2010 Open Source Think Tank

The Future of Commercial Open Source

Executive Summary Report

2½ Days of Brainstorming Sessions and Panel Discussion with Leading Open Source CEOs, CIOs, VCs, Attorneys and Luminaries

Hosted by

olliance | group

DLA PIPER

Table of Contents

The Open Source Think Tank 2010

The 5th annual Spring Open Source Think Tank was held April 15-17, 2010, at the Meritage Resort in Napa, California. The event is structured as a "think tank"; different from a typical software industry conference, it includes group activities for networking and three different brainstorming sessions. The purpose of the Think Tank is for executives from all segments of the open source industry to openly share ideas on commercial opportunities and threats, and collaboratively develop strategies and possible potential paths forward.

The following is a synopsis of the event. It is not meant to represent the official company positions of any of the hosts, sponsors or attendees, but rather to provide an accurate as possible summary of the many substantive discussions held over the three days. The authors of this document have made every effort to keep personal opinions out, while faithfully recording the substance, facts and flavor of the event.

Hosts

The 2009 Open Source Think Tank was hosted by Olliance Group and DLA Piper.

olliance | group

Olliance Group is the leading global open source business and strategy consulting firm. Our mission is to help clients capitalize on the strategic, technological, and financial benefits of open source software. Olliance offers a set of strategic, business and technology planning, risk management, investment, and community development consulting services. For more information please visit the website at http://www.olliancegroup.com.

DLA PIPER

DLA Piper became one of the world's most prominent legal service providers in the world in 2005 through a transatlantic merger of unprecedented scope. Building strong and substantial client relationships was and remains the compass for DLA Piper's business strategy and future development. Today we have nearly 4,000 lawyers in more than 65 offices in Asia, Europe, the Middle East and the United States. For more information please visit the website at http://www.dlapiper.com.

Sponsors

Olliance Group and DLA Piper would like to thank our event sponsors for their generous support to make this event possible. This year's sponsors were:

Platinum Sponsors

Gold Sponsors

Silver Sponsors

Conference Attendance

118 individuals from 89 organizations attended the 2010 Open Source Think Tank. Attendees represented open source vendors, customers and end-user organizations; open source communities and foundations; investors and financial analysts. Although the majority of participants were from the US; European, Asian and African organizations also participated. The Think Tank is a "by invitation only" event, limited to senior-level open source executives and experts. Total attendance was capped to ensure the event would fulfill its purpose to enable interaction, discussion and relationship building opportunities for industry leaders in a small group format.

Event Format

The event was structured to provide attendees an opportunity to work together to share ideas and develop potential solutions to the business challenges facing commercial open source companies in the near-term future.

- Group sessions with business-oriented topics, developed by the attendees prior to the event
- Brainstorming meetings, where groups are structured to represent a cross-section of the industry, a particular topic from the general session is discussed and the groups presents their findings back to the full conference
- Interactive audience voting, throughout the event, where questions were posed to the audience for immediate response and tabulation
- Three panels representing a cross-section of commercial open source companies and customers discussing what is most important to them about open source software
- Organized networking and social activities provided ample opportunities for industry relationship building

Conference Agenda

The Think Tank consisted of two and a half days of activities and meetings, spread over Thursday, Friday and Saturday morning. The schedule included:

- Opening and closing remarks by the hosts
- Open Source Business and Legal Perspectives by Mark Radcliffe, Partner at DLA Piper
- Panel discussion on open source with CIOs from three private sector companies
- Panel discussion on open source with CIOs from several public sector CIOs
- Panel discussion on Open Source M&A with Mark Brewer of VMware/Springsource and Damien Eastwood of Sun Microsystems
- Panel discussion on Cloud and Open Source with Larry Augustin of SugarCRM and Tim Golden, SVP of IT, Bank of America
- Mobile and Open Source Business Case Workshop with introduction by Bill Weinberg of Olliance Group, with brainstorming and reports by eight working groups
- State of California Business Case Workshop – Sponsored by Teri Takai, State of CA, CIO and presented by P.K. Agarwal, Chief Technology Officer and Lee Mosbrucker, Director of Enterprise Architecture, State of California, with brainstorming and reports by eight working groups
- Open Source in Cloud Computing Business Case Workshop with introduction by Miriam Tuerk, Olliance Group, with brainstorming and reports by eight working groups
- Group networking events at the conclusion of each day

Business and Legal Update

Mark Radcliffe, Senior Partner at DLA Piper and Co-Counsel for the Open Source Initiative (OSI)

Mark provided an excellent overview of the last year's development in the business and legal dimensions of open source software. He described the overarching trend as open source software becoming ubiquitous, with much more awareness and some new problems.

Mark commented on a number of significant legal happenings in the world of open source software over the last year. These included:

The SCO case came back from the dead, but in the last two weeks a jury decided that SCO does not own the copyrights to Unix. A Think Tank participant asked if this means we should be concerned about Novell's ability to instigate infringement claims if it were sold into private equity. Mark indicated that there should be no real concern given Novell's current business, but possibly if it were sold and re-configured. However, IBM suit activity seems to indicate that there is no Unix code in Linux.

The Software Freedom Law Center maintained its high level of legal activity around ***busybox***. Largely as a result, several new efforts to establish standard reporting forms for supply chain software have been launched - Motorola is leading one of them.

The Jacobson case was decided, and the jury ruled for Jacobson. Injunctive relief for open source license violations still stands, but the standard has been set high. This means that you must select your OSS license carefully.

OSS litigation expanded beyond Germany, Isreal and the US with a new decision in France.

The Software Freedom Law Center published a new Exception for GCC, which allows it to be brought under the GPLv3.

The American Law Institute (ALI) published "Principles of the Law of Software Contracts - Official Text" which contains some provisions that could cause serious difficulties for software vendors. In particular, the idea that all software carries a non-disclaimable warranty of non-infringement and a non-disclaimable warranty of "no hidden material defects" present serious problems for most software companies. Efforts are underway to counter these recommendations from ALI.

Mark also commented on a number of new issues that have surfaced in 2010:

In the area of patents and open source the matter of TurboHercules and IBM has gotten much attention. Although there has been talk about IBM using its patents against open source, the facts here are simply that TurboHercules made a commercial proposal to IBM, and IBM responded "no" with a list of patents.

Mark and his colleagues are seeing that the rise of hybrid products is increasing the potential for conflicting license obligations. For instance, in a recent cloud technology acquisition the discovery of such conflicts caused substantial changes to the deal, including in increase in the escrow amount, a lengthening of the escrow period and an increase of 40% in the limit for additional claims. Mark advises companies to understand and address their license conflict issues as early as possible.

Cloud computing is creating a whole new world of issues around open source licensing. Many open source software vendors may need to re-evaluate their licensing strategies to take cloud computing into consideration.

The software industry is seeing increased scrutiny of use of OSS in its supply chains. As noted above some efforts are under way to develop standards for supply chain reporting. Vendors selling into supply chain situations should be prepared to provide detailed reports on the software, licenses and obligations incorporated in their software.

The trend for greater scrutiny in M&A continues (see above example). It is more important than ever that companies address their OSS (and any other licensing) issues before entering an M&A transaction.

There is a great deal of discussion underway in the OSS legal community regarding the use of assignment vs. licensing in contribution agreements. There are pros and cons, so it is advisable to understand how these approaches pertain to your business situation.

The Software Freedom Law Center has become more aggressive, as evidenced by their suits against 14 consumer products companies, including Samsung and Best Buy. Not only are they becoming more aggressive in filing suits, but they are also requiring the appointment an OSS compliance officer as part of the settlements.

