric world. In this type of database deployment, the application is delivering the value to the business. The database is there to support the application, but it's the application functionality and the business logic it delivers that's important and is at the center of the operations here. Look at your environment; do you fall into a database-centric camp or an application-centric camp? If you're really all about the application, why are you paying such high premiums for the databases that support those applications? How much value are you getting from this and how much innovation is afforded to you in your environment through the use of your database technology?

For Ingres Icebreaker, we've taken a Linux distribution that's provided by a company called R Patch; R Patch Linux is a componentized version of Linux in which they break out the operating

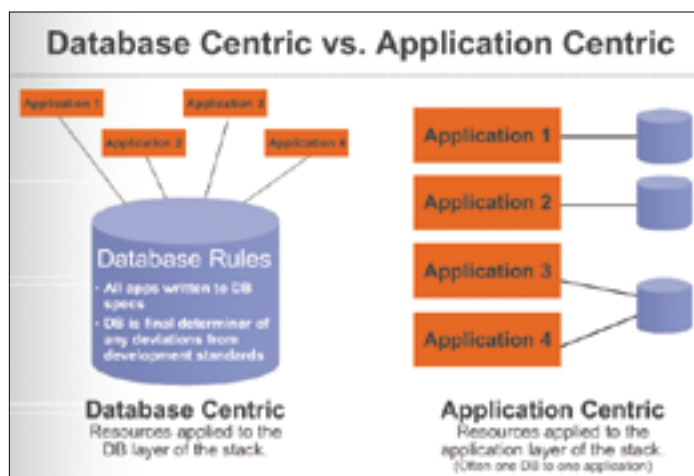


Figure 2

system into as many components as they can afford. We've looked through these sets of components and identified the ones that are required to support the database. What you'll find is that you'll have general purpose operating systems. If you look at something such as Windows, even in your desktop environment there are a host of services, user accounts, and technologies that are part of the Windows operating system and you probably don't know what they're for. You probably know how to secure them, and you may have the inconvenience of having to patch them without ever using them.

What the Icebreaker solution does is combine only the operating system components that are required to support Ingres, greatly simplifying deployment. We only have the pieces required by Ingres. We provide a single installed image so what you get is the database and the operating system. There aren't any extraneous user accounts that often compromise the security of the system, no extraneous daemons and services running in the background. There is an integrated maintenance stream, so as you get patches for the environment, the patch will address issues with both the operating system and the database. There aren't separate patch lines for this; it's a single integrated environment.

Icebreaker is a tool for Eclipse users to utilize as the database platform for the Eclipse IDE so as they develop applications, they can focus on what they do best, which is application development, and not have to worry about the installation configuration and maintenance of the databases that might support those applications.

Icebreaker allows you greater freedom to innovate. It's an open source solution that's delivered under the GPL and is perfect for virtualized development environments.

To view the complete Webcast, please go to <http://www2.sys-con.com/webinararchive.cfm?pid=ingresmarch07>.

2007 VIRTUALIZATION CONFERENCE + EXPO

www.virtualizationconference.com



Register Now!
**EARLY-BIRD
SAVINGS!**
\$200

ROOSEVELT HOTEL
NEWYORK CITY

June 25-27, 2007

Virtualization: Solutions for the Enterprise.

Delivering The #1 i-Technology Educational and Networking Event of the Year!

As today's CIOs and IT managers rise to the challenge of using their enterprise computing infrastructure as cost-effectively as possible while remaining responsive in supporting new business initiatives and flexible in adapting to organizational changes, Virtualization has become more and more important.

A fundamental technological innovation that allows skilled IT managers to deploy creative solutions to such business challenges, Virtualization is a methodology whose time has come. The fast-emerging age of Grid Computing is enabling the virtualization of distributed computing, of IT resources such as storage, bandwidth, and CPU cycles.

But Virtualization can also apply to a range of system layers, including hardware-level virtualization, operating system level virtualization, and high-level language virtual machines.

Register Online!

www.VirtualizationConference.com



THE FIRST MAJOR VIRTUALIZATION EVENT IN THE WORLD!

**HURRY, FOR EXHIBITOR AND
SPONSORSHIP OPPORTUNITIES
CALL 201-802-3020**

*Learn from
the Experts...*

— BROUGHT TO YOU BY —



For more great events visit www.EVENTS.SYS-CON.com

COPYRIGHT ©2007 SYS-CON MEDIA. ALL RIGHTS RESERVED

ENTERPRISE
OpenSource
MAGAZINE

SOAWORLD
MAGAZINE

How to Acquire an Open Source Software Company 2.0

by Douglas Levin

These days, executives realize that there are “new school” ways of acquiring a company with a software asset. For all of the immeasurable benefits it has brought to the development community, open source technology has added a complex variable to relevant parties calculating the M&A equation. Open source code and the general reuse of open source and proprietary software components in software development have further complicated the process of acquiring a software asset.



Closing a deal that includes a software asset these days is not much different than shopping for a car. On the outside, the streamlined exterior and highly finished shell might look very glamorous. But the question needs to be asked, “Would you buy the car without first looking under its hood?” Purchasing any software asset demands that due diligence be paid to evaluating the integrity of that software’s code. It’s in the best interest of the acquirer to make sure that the asset they are acquiring complies with all intellectual property and licensing requirements. In other words, you want to be sure that the amalgam of parts used to build the engine, assembled in piecemeal fashion, meet the criterion set for passing vehicular inspection.

Virtually all companies that develop software are now working in a “mixed-IP environment.” The reuse of internal and externally available source code by developers is inevitable. Most software architects use a “component assembly” approach to creating this hybrid type of software. It usually involves using a combination of third-party code, open source code, and proprietary code. However, such uncontrolled re-use of available source code introduces the need to assure the code is used in accordance with applicable licenses.

Manually evaluating software developed through a “component assembly” approach is the “old school” way of performing due diligence. This approach can prove to be an

exhausting and time-consuming exercise. The typical parties normally involved in the M&A of a software asset can potentially spend many hours trying to decipher what is in the code and what are the relevant licenses that apply to the code. For legal teams, auditors, and executives involved in such an M&A deal, time is of the essence.

All software carries licensing obligations, which may not be onerous, but companies are required to abide by them. In a mixed-IP environment, the volume of licenses to be understood and tracked can quickly become a challenge, especially if those licenses and their many different obligations conflict with one another.

Without insight into the composition of software and licensing restrictions, software assets can be laden with serious intellectual property problems. If license conflicts arise too late in the M&A transaction, it can lead to costly code reviews or redesigns for internal development teams. License violations can also cause embarrassing public and investor relations nightmares for companies.

With the onset of the Free Software Foundation’s release of General Public License (GPLv3.0) on the near horizon, now more than ever companies need to be able to quickly evaluate the code they are acquiring, and be able to distinguish usage rules from GPLv2.0 and v3.0.

Given the time-sensitive nature of M&A transactions, acquirers need a software compliance management solution that stream-

About the Author

Douglas Levin founded Black Duck Software (www.blackduck-software.com) in 2002 and has been its chief executive officer and president since its inception. He also holds a certificate in international economics from the College d’Europe in Bruges, Belgium.


lines the due diligence process. Furthermore, such a solution needs to provide the acquirer a snapshot analysis of the software asset that they are acquiring. But at the same time, the system also needs to be able to protect the source code and other intellectual property of the acquired party's asset. The solution needs to be able to keep the two parties involved in the transaction separated, while providing each the information it needs to continue the process.

Essentially, such a solution needs to be able to move quickly, maintain confidentiality, and match the pace of the due diligence process. Also, all relevant participants in these types of transactions must be able to easily interpret the information produced by such a solution in a user-friendly format.

Both large and small companies have a need for this type of compliance management solution. Many large companies are typically involved in multiple transactions per year that could potentially involve licensing issues in the software they are looking to acquire. If detected too late in the transaction process, these types of licensing issues could force the acquirer to renegotiate the value of its acquisition. Consequently, a company

could save itself millions of dollars by investing in a software compliance management solution that would mitigate the risk.

Likewise, a small software company wishing to be acquired could certainly benefit from deploying such a solution. The company would need to be able to rely on being able to utilize the solution to quickly review their code and remediate all issues prior to any potential transaction. In an era when many enterprise open source assets are being rapidly acquired, small software companies have a strong need for a solution that offers the ability to employ this functionality in record time.

The reality facing the industry is that software reuse is better, faster, and cheaper. By the same token, it is now harder to track and manage embedded code used in the development of a software asset. The risk of being in violation of certain license restrictions is heightened for a purchaser acquiring a software asset, when that asset originates out of a "component assembly" approach. A software compliance management solution that allows for both parties in an M&A transaction to mitigate the risks and reap the benefits is sorely needed in this age of mixed-IP environments. 


from the editor —continued from page 3

the course of the upcoming years, the goal is to put Linux laptops into the hands of children throughout the world. Should OLPC be successful, they will put millions of Linux computers into hands, increasing the ranks of open source users substantially.

In addition, based on lessons that they learned while partnering with the OLPC project, Red Hat launched a new desktop product on May 9, 2007: the Red Hat Global Desktop, a new commercial client operating system. This was a divergence from former Red Hat Linux desktops that consistently mirrored the Red Hat server products. The Red Hat CTO ushered in the new desktop product saying, "Users, requirements, and technologies have changed so dramatically over the past few years that the traditional one-size-fits-all desktop paradigm is simply exhausted. Our strategy is to deliver technologies that are specifically appropriate to these varied constituents, all based on open standards." In addition to this new product, Intel announced that they would, in partnership with Red Hat, offer pre-certified, cost-effective PCs in Intel's reseller channel running the new desktop Linux product.