As you can see, 2010 will certainly be another “interesting” year in the open source business and legal arena.

Panel Discussion: Private Sector CIOs

Colin Bodell, VP Web Platforms, Amazon
Tim Golden, SVP IT, Bank of America
Yuvi Kochar, CIO, The Washington Post Company

Q: How are your companies using open source software (OSS) and why?

Yuvi: We are primarily using OSS on the infrastructure side with gradually more adoption up the stack. Washington Post is highly diversified group of companies (not many people know that Kaplan contributes more than half of our revenue). We de-centralize IT, so we have just about every software there is in our portfolio. OSS adoption is highest in the businesses under significant financial difficulty. At Newsweek, adoption is high and at Kaplan lower. In most cases the argument for OSS starts with cost reduction, and then as sophistication increases, the drivers tend to become more about functionality, etc.

Tim: At B of A, OSS is pervasive – we use OSS in just about every application domain. For us, the driver to use OSS usually depends upon who is making the decision (Central IT or a Line of Business) or the pressure inherent in a particular Line of Business (i.e. cost, functionality, time to market). Right now, B of A technology is decentralizing, so I expect that many more of these decisions will be driven by the Lines of Business and be based upon time to market pressures.

Colin: I run the server infrastructure for the retail and cloud businesses, and sit on the company's Open Source Review Board. Amazon runs on OSS. We build a lot of our own applications, but if something has been done by someone else and works, we use it. For us OSS is the most expeditious way to get technology deployed for the business. Cloud is a real frontier – we are making it up as we go, and this requires lots of quick innovation.

Q: Do you see OSS as primarily strategic or tactical?

Yuvi: We think of it as strategic. We consider it on every project we implement.

Q: How do you manage this or plan to?

Tim: There is no significant difference in our approach to OSS or proprietary software. We try to find the best solution, period. SaaS is coming on very strong; however, none of our deployed SaaS applications are currently OSS.

Q: What are biggest lessons you've learned so far?

Tim: We seem to make the same "if we build it they will come" mistake over and over again. We tend to realize a higher incidence of adoption and success when we mandate that an OSS component or application must align to an existing use case and user base. Another major challenge we struggle with is Enterprise Licensing Agreements (ELAs). We are continually searching for methods to better position OSS contractually as compared to our larger entrenched ISV partners.

Q: Do you see OSS as primarily strategic or tactical? What would you have done differently?

Colin: If you try to walk under banner of OSS, you run into problems with everyone's different idea of what OSS is. We have often encountered very strong beliefs in openness and giving back to the community may conflict with our business objectives. As a result we've shifted from revolutionary model to more of an evolutionary model, less about the banner of OSS and more about optimizing decisions for business.

Q: Is adoption of OSS now different than adoption cycles in the past?

Colin: We are only 15 years old, so we really don't have a lot of historical perspective. I would guess that it is probably not that different. Everything is customer centric for us, so decisions are balanced compromises to optimize that.

Yuvi: I think we've moved beyond thinking that OSS is different. Now that we have a pretty broad understanding of what is OSS, its all about better functionality, quicker, great support. Security is not a critical issue when considering open source. Cloud computing presents a real opportunity for us to re-innovate: not so much in technology as in licensing and distribution.

Tim: I have lived through 4-5 major technology transitions over the course of my 26 year career, and I have a different perspective. These earlier 'transitions' were macro. That is, they were driven by a few large companies pushing a few key innovative products. By comparison, the OSS adoption cycle has been much more broad-based with 200+ players all jockeying for position, relevance, and ultimately acquisition (vs. market dominance), In my opinion, this has made the industry's transition to OSS a bumpier road marked by confusion and outright contention in those software domains occupied by many players with little differentiation between their products.

Colin: We tend to look at it in two ways: How code is created - it's well understood how to leverage this - and how it is consumed. The latter is becoming more straightforward for us and it's mostly about whether the license is compatible with our business? We spend most of our time looking at the license issues: Working from a list of "blessed licenses" and "prohibited licenses" and grey areas that need to be looked at carefully (maybe even case-by-case). Sometimes we find a commercial license is available where the OSS license is not acceptable.

Q: How important is licensing and risk-mitigation?

Yuvi: We take these on a case by case basis. We don't have a policy, approved license list, etc. We try to go with more mainstream, proven products.

Q: What one thing would you recommend to OSS vendors?

Tim: I would like them to understand how to approach a large banking institution. Quantify the client's budget before you invest in a long sales campaign. Learn to utilize software ELAs. This is a primary weapon used by my larger ISVs and hardware OEMs. Resist the temptation to get as much as you can all up front in one big bang. Use the distribution advantages you enjoy as a result of the OSS model. For example, ELAs that permit and encourage free interim use with a 'true up' in three, six or twelve months might be very effective.

Colin: It's all about delivering value (nothing else). I prefer introductions through Linked-In or personal referrals. Don't lead with OSS, how its built, etc. Tell me how it solves a problem that we haven't already solved, e.g. the niche solutions that we don't have time to address. For instance, Black Duck is now working with us to review the IP of Zappos, but it took several years to find the right opportunity and need.

Q: OSS seems to be prolific in your infrastructure level. Why is OSS lagging at higher levels of software?

Yuvi: We look at all sources of solutions, if there fewer OSS wins in applications; it reflects competitiveness of the value propositions available at that level.

Colin: We run our business on OSS, but the applications tend to be differentiators for us and custom to our business.

Q: What do you see as your corporate responsibility to the community?

Colin: This is probably one of the biggest issues with our engineering staff. When engineers say they want to contribute back, we need to know why. What is the value of giving away our efforts to the community? Typically things like bug fixes benefit us more through community, but we have to look out for IP leakage. Things that are of value to our customers also get released. None of this is interesting to us as an intellectual discussion, though; the issue is, "Wwhat is the value to our business?".

Yuvi: We don't contribute a lot back to the community. Our technical organizations are highly fragmented, so there may be contributions coming from individual business groups.

Q: Do you consistently get or buy support for OSS?

Yuvi: Always.

Collin: We are willing to consider several approaches to support.

Tim: We buy support based upon need using technical difficulty and business risk as our primary factors. If the difficulty or risk is low, we may decide to rely on internal or community support. When the situation requires us to obtain commercial support for an OSS component, we may buy support directly from an ISV or use our support aggregator, OpenLogic.

Q: Do you have an IT policy that requires that appropriate support be in place?

Tim: Yes, absolutely.

Colin: No global policy - each business evaluates its own approach to guaranteeing business continuity.

Q: When you are contracting with an OSS vendor, how important is IP indemnification?

Tim: B of A's answer to this question has evolved over the last five years. In earlier stages of our OSS program, we used to assess IP risk absolutely. Over time, we have shifted to a relative risk model based upon the software, the vendor, and the actual use case. Understanding that IP indemnification is just one of several ways to address legal OSS risk, relative risk analysis can help companies understand that there are a range of approaches that can be used to offset IP risk, not just indemnification.

Panel Discussion: The Cloud and Open Source

Larry Augustin, CEO, SugarCRM
Tim Golden, SVP IT and Cloud Strategy, Bank of America

Larry Augustin's Presentation:

Larry focused on two interesting questions about the cloud computing trend:

- Is cloud computing killing open source software?
- How is cloud computing impacting SaaS?