A final encouraging note is the emergence of cross-distribution application vendors. Specifically an offering announced by Linspire in January to bring one-click deployment of Linux applications to not only their own distributions, Linspire and Freespire, but also Debian, Fedora, OpenSuSE, and Ubuntu. This is an interesting step as it eases adoption of open source and commercial applications uniformly across Linux distributions. It also provides a legal way for desktop users to download commercial codecs for playing Windows Multimedia files and solves other problems associated with delivering commercial or proprietary applications to open source platforms. While I am a fan of Linspire, the

truly interesting point is the emergence of cross-distribution desktop vendors. This would almost be analogous to having Windows Update work for Mac OS and Solaris as well as Windows.

My prediction is not that there is an upcoming flood of desktop Linux users. Nor am I trying to position Linux as a David to the Goliath we know as Microsoft. What I do know is that Linux will continue to improve and become a viable alternative to today's mainstream desktops; that support from companies like Red Hat, Dell, Intel, and Canonical will cause slow and steady expansion into the developed and the developing world. 

Other Links

- *Ubuntu*: www.ubuntu.com
- *Parallels*: www.parallels.com
- *Linspire*: www.linspire.com
- *One Laptop Per Child*: www.laptop.org

Endnotes

1. *At this point I was using a Dell Inspiron 9100 running SUSE Linux.*
2. *I would typically run Windows 98, with Microsoft Office from within Win4Lin. I was COO of Win4Lin Inc. at that time.*
3. *Mac OS X release 10.0 Cheetah was released in March 2001; it was based substantially on the BSD operating system.*
4. *Windows to Linux Business Migration* by Mark R. Hinkle. Charles River Media, January 2006.
5. *Announcement was made via Direct2Dell Blog - <http://direct2dell.com/one2one/archive/2007/05/01/13147.aspx>*

Enable a Module-Based Approach to Constructing Services

Using the force (and open source)

by Franz Garsombke

The Jedi mind trick is a Force power that can influence the actions of weak-minded sentient beings. Vendors will often try to apply the Jedi mind trick in selling silver-bullet software solutions that solve global warming and stop celebrity feuding while enabling service-based architecture development. Let's quickly put on our aluminum foil caps and repel the Jedi mind trick by turning to open source solutions. Service-based architectures are being touted as the next step in reaching programming nirvana. With these marching orders it's often difficult to build a framework that allows for simple service creation.



This framework should also be flexible, scalable, and lightweight as well as easy in exposing services externally. Without the correct framework(s) and guardrails in place your application services can quickly become a jumbled mess. Circular dependencies, massive Ant scripts, zero component reuse, and painful Web Service creation are just some of the problems faced by developers today. This article will demonstrate concepts on software lifecycle automation as well as a lightweight approach to creating a service-based architecture.

Building a Better Death Star

Everyone has worked on a software project delivered late (if at all) and over budget. I'm sure building the Death Star was no different. This article will demonstrate how to alleviate some of that pain by building a modular, reusable, and service-based architecture using a combination of open source projects. Maven, XFire, Spring, and Hibernate are all powerful frameworks when used alone. The real power is realized when they are used in conjunction. Maven is a software management tool that allows for module inheritance, dependency management, and overall project comprehension. XFire is a simple yet feature-rich framework that can expose any plain old Java object (POJO) as a Web Service. Spring is a layered Java application framework used to wire services together. Hibernate is an object-relational-mapping persistence and query service for Java.

You Say Jabba Desilijic Tiure, I Say Jabba the Hutt

Jabba Desilijic Tiure was one of the most notorious crime lords in the Star Wars galaxy. You probably didn't know who the term *Jabba Desilijic Tiure* referred to but if *Jabba the Hutt* was mentioned you'd know right away the creature in question. A common data dictionary is beneficial on any project or document. Before we continue a few key terms should be defined so it doesn't sound like this article was written in *Shyriiwook* (Wookiespeak).

- **Service:** The term service refers to a "discretely defined set of contiguous and autonomous business or technical functionality." One of the goals in creating a robust application framework is the ability to expose services through a number of different protocols. Creating services as POJOs will let us use other frameworks either to expose these externally or consume them internally.
- **POM (Project Object Model):** The POM is an XML representation of a Maven project. A POM consists of information regarding dependencies, defect tracking, repository metadata, mailing lists, plug-in goals, and any other configuration details used to encompass fully the complete lifecycle of a project.
- **Module:** A module is a logical as well as physical part of your application. Layers and services in your application can be represented by physical Maven modules that logically make up the different components

About the Author

Franz Garsombke has been developing and architecting enterprise software solutions in Colorado for the last 11 years and is currently employed at Rally Software. He is a huge proponent of open source frameworks and passionate about developing and delivering simple quality pragmatic applications. He is proud to be the co-founder of a Java Bean mapping framework (<http://dozer.sourceforge.net>).
fgarsombke@yahoo.com

- of your architecture.
- **Build Dependencies:** Maven manages different types of build dependencies for projects and cross-projects. A dependency is typically a jar file that a project needs for compilation, testing, or runtime behavior. Dependencies can be managed declaratively in your Maven project file. Maven also seamlessly manages transitive dependencies (dependencies of dependencies) and circular dependencies. Transitive dependencies are impossible to manage in Ant. Circular dependencies are particularly hard to manage in Ant since most of the time the source code for an application is compiled in one big mess. No feedback is given if, for example, the core layer of your architecture has a dependency on your service layer.
 - **Build Lifecycle:** Software projects typically need the same tasks done to build and deploy code. Maven has a predefined set of phases each performing a series of actions to encompass a build lifecycle. Some common phases are validate, generate resources, compile, test, package, integration test, and deploy. Maven plug-ins can be used to add functionality to a standard build lifecycle. Lifecycle phases should be managed by your software project management tool and not bolted onto a tool like Ant. Figure 1 shows some common lifecycle phases and the tasks they perform.

Just Say No to a 'Jabba the Hut' Architecture

It's more than likely that everyone has worked on an application that suffered from this syndrome. All it took was consuming one or two salacious crumbs a day without giving the application the care and feeding it needed. Bloating, circular dependencies, and the absence of separation of concerns are telltale signs that your architecture is turning into Jabba the Hut. Anyone using Java has surely written a few Ant files in the course of their career each consisting of thousands of lines of XML. Everything seems fine until your code base grows or more developers are added to the project. It's not simple to manage modules (or sub-projects) and dependencies using Ant as your build system. It's a great tool but it's outdated and too many people have tried to turn it into something it shouldn't be. A good analogy of Maven compared to Ant is the symbiotic relationship between Hibernate and JDBC. Hibernate provides an abstraction layer on top of JDBC and can provide a broader scope and range of features than the JDBC framework. The Maven plug-ins provide all

the features of Ant as well as a standard way of interfacing with external tools and frameworks like Cargo (application container management), configuration management repositories, Hibernate, XFire, etc. Maven also brings a much broader set of features and concepts to the table than Ant.

Typical Project Object Model (POM) Build Life Cycle							
validate	compile	test	package	integration test	verify	install	deploy
validate all necessary information for the project	compile the source code	test the compiled source code	package up compiled source and other resources	deploy package for integration testing	verify package against criteria	install the package into the local repository	deploy the package to a remote repository

Figure 1

Star Wars Convention? No, Not That Kind of Convention...

Maven's magic in building software is similar to the Ruby on Rails approach to programming in that most common tasks have default sensible strategies in place. This philosophy is called "Convention over Configuration." Trading flexibility at the infrastructure level by having standard directory layouts, standard naming conventions, and standard lifecycle phases promotes less emphasis on mundane details like building and deploying and allows for more time spent on the application's design and implementation. The guardrails Maven puts in place are necessary for a tight application but often don't get installed and are typically not given the care and feeding they need.

Software build structure layering should be the foundation of your application. This layering is prevalent at many levels in your ecosystem. It starts at your build system and feeds into your application. At an even broader scale this can be seen from your application into the enterprise. This was all made possible by a well structured build system.

I Am Your Father Luke

Project inheritance? Are you crazy? Wouldn't it be nice if there was a build system that handled software management lifecycle tasks and had some familiar and powerful object-oriented constructs like inheritance, encapsulation, cohesion, composition, and aggregation? I first used Maven 1 on an open source project solely for dependency management and site documentation. Frankly, I thought everything else it did was pretty miserable. I found myself always needing to drop into Ant

to do trivial things. With the release of Maven 2 these highly desired features are present along with a lot more. This article is by no means a Maven tutorial but will show some key concepts as well as how to integrate it with some other important open source frameworks.

Maven comes with a feature rich set of plug-ins that provide core as well as enhanced functionality. Plug-ins are executed as part of the build lifecycle to perform tasks necessary to build a project. These plug-ins run unit tests, compile code, package software, and generate project reports among other things. Maven also has an extensible architecture that allows one to write any number of custom plug-ins to integrate with third-party tools or fulfill special tasks. One of the main benefits of using Maven is the ability to define a large amount of your build infrastructure in a parent project and inherit these dependencies and preconfigured plug-ins as well as any other lifecycle logic in your sub-modules. Here's a quote from *Better Builds with Maven* that sums up the tool, "Maven is a way of approaching a set of software as a collection of highly interdependent components that can be described in a common format. It's the next step in the evolution of how individuals and organizations collaborate to create software systems."

Now, Onto Building a Better Death Star

Everybody knows that the Death Star was destroyed by Luke Skywalker in Star Wars Episode IV: A New Hope. What they don't know is that another Death Star is being built using the latest and greatest Java frameworks. I guess you could say it's our lucky day. It's not easy constructing a monumental bringer of death out in the middle of nowhere. Droids, and many of them, must be gathered from across the galaxy to help in its construction. The system we'll be building for the Galactic Empire is an android (droid for short) provisioning system. I know that the Federation is in essence the Dark Side but the rates and benefits are outstanding. It doesn't hurt that there's free light saber training and laser target practice on Wednesday nights.

The key concepts that should be remembered throughout this exercise are:

- All of the services (modules) are POJOs that can be injected into any other module. These services could be a Web Service, data access service, common framework, business logic, or any number of other things required to make your application function.
- A service is a service is a service. Since our services are POJOs as well as Spring wired

beans they can be exposed externally through any number of transport protocols. There are many frameworks that are tightly integrated with Spring that let you expose a Spring bean over HTTP, JMS, RMI, etc.