He examined the first question from three angles:

1. Using Google search term volume as a metric for level of active interest
    - The term "SaaS" is increasing 2004 – 2009
    - The term "cloud computing", is growing faster and has now surpassed SaaS
    - The term "open source software" is down slightly, but it is still 4 times more popular that SaaS or cloud computing
2. What kind of software is used in the cloud environments themselves
    - Cloud computing vendors are building their environments primarily with OSS.
3. What kind of software is being deployed on the cloud?
    - Commercial software vendors have not offered licensing models that are compatible with mass computing, so relatively little proprietary commercial software is being deployed
    - A survey of cloud deployment data (from one of the largest public clouds) shows that more than 90% of VMs on Amazon are Linux VMs

Larry's conclusion is that cloud computing is accelerating the overall use of OSS.

How is Cloud Computing impacting SaaS?

> We've seen the rise of SaaS over the last 10 years or so.
>
> The first generation of SaaS venders delivered software that ran only on the vendors "cloud" through a web browser (universal client) – they solved two problems: they rewrote applications for a web browser interface and made servers cheaper through economies of scale.
>
> The second generation SaaS vendors have re-architected their applications for the cloud, but they only have to solve half the problem because cloud providers provide the infrastructure (running on any cloud).
>
> At Sugar we don't think so much about porting to Windows and Linux, but rather to Amazon, Windows Azure, RackSpace, etc. But our customers are still getting their application as SaaS. The current SaaS vendors will now have to decide whether they are in the application business or the infrastructure business

Larry also offered a very interesting cloud story from a personal angle:

> When I built my house I put in a little data center, 19" racks, special cooling, power - everything I needed. At this point, though, I've turned all of the servers off except one, and I have one little router. The rest of my servers are now out there at the other end of that router. My credit card bill from Amazon costs me less than the power to run my server room used to cost. And the neat thing is I can test out more things. If there's an interesting new piece of OSS software out there, I can spin it up on a server, try it out, throw it away when I'm done, and when I get the bill it's just a couple of bucks.

Tim Golden's Presentation:

Tim Golden discussed Bank of America's enterprise private cloud platform.

Why did B of A decide to build its own cloud?

When we merged with Merrill Lynch, we found that we had similar midrange architectures with similar virtualization, consolidation, and environment pre-provisioning goals. When we added up and normalized these architectures, we realized that our next generation midrange computing platform looked a lot like a cloud.

Our initial design focused on solving four problems:

- A scalable architecture in which converged server, disk, and network resources are deployed as an integrated components
- Automated request and fulfillment processes
- Intelligent, automated workload placement
- Policy-driven workload management

One key initial challenge we faced is the fact that most software is designed to be built once and left in place forever. We spent a lot of time learning how to boot OS images, launch applications, and pull data from external NAS disk resources.

Next, we focused on training our OS and applications to operate statelessly, In the stateless mode of operation, server operating systems and applications are booted and launched from gold images. Once they are stateless, hundreds of servers can share these images – they do not need there own copy. Each server's state – defined as 'knowledge of what happened previously' – is stored on external NAS in a specific are allocated to each server.

This design has led to many fewer moving parts in the cloud solution. To instantiate or recover a server image in our cloud environment, the workload manager and provisioning engine (working together), simply need to allocate the server resources (CPU and memory), boot from the OS gold image, mount the 'state' volume, and launch the application from a mounted file system that contains the shared application binaries. This entire process takes only six to eight minutes.

As I just mentioned, shared application binaries – or stateless application operation is another key advantage for us. We store our stateless applications in EFS (Enterprise File System). It's basically a NAS volume with a managed namespace built on top of it - somewhat like DCE/DFS. With EFS, we have bypassed the need to support many disparate packaging and distribution methods. Software is deployed directly to the EFS depot which is then replicated into caches located throughout our server environment. Software upgrades and maintenance change control activities now consist of stopping the application, changing the startup script to point to the new or patched version of the application, and restarting the application. This process takes just minutes. Rollback is just playback of the same process in reverse,

In closing, I would like to offer the following observation. Throughout this conference much of the discussion has focused upon SaaS and external public cloud providers offering IaaS/PaaS solutions. Unfortunately, each of the leading SaaS/cloud providers requires end users to embrace proprietary or unique approaches to software packaging to effect distribution to their respective environments. While there are many vendors working on cross-provider portability (ala CloudSwitch), this disparity makes it difficult for enterprises to consume services and support operations from a variety of providers at once. Enter the enterprise private cloud. While everyone is focused upon answering the question 'how quickly will enterprises adopt external cloud services?', we think about this problem from the other direction. That is, how soon will software suppliers learn to package products as portable software appliances and embrace enterprise private clouds?

Q: What are the key issues for OSS vendors:

Tim: Systems management tools are a huge pain in cloud operations. Many of these tools require manual installation, configuration, and de-provisioning, These are undesirable behaviors. Performance, metering, and monitoring tools that do not understand the nuances of virtualized workloads are similarly

aggravating. Lastly, there is a profound lack of tools that are designed to effectively collect, store, manage, and report on cloud metadata. This is especially important for larger regulated entities that must be able to illustrate usage, configuration, and transaction states at various points in time.

Q: Does BofA have any plans to participate in one of the cloud interface standardization efforts? (LibCloud or DeltaCloud)

Tim: We are still looking at this and trying to understand how to go about it. Quite frankly, our image-based stateless approach to cloud computing is just only now starting to be reflected in approaches being used by larger commercial providers. To further drive standards, we'd first need to understand whose strategy most closely aligns to ours. Towards this end, we've issued an RFI to a select group of suppliers/organizations. But, this is still vry much a work in progress.

Larry: Getting worse as we move from IaaS to PaaS. Now there is much, much more to standardize, like SQL as a service, Windows services, etc.

Q: You talked about cloud agnostic virtual appliances. Do Sugar end users care about cloud-agnostic virtual appliance or are they just interested in managed SaaS application

Larry: Many end users are interested in choice, flexibility, service level. Different cloud providers offer different levels of services and different choices to the end user. The SaaS appliance allows us to address and serve a whole slew of markets that the generic one-size-fits-all service would not address.

Q: I agree with you about OSS and Infrastructure, but in our conversations about mobile we see that the consumer doesn't care if an app is OSS. How do you see OSS moving to consumer apps or enterprise software apps where the subject matter expert is not a developer. How does it get developed, deployed, used?

Larry: I agree about consumers, but enterprises tends to be more educated about issues like vendor lock in, and usually have a check-list about multiple sources, etc. There are some great OSS platforms for mobile. Sugar is now in beta with its multi-platform mobile client (sync, offline, Android, iPhone, RIM). This is all built on Appcelerator and Funambol – both open source mobile platforms. The only way for us to get there was these great OSS platforms. The apps don't necessarily need to be OSS, but the OSS platforms are essential.

Q: What are the primary security risks that you are concerned with?

Tim: We are still looking at this and trying to understand how to go about it. Quite frankly, our image-based stateless approach to cloud computing is just only now starting to be reflected in approaches being used by larger commercial providers. To further drive standards, we'd first need to understand whose strategy most closely aligns with ours. Towards this end, we've issued an RFI to a select group of suppliers/organizations. But, this is still very much a work in progress.

A number of data issues remain unsolved: PCIS, compartmentalizing, scrubbing on de-provisioning.

We also have general concerns on public clouds

Q: How do you deal with software maintenance?

Tim: In our system we update a shared OS gold images or application binary and basically, we're done. The change is automatically replicated and cached to our server environments. Change control (activation of the patched OS image or application binary) can be accomplished later using the workload manager.

Q: What business models do you see the software vendors adopting for cloud use?