Let's look at the services that comprise our droid provisioning system as well as the key technologies that are used to bring it to life. Figure 2 shows the Maven modules of the droid provisioning application and their mapping to logical services with technology implementations.

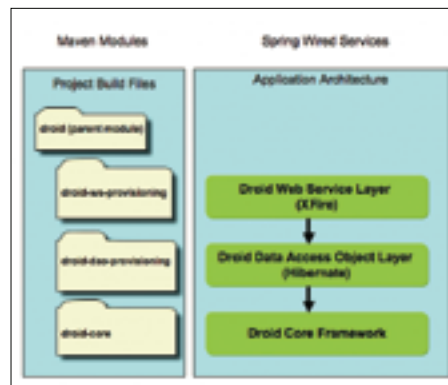


Figure 2

This sample application will demonstrate injecting services (both through Spring and Maven), WSDL-first development using XFire, and the creation of a data access service layer using Hibernate. All of the corresponding code for this example can be found at <http://dozer.sourceforge.net/example.zip>.

Anatomy of the Parent POM

Maven provides the concept of a parent module that is key to project inheritance. This module lets you define common dependencies, organizational information, deployment information, and common plug-ins. A skeleton of the provisioning application's POM file is shown in Listing 1.

This listing shows a typical POM layout structure. This file defines project-wide dependencies, project metadata, and common plug-in configurations.

In a Galaxy Far, Far Away

Let's code. The first thing we need is a common module that can be used by all upstream modules. This core module is comprised of one service but could potentially hold many others. Our core service really doesn't do much except elaborate on the concept of injecting and reusing services. Here's our service de-

fined as a Spring bean in our module's Spring context file:

```
<bean id="coreService" class="com.examples.droid.core.service.CoreService"/>
```

This bean references a Java class called *CoreService* that executes some core functionality.

The Maven configuration for this module is in Listing 2.

By defining our core module as a packaging type of *jar* Maven will compile our code, run unit tests, and package our classes into a jar file. The parent module inheritance is defined at the top of the configuration. No dependency version numbers are needed since they are defined in the parent. Let's see how this service can be used in sibling modules.

Springtime for Storm Troopers

Maven has the concept of dependency *management* to Spring's concept of dependency *injection*. At an abstract level the two are very similar and work hand-in-hand allowing for a reusable pluggable architecture. To add the core module as a dependency of another module we simply have to define it in our sibling module's POM file.

```
<dependencies>
  <dependency>
    <groupId>com.examples.droid</groupId>
    <artifactId>droid-core</artifactId>
  </dependency>
  ...
</dependencies>
```

The version for the dependency has been defined in the parent module. To use the service defined in the core module we'd simply inject it into the service defined in the sibling module's Spring context file.

```
<bean id="provisioningServiceImpl" class="com.examples.droid.ws.provisioning.ProvisioningServiceImpl">
  <property name="coreService" ref="coreService"/>
  ...
</bean>
```

To reference the Spring context defined in the core module it needs to be included in the list of context files that make up the system's Spring *ApplicationContext*:

```
String[] configFiles = new String[] { "spring/droid-core-beans.xml",
                                     "spring/droid-dao-provisioning-beans.xml" };
BeanFactory factory = new ClassPathXmlApplicationContext(configFiles);
```

Web 2.0 and Rich Internet Apps

CALL FOR PAPERS NOW OPEN!

AJAXWORLDTM CONFERENCE & EXPO

Providing developers and IT managers alike with comprehensive information and insight into the biggest paradigm shift in website design, development, and deployment since the invention of the World Wide Web itself a decade ago.

On September 24-26, 2007 over 1,000 developers, architects, IT managers, and software professionals of every stripe will be converging in Santa Clara, CA to attend the West coast AJAXWorld Conference & Expo -- the most comprehensive meeting on the most significant technology subjects of recent times: AJAX, Rich Internet Apps & Web 2.0.

Experience AJAX, RIA, and Web 2.0 Knowledge and Best Practices at AJAXWorld Conference and Expo 2007

Delegates will hear first-hand from the creators, innovators, leaders and early adopters of AJAX, and a slew of leading vendors will provide sneak-peeks of the very latest frameworks, tools, products and applications during the conference, which will have over 100 sessions and presentations given by 125 different speakers - the largest AJAX-focused speaker faculty ever assembled in one place at one time.

AJAXWorld Conference & Expo 2007 West will be jump-started with a full one-day "AJAX University Boot Camp," followed by the Main Conference and Expo over the following 2 days.