Tim: In the short term, nothing has really changed. It's still software consuming network, CPU, memory, and disk resources. The main difference for B of A is that we can now more precisely match workload to these resources. This means we do not pay for excess idle license capacity or resources. Any vendor that designs their products or services to allow us to fully take advantage of this benefit (and others) will be viewed as a 'good citizen'. Ultimately, we want to get away from the shelf-licensing model altogether and move to a model based solely on usage.

Q: We all seem to agree that there is more OSS because of cloud in the near term. In the longer term what incentives exist in the new environment to launch new OSS business initiatives?

Larry: at least one assumption: OSS is driven by the companies that bring OSS to market. Though the funded start-up model has been good to OSS, I'm not sure I agree with this in general.

With SaaS, companies don't rely on source distribution to spur adoption via the "freemium" model, but OSS models are still useful for other companies to drive distribution.

OSS is still a critically useful model for opening up and interoperating with other elements. For instance, B of A is looking to release some of their code as OSS to drive down their costs.

Panel Discussion: Public Sector CIOs

Brad Wheeler, CIO, Indiana University
Shelton Wagener, CIO, University of California Berkeley
Chris Vein, CIO, City and County of San Francisco

Q: Where are you in the evolution of OSS?

Brad: At Indiana University we are eight campuses, 107,000 students, 17,000 faculty and staff and very centralized. We use open source at the core, though not as far as I would like to go yet. We've not made much progress with OSS on the desktop, but we are working on a lot of desktop virtualization. We have about $120 Million /yr IT budget. We've been very involved with other Universities pulling together investment in some of the big systems that we need, such as ERP, teaching and learning systems, etc.

Shell: Berkeley is a community of 50,000 (24,000 undergraduates, lots of graduate students, 15,000 faculty and staff). Our technology footprint is similar to other large research institutions with an average of 1.3 computers/student, 70,000 nodes on the campus and 35,000 different funding sources. We are a significant contributor to the community and open source both at an individual level and an institutional level.

Chis: In San Francisco we are about as opposite to these other institutions as could be: we are completely de-centralized. I am the first CIO for the city that even has any authority to make some sense of this wild, wild west we have in the City of San Francisco. The City is essentially a $6.5 billion/year "company" with six lines of business, 28,000 employees, we spend about $250million/year on IT. My job is to set the technology vision for the city, make sure the policies and procedures are in place to get there and then make sure the centralized IT organization is there to provide the services that are needed. One of the first things I did was to create a small R&D organization within the department. That was because we were so busy fighting about the big systems, that no-one was focusing on giving tools to the departments that so desperately needed help. We created this small R&D group and a sandbox environment and began looking at open source tools. Through a grass-roots process a group of users and developers has grown to the point that the policy making body has now passed a policy that any department buying software for more that $100,000 must look at an open source solution during the procurement process. We provided a framework to make that evaluation, but did not mandate that framework

Shell: To build on what you said about becoming a "bad guy" and having to stop some activities. We did not do that, but the world's economy has done that for us. There is a growing realization that if we don't have leverage over the purchases, we can't get out of the hole we are all in. Technology is being seen as an enabler to redefine those business processes in the public sector that historically have been taboo to touch because of political 3rd rails or embedded constituencies. It is now being look to as an infrastructure layer that will get past a lot of those political barriers because the money is no longer there. I don't believe that you need a passionate leader like the mayor of San Francisco to support OSS. I have a pre-eminent physicist as a chancellor at Berkeley. He doesn't care about OSS, he cares about open and about innovation. What he really cares about is innovation and he understands that anything that we implement today will be at end of life much sooner than anything that we have ever done in the past. Most of my solutions at the enterprise level are older than half of the people in this room. Many of these systems were written in the 1970s running an institution that re-invents itself every 180 days. How is it possible those are still there? Because IT was never really viewed as an innovation engine – it was viewed as plumbing – and that is really changing a lot now.

Q: Do you have any kind of procurement policy like the City does?

We have approached controlling procurement, but in a public procurement process it is a challenging thing. Procurement policies are governed by the Office of the President which is almost at the same level as the State of California. We've been able to dove-tail off of those and add some additional criteria. The easiest, least political tip of the spear has been security, so we have established some implemented some minimum security standards. We require, for example, open code review. This tends to eliminate a

lot of folks from playing if they are not willing to allow for open code review. We've found that to be quite successful because I have exception authority at my desk, and I have to sign off for all purchases over $5,000.

Brad: The way it works for us is that I have absolute authority from the IU trustees over our networks. Departments may buy things but they may not be working for very long. Over time the organization has learned its lessons from that authority.

We use the term community source for open source projects (typically under the Apache license) that are developed in an aggregated model. Higher Ed did not have a money problem, we had a coordination problem. We were spending unspeakable amounts of money on things, but every institution was spending in fragmented ways. Supply side benefited from acquisition after acquisition, but demand side was absolutely fractured with no power. This was for functionality that was core to our mission, like the teaching and learning systems. Our thought was instead of licensing this from firms that were subjecting us to bad behavior, could we use open source methodologies to develop an alternative. It was a long way from that idea to an enterprise scale open source system that had a commercial ecosystem behind it for support - This project was called the Sakai Project. Mark Radcliff helped us a lot with this. We had to educate every university lawyer one at a time about open source licensing, and every technology transfer officer about sharing software. We learned a lot really fast about universities being able to give developers tools, to commit money to 3rd party development. We sorted out all of that in just a few years. I should mention that Berkeley is also part of this project and running Sakai.

We have subsequently developed a follow-on project for ERP. The money is in the ERP systems. Conservative estimates say higher education will spend $5 billion in the first five years of this century on their ERP applications. Its bad and it's the gift that keeps on giving. We decided not to purchase the PeopleSoft financial module at an estimated cost of $30 million. We took our home grown financial system, rounded up about $7 million in investment from other universities and built Kuali. That system is not in production yet. The University of Colorado recently put this system in for $2 million. Our total cost in this new system was about $5 million, saving us $18 million. Still it's an uphill battle convincing CIOs at Universities to go with Kuali when they can spend $25 million on Oracle because nobody ever got fired for buying Oracle.

Shel: Now my reality is that I can't spend $25 million to do that. But to Brad's point, the money is there because it is not leveraged. If we could achieve the same thing with $4 million we would choose to put the difference back into the students and the classrooms. Education is chronically under-funded.

Another way that we do that at Berkeley is through student involvement: our students gain an enormous amount of work experience, visibility and learning opportunity by engaging with open source. We are seeing students creating and innovating applications, content tools, through the whole stack (and increasingly in the mobile world) as stepping stones in their careers. If we have a competition for applications; one of the things that we require is that the content be submitted in an open source fashion. We offer license suggestions but we don't require a particular license. We have an enormous number of contributors that are in the form of individual students. My organization supports institutional contributions by encouraging everybody to always contribute back. Most of my employees tend to contribute back as an individual. I see a huge opportunity if we can find a way to keep the cloud contribution model engaged. If we put the cloud behind the wall and we close off the contributions channels, we're going to lose an innovation engine that we can't afford to loose. We want to make sure that all of these students and other developers have a means to make a name for themselves and can add intellectual contributions to each of these projects. There are a number of approaches one of those being that you can make your cloud offering fully extractable. We run about equal parts commercial software and open source software. There is absolutely no inherent bias – it's about the fastest path to features and solutions. Higher Ed is not particularly lucrative market for software because of the margins and our niche challenges. Open source solutions can often be adjusted and added to our other systems. We just released our first open source solution as a SaaS offering for higher education. We are running that as a community SaaS with 10 partner institutions.

Q: Chris, you are relatively early on in your OSS activities. How would you characterize where you are and what are some of your thoughts and plans for moving that forward?

Chris: My management approach is keep throwing stuff at the wall and eventually something will stick. We are finding that applications like SugarCRM are sticking, and they are actually exciting in the way that they are meeting business needs. What we are doing is leveraging that by creating a whole and mostly new platform. One of the first things we are doing is creating APIs. What we've just created with Boston and Washington DC, is an open 311 API. It is vendor agnostic on the bottom, but using social networking tools and portals into a call center. We now have seven cities working together to develop some rules and regulations and sharing them among cities to provide a community network for releasing data to the public, and getting the public to write their own applications to use the data to extend access to government services. What's exciting is that it is taking off, not just in the US but internationally as well. We see this as an effective force to start changing the public sector marketplace.

Q: Over the last couple of years enterprise adoption has really started from the ground up. Is that the way it has been with the city or has it been more of a top-down approach?

Chris: Probably a little bit of both. We came up with a vision and started to force it through policies, but the carrot really comes from getting individual programmers or application providers excited within the marketplace. We don't have a lot to excite people in government, so the recognition is appealing to their ego is about the only thing you can do. So we are doing it top down and bottom-up, but it's probably the bottom up that is most sustainable and scalable.

Brad: Shel and I will both agree on that last point. Our staff engagement in open source projects has been great for developing our people. Those people that learn to work with distributed teams over which they have no authority and still work in the reality triangle of pressure, resource and timeline, have all been promoted. It has been a tremendous HR development for our team.

The universities in these community projects are not looking to monetize their contributions. The core foundations are simple coordinating entities that hold the IP, coordinate conferences and do some QA. If there is money to be made in these endeavors that is left to the commercial members of the foundation. We made the choice early on in the design of these organizations that there is no differentiation between commercial members and educational members. In this way we are trying to shift the mentality out of a fairly hostile procurement mechanism into more of an ecosystem that can work.

Shel: It is clear to me that bonuses for completion of internal projects don't hold nearly the same level of professional interest for our developers as participation in broader activities that expose them and their successes to larger communities. One of our biggest problems is employee retention in a very competitive market. We can't compete with private industry in salaries, so these community incentives are really important.

Q: As you look forward 12 – 24 months, aside from the financial incentives of consuming more open source what other benefits do want to see the city accrue?

Complete business transformation. The City has all the same issues that we've talked about. Legacy systems, $100 million to replace a general ledger system, we just don't have the money to do that. What we need to do is rethink the delivery of government services. And we think that the OSS community has matured enough that we can introduce open source as a bona fide systems capability. We have that opportunity that we've never had before, and we're using the budget crises as a way to introduce that change.

Another thing we struggle with is that we are 100% civil service and 100% unionized. How do you convince a union employee that this type of open development is actually a good thing and not a bad thing? How do I get them to understand the vision that we all see and make that resonate for individuals that are just scared about losing their jobs?

Q: We have government and education as customers and one of the things that seems counter-intuitive, is that cost savings for them does not resonate as strongly as for the for-profit customers. Why is this?

Brad: I see this too. One of the reasons is that making a decision in a university is very difficult – a vote of 23 to 1 is a tie. Often there is not a point of authority. At an administrative level there are a lot of people very afraid for their jobs. "If we go open source and project doesn't go as expected, will we lose our jobs?"

Shel: We don't arbitrate decisions by the dollar. It's just not possible in a university environment like ours. In mid-market or smaller institutions they have small pools of administrators that have to do everything. The thought of them being pulled off their other tasks to learn a new environment is very, very challenging. SaaS may be the disruptive activity that can change that because it lowers the barrier to entry.

Q: Chris, as you talk about cities working together is the collaboration at the level of building something or is more about lessons learned or how do we get started?

Yes, we are actually building systems together. We are now looking at how to manage the code base.

Q: In universities is it still the case that commercial vendors offer free use of their products so that students get used to them?

Shel: One of the biggest problems that I have is that we get too much donated. Every company in Silicon Valley gives me their stuff and I end up having to run one of every one of those things. The volume of donations is very high, but the leverage on those products is very low.

Panel Discussion: Open Source M&A

Mark Brewer, SpringSource Division of VMware
Damian Eastwood formerly of Sun Microsystems

Q: Can you give us some background about where you are today?

Mark: I've been involved in a few acquisitions both in the selling side and the buying side – six in the last two years. First when Covalent was acquired by Spring Source. Spring Source acquired G2One then Hyperic, which was actually part of Covalent in its early days, and finally we acquired Cloud Foundry right as we were being acquired by VMware. Since that acquisition by VMware we closed our first transaction, Rabbit Technologies which has an open source messaging system.

Damian: 16 years at Sun where I supported all aspects of the business including acquisitions. First acquisition was an app server company back in 1998. The last three transactions represented just over $8 billion.

I thought I'd chat about three transactions: the acquisition of MySQL by Sun, a failed transaction to sell Sun to another company, and finally, the acquisition of Sun by Oracle.

Most of you probably read of the Sun-IBM transaction as it unfolded. This was an absolutely bizarre event – I've been involved in many transactions public and private on both the buy side and the sell side and typically confidentiality is key. You may have noticed that there has been an insider trading charge against an IBM insider who apparently leaked the information. The Sun-IBM discussions were bizarre because everything was being leaked to the public. Every morning I'd check the Wall Street Journal and it was surprising the information that was being leaked.

Mark: I thought I'd talk a bit about the failure that we just experienced. SpringSource was in the middle of three transactions at the same time and that was probably one of the reasons for the failure. The real reason that the deal fell through was that we failed to understand the real motivations of the founders. It seems pretty basic but sometime you get lost in negotiating with the investors and in this case the CEO of the company was not a founder. We just didn't really connect with the founders very well. If you look at our successful acquisitions, we did a great job making sure we know who were the people who built this technology, how important are they to retain and what are their motivations (are they looking for money? Are they looking to grow their product or project?) We got to the last minute of the no-shop clause (and they had a short one), and they seemed to have taken advantage of an offer on the table to get more funding.

If your selling your company to another company, especially a public company, make sure you know what your employees motivations are, especially the ones that need to be retained. It often isn't only cash or stock - that may be part of it, but not be the most important factor.

One last comment on earn-out: try to get most of the details on the earn-out in the LOI. I've pulled the plug on a deal for exactly this reason when the earn-out term turned out to be ugly.

Q: When you're getting to know a buyer or seller, are you talking about formal interactions or informal interactions?

Mark: You're really going to learn the most in informal interactions. When we're looking to acquire a company we might send one of our technologists to really get to know their key engineers. With Rabbit we were dealing with a fairly young OSS project. They didn't have much revenue, a dozen engineers and one business person. We had each engineer meet with one of the engineers on the SpringSource side. Sure I met with them, but I didn't matter that much. It was more about them feeling comfortable that their technology was going to be taken forward and that is was going to have a key place within SpringSource and VMware. So I would say that understanding the motivations on an individual basis through the informal interactions was really important.

Q: When you were interacting with MySQL were you try to have your team have the same kind of interactions?

Damian: In informal discussion, but doing this was a part of our formal process. Yes, that's sort of a standard approach that we take to have face time interactions – you sit down and say "Tell me the story about your business, tell me about your technology, who are your main players who are the key developers…"

From a process perspective, this is how a transaction unfolds:

1. Sign a CDA
2. Enter into discussions
3. Sign a term sheet, MOU, LOI the fundamental
4. A document request goes out and that's the start of the due diligence process
5. In parallel the definitive agreement (DA) is negotiated

Depending on whether it's a public or private transaction you'll have between 10 days to 10 weeks. Personally I prefer public days because it's 10 days, you're in, you're out, you're done, because it's an absolute drain on resources. It's a double or triple job. My advice is to comp whoever's on it. If you don't want them to be doing the job correctly not worrying about their exit strategy, tell them up front.