Highlights Include:

- **100+ conference speakers** selected from among the world's leading AJAX technologists and practitioners, including high-level IT executives, industry thought leaders, VCs and analysts
- **Topical, Interactive "Power Panels"** simulcast on SYS-CON.TV
- **Demos from 30+ vendors** both in General Session and in the Exhibit Hall/AJAX Showcase
- **Pre-Conference "AJAXWorld University Bootcamp"** for those wanting an all-day, hands-on encounter with AJAX
- **Leading-edge sessions** on issues like Offline AJAX, Mobile AJAX, Enterprise AJAX, AJAX Security, Desktop AJAX, Semantic Mash-Ups, Next-Generation SaaS, JavaScript & XML, Google Web Toolkit, Adobe Apollo, Adobe Flex, AJAX RIA GUIs, and Appropriate vs Inappropriate Use of AJAX.

.....> **September 24-26, 2007**

.....> **Santa Clara Convention Center**
Hyatt Regency Silicon Valley
Santa Clara, CA

.....> **To Register**
Call 201-802-3020 or
Visit www.AJAXWorld.com

.....> **Hurry! Limited Seating**
This Conference Will Sell-Out!



Hyatt Regency Silicon Valley
Santa Clara, CA

.....> **Join Over 3,600 Early AJAX Adopters Who Attended the #1 Web 2.0 Event in the World!**



.....> **Early Bird Price: \$1,495**
(Register Before May 25, 2007)



For more great events visit www.EVENTS.SYS-CON.com

VISIT WWW.AJAXWORLD.COM FOR THE MOST COMPLETE UP-TO-DATE INFORMATION

We now know how to create a service module and inject it through Maven dependency *management* as well as through Spring's dependency *injection*. Let's create some more services while working up our layers of the architecture.

Darth Vader Wants His TPS Reports!

Darth Vader has seen the initial budget for the new Death Star and is having heartburn. He needs a way to run some queries on our android provisioning database. Darth is pretty savvy so he chose Hibernate for the application's object relational mapping (ORM) persistence framework. Using Hibernate tools to reverse-engineer the database lets us dynamically generate our ORM files and data access objects. Since we rely on this generated code before we compile our persistence module we can simply bind the Hibernate tool's source generation task to a particular Maven lifecycle phase. The Hibernate source generation is done at the *generate-sources* lifecycle phase in Maven by defining our plug-in's phase element as shown in Listing 3.

This task was done in a *maven-antrun-plugin* as an Ant task since when this article was written the Hibernate tool's Maven plug-in wasn't as robust as the Ant task. The Maven variable *maven.compile.classpath* lets you have all of the module's dependencies and transitive dependencies at your fingertips.

Typically, a real-world application will have a database running for these Hibernate tools tasks to reverse-engineer against. The example code uses a HSQL database engine to simulate this behavior. The source code to start-up, create, and populate the database can be found in the example application.

The data access service is defined in our Spring context file in Listing 4.

The database consists of one table called *ANDROID* that's mapped to a Java class called *Android*. The generated data access object can then be used in our data access provisioning service:

```
public class ProvisioningDaoImpl extends AbstractDao implements ProvisioningDao {

    public Android findDroid(String id) {
        return (Android) load(Android.class, id);
    }
}

public abstract class AbstractDao {

    public Object load(Class clazz, Serializable id) {
```

```
        return this.sessionFactory.getCurrentSession().
load(clazz, id);
    }
}
```

The configuration needed for Hibernate is performed behind the scenes using Spring.

WSDL-First, You Must Do

Vader is getting impatient. He still can't access his TPS reports since there's no external API. We can expose our data access service with a Web Service using a simple framework called XFire. Let's examine the steps to enable simple top-down Web Service creation.

WSDL-first development is the concept of writing the interface for your Web Service before you write the code. WSDL-first development makes sense because it focuses less on a particular programming language and more on the messages between systems. Many people take the approach of generating the WSDL from existing code whether it's an EJB, POJO, or some other programming construct. WSDL-first takes a more top-down approach wherein the schema and WSDL are designed first and code generation happens from there. XML Schema is language-independent and provides more flexibility than a lot of programming languages. Client code, interfaces, implementation code, and schema types can all be generated from a WSDL and schema by a number of SOAP stack frameworks. One of the main benefits of WSDL-first development is improving interoperability between disparate systems programmed in different languages.

Taking our top-down approach let's look at an operation defined on our WSDL:

```
<wsdl:operation name="findDroid">
  <soap:operation style="document" soapAction="findDroid"/>
  <wsdl:input><soap:body use="literal"/></wsdl:input>
  <wsdl:output><soap:body use="literal"/></wsdl:output>
</wsdl:operation>
```

The *document/literal* approach to constructing a Web Service is more straightforward and solves many interoperability issues since it simply relies on XML Schema to describe exactly what the message looks like when delivered. Also, *document/literal* is WS-I (Web Services Interoperability) compliant. The request and response messages are defined below:

```
<xs:element name="findDroid" type="provisioning:findDroid"/>
<xs:complexType name="findDroid">
  ...
```



```

</xs:complexType>
<xs:element name="findDroidResponse" type="provisioning:
findDroid-
Response"/>
<xs:complexType name="findDroidResponse">
...
</xs:complexType>

```

By using a combination of XMLBeans (one of the many XML binding mechanisms supported by XFire), XFire, and our WSDL we can use separate tasks in Maven to generate everything needed to build a Web Service. There's an execution step defined in our *maven-antrun-plugin* in Listing 5.