With MySQL we had a pretty open kimono discussion with Marten. Sale was not their exit strategy so we had to make the offer attractive. Marten gave us insight about the people, about the processes, about who to keep and who not to keep. There were two big-named people, Marten and Monte, who were most visible to the outside world. The decision was made that Marten was the most important to keep – he was the public face and was very well liked and recognized as the person who had made the company what it became. Obviously we did not understand some of the motivations, because Marten stayed for less than a year.

The acquisition was interesting. There are all sorts of games that are played: you've got lawyers, bankers and wankers. My approach is to get everything right out on the table up front and let people figure it out. If you've got a big red herring or a smoking gun, get it out up front. If you want your price in the term sheet, you want certain terms in the term sheet, if you've got a big issue for DD get it out on the table. It will come out by the end, I guarantee it. The acquirer will not give you an exception to the reps and warrantees unless you provide all of the documentation they want.

Q: Can we get your perspective on SpringSource's transaction with VMware?

Mark: With the VMware acquisition of SpringSource we had a very quick transaction. The diligence process was accelerated, and we had to get people to work 24 hours a day just to get through it. It was very important to get Purchase Agreement's "Schedules" right, because whatever is not in the acquiree's schedule becomes their liability, and that liability is usually taken out of their escrow. But that was probably the most pleasant transaction I've been involved with.

One thing was surprising to me in our acquisition of Hyperic: it is important to know not only your open source licenses, but also the terms of your commercial licenses to customers. We discovered that Hyperic had a standard practice to give uncapped IP infringement indemnification. That used to be the standard practice for most commercial software companies, but it's not even the standard for most startups anymore. I guarantee that any public company will not be pleased to find that you are providing uncapped IP infringement protection not only on your bits but to all the open source bits that you use. We ended up having to change the whole transaction from a full purchase to an asset purchase because of all of these "toxic assets,". We ended not buying some of the old customers because of those contracts. If you have such terms and you have the opportunity to change them, you should definitely do that before trying to sell your company. Obviously make sure you know your open source licenses and that you are in compliance. I would encourage you before you consider selling your company get Black Duck or Palamida to do a full scan on your software ahead of time. It's worth the money spent. It's going to be done to your code anyway, but knowing what's in there ahead of time and better yet, fixing issues, will save you a lot.

Q: What would you say is the acceptable limit of risk in a transaction?

Damian: It's all over the map –different companies have different risk profiles. When we did the MySQL transaction there was one technology that was pretty critical to the product, InnoDB. InnoDB was a small Finnish company that was acquired by Oracle. We had a big diligence issue, so the question was what is the real issue and what is the resolution. In this case MySQL refused to show us the document from Oracle, so we had no idea what was going on in that relationship between MySQL and Oracle around InnoDB. So we had to do an analysis to understand the risk profile. Ours was "what is the impact on the business, what is the revenue that's tied to the transaction engine, what are the alternative transaction engines currently available and planned? We tried to understand the revenue impact, the number of customers affected? We also had to look at the architectural issues and what would happen to revenue if MySQL lost the right to use the InnoDB engine? So you can see we had a pretty flexible risk profile.

In the case of the failed Sun - IBM transaction, IBM wanted to see everything. No stone was left unturned.

Q: What is an acceptable limit to IP indemnification?

Mark: From VMware's perspective, for anything that has open source in it you should say that the open source is carved out. If you do provide protection, you should limit it to 2x or 3x of fees paid and if you can limit it to the last 12 months, even better. If you have uncapped liability or if you don't at least address open source differently than the software that you own, you will have problems.

Also, when you are in the process of negotiating deal terms, make sure that you understand as much as possible the motivation of the buyer. It's not always a multiple of revenue or if it's a technology buy, some multiple of investment, though that's what you'll end up talking about in the negotiation. For SpringSource selling to VMware we clearly understood that Paul Maritz and the rest of the leadership at VMware saw us as a strategic, game changing move, and that, of course, got us a very high multiple. Some people thought the multiple we got was absurd, but it wasn't absurd because VMware saw this as a way to literally change who they were and how they address the market.

Q: At what point or sale value do you need a banker vs. doing the deal yourself?

Mark: I'm probably the wrong person to ask. I had a banker for a deal where I was trying to sell Covalent in the mid 2000s. The banker got lots of introductions, but I did not close a transaction. Everything I've done since then, either buying or selling, was without a banker.

I would say the question is: are you or your investors capable of negotiating the terms of a deal? If you don't have somebody that has gone through this or know the individual buyer and how to negotiate with them, you should bring somebody in.

Brainstorming Sessions

The heart of the conference was the three brainstorming sessions. Each session commenced with the introduction of a business case (which was distributed to the audience the week before the event), after which the attendees broke into groups of 10-12 participants to brainstorm ideas and solutions. Each team was asked to discuss and prepare a presentation for each question in the business case. One person on the team was "elected" to present the team's analysis at the end of the session. Below we have attempted to capture and summarize the common threads and responses from each of the groups. Where a single group may have had a unique and insightful response to a particular question we have noted it.

The suggested approach was to review the case to make sure everyone understood the premise and agreed on the assumptions, core elements, issues and priorities. Groups were not required to answer all the questions for each business case, but to be sure that any responses provided were adequately supported.

Mobile and Open Source Business Case Workshop Summary

Introduction by Bill Weinberg, Senior Executive, Olliance Group

Selecting a Mobile Platform for Application Development (v4)

The mobile wireless ecosystem is increasingly dominated by open applications platforms, some of which are also open source environments (per recognized norms, e.g., OSI definition). Whereas several years ago, most of the "action" focused on defining and building *platforms*, today most of the industry focuses on enabling, building and commercializing mobile *applications*.

This business case examined factors in going to market with mobile applications, and/or rolling them out internally. For the complete instructions for this brainstorm session visit the Think Tank Website.

Workshop Results

Key take-aways from the exercise included

- While mobile platforms are increasingly open (e.g, Android, Symbian), the choice of open vs. closed licensing is entirely orthogonal to platform licensing
- Open source mobile applications are desirable, but are trumped by open, extensible APIs
- Cross platform interoperability, while important, is not as critical as end-user experience
- The Cloud is increasingly important for deployment of mobile applications
- The iTunes App Store may represent a bottleneck to deployment, but is worth the investment, for its own sake and for access to the popular, high growth iPhone platform

Each of the following roles was selected by two groups: Commercial ISVs, Small Software Developers, Enterprise / SMB IT Departments, Financial Services Organization. No groups chose Mobile System Operator.

Commercial ISVs

The groups that represented Commercial ISVs positioned a mobile alerting application and a mobile helpdesk application respectively. The AlertME application is implemented in HTML in order to be ubiquitous and platform independent, and it serves as the client interface to the company's alerting subscription service which exists to drive revenue. The server side of the application runs in the cloud and is closed source. The client is open source (it's HTML), and all APIs for both are open. For them openness is important to encourage community support for integration with other applications and for

localization. The group expects their primary channels to be Salesforce.com and other data stream providers, and the hope that the openness of their client and APIs will spur integration that creates additional channels.