All of the Web Service code generation is done in the *generate-sources* phase of the Maven build lifecycle. This lets us generate all of the messaging objects we rely on before our implementation code needs to compile against them. Since all of our messaging objects have been generated we can now implement our provisioning Web Service. XFire creates an implementation class based on the WSDL provided. That same class will be used to implement all of the business logic needed by our system.

The provisioning service is injected with the core service and data access service defined and implemented in earlier modules. The code in Listing 6 shows the provisioning service using the two services it's dependent on to look up android information.

The only thing left to do is wire up our provisioning Web Service. An important thing to note is that the provisioning service is a POJO. We could easily inject it into a different transport protocol implementation. XFire is

used in this instance. We define our provisioning service as a service bean in XFire's service factory and are off and running.

```

<serviceFactory>org.codehaus.xfire.xmlbeans.XmlBeansServiceFactory</serviceFactory>
<serviceBean>#provisioningServiceImpl</serviceBean>
</service>

<bean id="provisioningServiceImpl" class="com.examples.droid.ws.provisioning.ProvisioningServiceImpl">
<property name="coreService" ref="coreService"/>
<property name="provisioningDao" ref="provisioningDaoImpl"/>
</bean>

```


XFire takes care of the Web servlet, marshaling/unmarshaling of the SOAP messages, and determining which service and which operation needs to be executed. If our requirements dictate that we need a composite service comprised of the provisioning service and some other service we could just create a higher-level module and inject the services that have already been built. The provisioning service would then be just a POJO as opposed to a formal Web Service.

Empire Building

The example demonstrates how Maven can be leveraged to create an entire framework that developers can work in. If the infrastructure is set up properly developers can leverage and reuse many tasks through project inheritance and common plug-ins. Consistency goes a long way in creating large enterprise applications. Adopting a "convention over configuration" approach will let developers

build a general framework that will scale to large projects.

The Saga Continues

I encourage you to dive deeper into the Maven application management tool and think about how it can really drive and enable a more modular service-based vision for your architecture. I also encourage you to look into using Maven with the other open source frameworks listed in this article. The build system is truly a reflection of your application's health in terms of maintainability and the ability to bolt on new modules in a timely manner. This article demonstrated at a conceptual level how many different open source frameworks could be glued together to create an extremely flexible layered architecture able to expose modules as services. These services can be reused in other parts of an application by merely injecting them into horizontal or upstream modules. 

Acknowledgements

I would like to thank Steve Meyer and Chris Swift for invaluable editing comments and ideas.

References

- Cargo: <http://cargo.codehaus.org/>
- Dependency Injection: <http://www.martinfowler.com/articles/injection.html>
- Hibernate: <http://www.hibernate.org/>
- Maven: <http://maven.apache.org/>
- Spring: <http://www.springframework.org/>
- Star Wars: http://en.wikipedia.org/wiki/Star_Wars
- XMLBeans: <http://xmlbeans.apache.org/>
- XFire: <http://xfire.codehaus.org/>

Listing 1

```

<?xml version="1.0"?>
<project>
<modelVersion>4.0.0</modelVersion>
<groupId>com.examples.droid</groupId>
<artifactId>droid</artifactId>
<packaging>pom</packaging>
<version>1.0</version>
<name>droid</name>
<dependencyManagement>
<dependencies>
<dependency>
<groupId>org.codehaus.xfire</groupId>
<artifactId>xfire-spring</artifactId>
<version>1.2.4</version>
</dependency>
</dependencies>
<dependency>
<groupId>com.examples.droid</groupId>
<artifactId>droid-core</artifactId>

```

```

<version>${project.version}</version>
</dependency>
...
</dependencies>
</dependencyManagement>
<modules>
<module>droid-core</module>
<module>droid-dao-provisioning</module>
<module>droid-ws-provisioning</module>
</modules>
<build>
<plugins>
...
</plugins>
</build>
</project>

```

Listing 2

```

<?xml version="1.0"?>

```

```

<project>
<parent>
  <groupId>com.examples.droid</groupId>
  <artifactId>droid</artifactId>
  <version>1.0</version>
</parent>
<modelVersion>4.0.0</modelVersion>
<groupId>com.examples.droid</groupId>
<artifactId>droid-core</artifactId>
<packaging>jar</packaging>
<name>droid core</name>
<dependencies>
  <dependency>
    <groupId>log4j</groupId>
    <artifactId>log4j</artifactId>
  </dependency>
  ... ..
</dependencies>
</project>

```

Listing 3

```

<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-antrun-plugin</artifactId>
  <executions>
    <execution>
      <id>generate-hibernate-source</id>
      <phase>generate-sources</phase>
      <configuration>
        <tasks>

          <!-- Enable the HibernateToolTask -->
          <taskdef name="hibernatetool"
                    tool.ant.HibernateToolTask" classpathref="maven.compile.classpath"/>
          <!-- Generate XML metadata mapping files from database schema -->
          <hibernatetool>
            ...
          </hibernatetool>

        </tasks>
      </configuration>
    </execution>
  </executions>
</plugin>