The Mobile Helpdesk group targets the iPhone, Symbian and Android platforms in order to achieve a wide customer base. Their business model is to drive demand for their company's helpdesk products and services and drive brand extension. To that end the mobile application is free and available as open source under the GPL license. They chose software platforms that deliver "snappy" native client applications on the most widely used smart phones in their markets. They view the developer community as very important for enabling end customer customization. Their primary channel will be their existing helpdesk sales channels, though they will also make the client application available for free on the app stores for each platform.

Small Software Developers

The two groups that represented Small Software Developers both created combined social media/game applications. The first posited an application called "Little Fuzzy" which is a multi-player game integrated with Twitter and Facebook. It is targeted to the iPhone platform for its initial North American target market due to best match with the target consumer demographics and the platform's features and capabilities. The product is implemented with a mobile client UI linked to a multi-player server hosted on the Amazon cloud. Although this group thought that open APIs were important, they felt open source was irrelevant to their strategy. They planned to go to market through the iTunes Store using the leading social networking sites for marketing.

The second small software developer group presented an extendable, massively parallel fantasy world game that was also to be integrated with social media. This group targeted iPhone and Android which they chose for functionality and popularity. The mobile device provided a client interface to their multi-player cloud server. Their mobile client will not be open source, but the fantasy world server is open for hobbyist and professional 3rd party development. The revenue model is to include subscription fees from players, sales of virtual goods and brand licensing fees. The latter is a unique element of their strategy, leveraging the extendable nature of their "world" to enable branded domains such as "Spiderman." The group plans to go to market through the platform app stores, and team with branded partners for marketing.

Enterprise/SMB IT Departments

One group presented a plan for a medium-sized business with 100 sales staff. Their company is currently using RIM clients and a SaaS solution for SFA. Their plan is to implement mobile access to their existing SaaS system. The goals of the project are to increase sales' revenue generation and to minimize the complexity of the implementation and training for the new capability. The application is considered a competitive advantage and will therefore be closed. It's not important to them that the RIM platform isn't open source, but the existence of many business applications and availability of many capable systems integrators is. Since this is an internal application there is no go-to-market element.

The second group presented a plan for a mobile sales automation application for a large multi-national enterprise with thousands of sales reps. Because the company is global, they must support different platforms in different geos, and all platforms must have custom application environments that will make caching for off-line use possible. They chose RIM, iPhone and Android. The business goals are to increase the efficiency of their sales force and make their multi-national deployment as easy as possible. Although this group favors open source platforms, this "feature" is not uniformly available, nor is it the most important selection criteria, but they would very much like to see standardization of APIs driven by a consortium of customers. They do plan to leverage community development where possible, contributing back bug exes, etc, but not anything related to their proprietary business processes.

Financial Services Organization

The two groups representing financial services organizations chose to be PayPal (does this make PayPal the brand more open source CEO's wish they could be?). These groups aim for ubiquity and plan to leverage developer communities, so they plan to support iPhone, Android, RIM and Symbian platforms. The architectures make use of mobile clients and centralized servers for location data, analytics and

transaction processing. Their strategy is to expand PayPal participation in POS activity, increasing their share of transactions and transaction-based revenue. They seen no compelling reason to release their mobile application as OSS, but will provide open, documented APIs to their proprietary financial server systems. Although the mobile applications are not revenue opportunities of themselves, they will use the applications stores for each platform to speed propagation of the applications and leverage existing PayPal vendor relationships to help drive marketing activities.

State of California Business Case Workshop

Introduction by P.K. Agarwal, Chief Technology Officer, State of California and Lee Mosbrucker, Director of Enterprise Architecture, State of California

P.K. Agarwal introduced this workshop with some basic facts about the State of California's IT:

- The public sector is 1/6th of all IT spending
- It is counter cyclical to the business economy
- State and local government IT budgets typically grow 2-4% annually

Governments have very substantial legacy portfolios. In California more than 50% of systems would be considered legacy systems. This means that the best entry point for new technology vendors is where those legacy systems need to be upgraded

California currently spends $3 billion/yr on IT.

The state's IT is a federated organization with about 400 different datacenters operated by 130 departments, commissions or operating units. Every one of these chooses and runs its own applications, and you can bet they guard their independence. This creates a lot of operational complexity

We are working strategically to increase shared services, for instance, consolidating core services like email. We are also working on standards that create economy and interoperability

Looking at business case, you might realize that we are very, very conservative. The public sector is a very conservative sector. The following story may help you understand why:

> Massachusetts decided to standardize a document format, but the decision became very political and in the end the CIO got fired. Some vendors tried to get the legislatures in TX, CA, other states to pass a policy about standard doc format, but none of these ever got passed. Making large changes has proven to be very difficult or impossible. Everyone in the public sector is now very, very careful.

Lee Mosbrucker provided some additional background:

Lee and his group have been trying to create some coherent policy to improve the overall infrastructure. The starting point has been a directive from back in mainframe days which required departments to;

- Identify their software
- Identify risks, etc
- Put this information in a file in a vault

This has at least provided a vehicle for tracking down what software is in use.

The State has officially declared that it is OK to use open source where it is appropriate to business needs.

Other areas that they are working on are: development environments and interoperability segments of architecture. There is an urgent need to create standards and policy in order to create more coherency.

We are truly going after open standards. One important example is in the area of federated identity management. This must be based on an open standard.

The key issue in business case is how do we create these standards and directives? What have you seen work? What doesn't work? The business case provides some background, but ultimately asks where do we go from here?

For the complete instructions for this brainstorm session visit the Think Tank Website.

Workshop Results

This was a very exciting topic and the most highly rated workshop among all the ones presented over the vents five years – the brainstorm groups all had great enthusiasm about how open source software and particularly its community approaches could help the State of California improve the efficiency and effectiveness of its IT. Three key themes emerged from the brainstorm results:

1. Leverage Open Source Community Approaches

One of the most powerful aspects of open source software is its ability to coalesce communities that drive not only the technology, but also the way the technology evolves and propagates. This "bottoms-up" approach is particularly effective in fragmented "markets" where direct control is limited.

One of the key success factors for a community is its leadership, and in an environment like the State of California developing this leadership will take some investment. The most effective approaches for developing community leadership are about recognizing contributors and their contributions. Attention is a powerful motivator and a particularly good incentive model given the monetary limitations of the public sector. Community techniques should be leveraged to create "Rock Stars" – not just developers, and leaders to drive initiatives.

Communities make it easy and safe to use common code, libraries and applications. The key resource for this would be a CAL Forge, a central place to find, share and maintain open source software. The concept of "blessed" OSS addresses the safety requirement. Initially "blessed" might mean only that some agency has approved its use. Standards will naturally solidify in the community over time.

Development of "best practices" for use and management of OSS within the State will be just important as developing software solutions themselves. Communities a very effective way to accomplish this. This can start with the CA agency technologists who are interested in the subject. Once started, the effort can include other states and federal government agencies and even international public sector agencies and communities.

One important point was raised by several of the brainstorming groups: the solution isn't just more OSS; it's as much about open processes and sharing of approaches and code.

2. Plan a Phased Approach

Recognizing the scope and complexity of the state's IT environment, most of the groups recommended a multi-phased approach.

Phase 1: Awareness, visibility, discovery

The first phase is about creating awareness of OSS, visibility for OSS initiatives and discovery of the technical and organizational reality necessary to develop a practical vision and a detailed plan to get to that vision.

The most widely commended step was the creation of an inter-agency community. This would provide a place to communicate the emerging vision and its official blessing. The heart of any community is its forum, the communications tool that allows community members to communicate in a structured way. A forum becomes a self-generating resource as communications accumulate. "Think Tank" style meetings are another effective technique for creating visibility and stimulating involvement and ideas.