```

Listing 4

```

<bean id="provisioningDaoImpl" class="com.examples.droid.dao.provisioning.ProvisioningDaoImpl">
  <property name="sessionFactory" ref="provisioningSessionFactory"/>
</bean>

<bean id="provisioningSessionFactory" class="org.springframework.orm.hibernate3.LocalSessionFactoryBean">
  <property name="dataSource" ref="provisioningDataSource"/>
  <property name="mappingResources">
    <list>
      <value>com/examples/droid/dao/provisioning/Android.hbm.xml</value>
    </list>
  </property>
  <property name="hibernateProperties">
    <props>
      <prop key="hibernate.dialect">org.hibernate.dialect.HSQLDialect</prop>

```

```

      <prop key="hibernate.hbm2ddl.auto">create</prop>
    </props>
  </property>
</bean>

```

Listing 5

```

<execution>
  <id>generate-wsdl-types</id>
  <phase>generate-sources</phase>
  <configuration>
    <tasks>
      <!-- generate and compile types from our schema -->
      <taskdef name="xmlbean" classname="org.apache.xmlbeans.impl.tool.XMLBean" classpathref="maven.compile.classpath"/>
      <xmlbean
        classgendir="${project.build.directory}/classes"
        srcgendir="${project.build.directory}/generated-source"
        classpathref="maven.compile.classpath"
        failonerror="true"
        javasource="1.5">
        <files dir="src/wsdl" includes="*.wsdl"/>
      </xmlbean>
      <!-- generate our service client, interface, and impl class -->
      <!-- using XFire's web service generation task -->
      <wsngen outputDirectory="${project.build.directory}/generated-source"
        wsdl="${basedir}/src/wsdl/${project.artifactId}.wsdl"
        package="${project.groupId}"
        generateServerStubs="true"
        binding="xmlbeans"/>
    </tasks>
    <sourceRoot>${project.build.directory}/generated-source</sourceRoot>
  </configuration>
</execution>

```

Listing 6

```

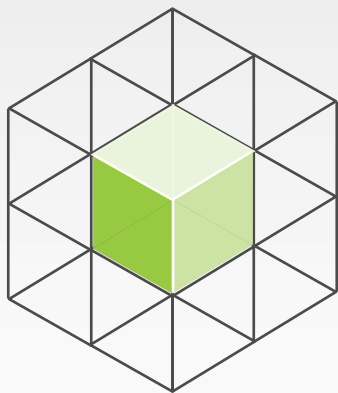
Public class ProvisioningServiceImpl implements ProvisioningServicePort {

  private CoreService coreService;
  private ProvisioningDao provisioningDao;

  public FindDroidResponseDocument findDroid(FindDroidDocument findDroidRequest) {
    FindDroidResponseDocument document = FindDroidResponseDocument.Factory.newInstance();
    // Calling the core service to execute some functionality
    getCoreService().doSomething("do something");
    // Use the data access service and call the method findDroid()
    // to find our droid
    Android android = getProvisioningDao().findDroid(findDroidRequest.getFindDroid().getDroidId());
    FindDroidResponse response = document.addNewFindDroidResponse();
    response.setIsError(false);
    DroidInfo droidInfo = response.addNewDroidInfo();
    // in essence this is our mapping of domain objects (Android) to
    // our message objects (DroidInfo). Never expose your domain
    // objects to the outside world.
    droidInfo.setName(android.getName());
    response.setDroidInfo(droidInfo);
    return document;
  }

  getters() and setters() for the two injected services...
}

```



Open Management Consortium

Systems Management is Now Open

In May 2006 the Open Management Consortium was announced to help advance the promotion, adoption, development and integration of open source systems /network management software. The founding members of the consortium are Ayamon, Emu Software, Qlusters, Symbiot, Webmin, and Zenoss.

Specific objectives of the Open Management Consortium include:

- Create awareness of open source management tools in the market
- Provide education and resources to help end users make informed decisions regarding open source
- Establish conventions and standards that enable integration and interoperability
- Enable collaboration and coordination on common development projects
- Promote collaborative open source systems management solutions

Open source systems management replaces monolithic vendor lock-in with a modular approach. Pick what you need, customize it to your exact specifications and add to it as your needs change. Because open source products welcome contributions by users, partners and other third parties, they must be standards-based. This standards-based approach facilitates interoperability between open source solutions across the systems management life cycle.



www.open-management.org



BREAK THE CYCLE. The HP BladeSystem c-Class, featuring efficient Dual-Core AMD Opteron™ processors, helps free I.T. from the cycle of server management. It's equipped with HP's exclusive Virtual Connect Architecture, which virtualizes your LAN/SAN settings, so you only wire once and can make network changes on the fly. This drastically improves efficiency and service levels and gives you the time to focus on the things that really drive your business.



Download the IDC White Paper on Blade I/O Virtualization
Call 1-888-225-7532
Visit www.hp.com/go/breakthecycle12

Set I.T. Free