The State of California's challenges in implementing OSS are hardly unique, so getting involved with other organizations can provide useful ideas and significant solutions. In particular, it would be beneficial to involve other public sector programs and OSS vendors.

Several brainstorm groups suggested the creation of an NGO "OSS in the Public Sector" organization to provide a common, neutral environment.

In order to create an effective vision and a plan to implement it, the State would need to inventory its code base and develop a clear understanding of agency attitudes and needs. With those facts in hand it will be possible to identify the best opportunities to leverage OSS.

Phase 2. Developing resources and standards

The second phase is about developing the resources (people, policies, processes and tools) to make change possible. Key ideas for this phase included:

With the information gathered in the Discovery Phase, it would become possible to create an overall vision and statement of direction to guide the highly fragmented activities and processes of the 100+ agencies toward a more coherent architecture that makes the best use of OSS.

Several groups recommended establishing an OSS planning and help center. A resource that provides planning support, help with identifying OSS components or applications and support for them. This would help to make up for the lack of big vendor support typical for OSS, but would be cost justified by the lower costs of acquiring OSS vs. traditional proprietary products.

A shared OSS Repository would further simplify OSS evaluation and acquisition by providing all OSS that had been evaluated and approved in one easy to find central place. If agencies find it necessary to customize any of this software, the enhanced versions also become available to everyone. Ultimately, custom applications may also be shared with this facility.

All of the groups talked about the importance of developing guidelines and best practices (that should evolve into formal policy in Phase 3). The key areas that should be covered are:

- Selection and use of OSS
- An ROI analysis model
- Standards for what goes into the common OSS Repository
- A process framework for OSS
    - Evaluation
    - Acquisition
    - Deployment
    - Management and support
    - Procurement guidelines

A key element of a formal OSS process is an OSS Review Board which typically owns the policy and process definitions and major decisions about use and management of OSS. During this phase the internal community OSS advocacy team begins to evolve into an OSS Review Board.

In the timeframe of this phase some successful OSS implementations should be deployed within the State. The best of these should be identified and supported as reference projects/implementations that demonstrate the viability, advantages and safety of using OSS.

One interesting suggestion heard from two of the brainstorm groups was to provide OSS applications and infrastructure in a State of CA Cloud. In this way the advantages of extremely quick deployment and lower long-term cost could be harnessed to seduce agencies to use common infrastructure and applications.

Phase 3. Infrastructure building and replacement

Phase 3 is about solidifying the understanding, vision and capabilities developed in the first two phases into formal policy, process and architecture.

The Guidelines and Best Practices developed in Phase 2 become formal policy enforced by procurement functions and the OSS Review Board.

The formal policy and processes should very strongly favor code reduction and rationalization. Where functionality is common across agencies, common architecture can be mandated. Some typical examples of these would be SOA / ESB, cloud implementations, identity management, etc.

Infrastructure and application consolidation now becomes practical through the use of common architecture and code.

At the culmination of this phase the balanced elements of facilitation and clout (primarily in the hands of the Procurement function) are deployed to drive the State's IT to new levels of integration and cost-effectiveness.

3. Change Management is a Key to Success

Many of the groups talked about the monumental challenges of driving so much change through such a large and fragmented set of agencies. Their suggestion was employing a change management "guru" to guide the evolution. This function would focus on driving "cultural" and organizational change through programs such as

- Education and socialization initiatives
- Establish "poster child" projects
- Make innovation competitive and rewarded
- Job assurance

Summary

The clearest message from all of the brainstorm groups was that broad use of OSS and community techniques can deliver exceptional benefits to State of CA. These include:

- Align IT costs with the state's financial reality
- Empower agility in departments and agencies
- Support interoperability while improving security

Open Source in Cloud Computing Business Case Workshop

Introduction by Miriam Tuerk, Senior Executive, Olliance Group

Challenges and Opportunities for Open Source Resulting from Cloud Computing Trends

The Cloud and Open Source business case will discuss and evaluate the implications of Cloud Computing for Commercial Open Source Vendors. Open Source is clearly a very large part of the Cloud eco-system, and as technology companies and end-users migrate to the Cloud, Commercial Open Source vendors must consider the implications for these customers:

How does Cloud Computing change adoption and use of products and services?

In what ways do vendors need to change product development?
- community activities?
- go-to-market strategy?
- approach to monetization and pricing?

For the complete instructions for this brainstorm session visit the Think Tank Website.

Workshop Results

The resounding conclusion from all of the brainstorming groups was that cloud computing is a very big and important change in the IT environment for open source software companies (and most other software companies as well). Most groups saw substantial opportunity for their companies in this emerging cloud computing marketplace, but recognized the threat it imposed if they did not respond to a fundamentally change in the way customers were buying and deploying software. The results also indicated that cloud computing was more than a simple platform change, and that responding effectively would entail development of new technology, changes to their partner and channels models, new pricing and packaging approaches as well as new approaches to monetization beyond the well-understood download and upgrade models.

Three groups developed a strategy and recommendations for Scenario 1. Two of these groups thought that cloud was both a threat and an opportunity for company ABC, but the group that took a VC perspective thought the cloud issue may be a red herring distracting us from more fundamental problems with the company's business model, in particular, that the company's commercial offering is not strong enough relative to their OSS offering.

All three groups' strategies entailed offering their products for cloud deployment, although the VC-oriented group recommended evaluation of an appliance or an OEM strategy as an alternative.

The first group plans to continue marketing to the same segments with a positioning based on their product's functional strengths (but now available on cloud platforms, too). Another group intends to focus their private cloud offering on 2-3 OEM/service provider partners in each target segment.

All three groups would apply development resources to bring SaaS or cloud offerings to market

All three groups would continue to support and develop their open source communities, but intend to change their open source licensing to Affero GPLv3 to protect their SaaS/cloud businesses.

The groups presented different monetization and pricing strategies. One would just add SaaS subscription pricing to their current dual licensing options. The other plans to price its cloud version aggressively to OEMs and Service providers on a royalty basis in order to establish market presence quickly, and employ up-sell and cross-sell strategies to grow revenue over time.

Four groups developed a strategy and recommendations for Scenario 2. Most of the groups saw the cloud as both opportunity and threat, but one thought that DEF had jeopardized its viability so seriously by ignoring the SaaS and cloud phenomenon for so many years that that the situation was seriously threatening to them.

All four groups presented strategies to add cloud capability to their product lines (but only if DEF could muster the expertise, the brand equity and the cash, cautioned one group). One group was quite bullish about the opportunities the cloud presented to develop several new revenue platforms, including public cloud products, private cloud products and custom cloud solutions through integrator partners.

The bullish group would position their offerings as "the ubiquitous management tools for both on-premises and cloud environments". Another group was most concerned about creating a graceful transition so as not to disrupt their existing business models. A third group felt that they needed to reposition their offering up the value chain in a deployment-neutral way, going from "monitoring servers" to "managing performance."

All of the groups recognized that they needed to develop significant new technology to compete in the cloud market, one called out requirements for new data types, another, the integration with hypervisors and virtualization infrastructure. The group that had started from a community felt that they would have to undertake this development themselves because the community would develop these capabilities to slowly. Another group thought they would probably need to acquire a small cloud-oriented company in order to bring this technology on line fast enough.

Two of the groups would leverage partners heavily, including reseller channels such as virtualization partners and hosting partners for products and systems integrators and other ecosystem players for custom configurations. One group, however, planned to retain its current web/direct sales model leveraging open source "frictionless" distribution.

The three groups that articulated community strategies all wanted to maintain their communities, but take steps to protect their investments in the new cloud technologies. Two would offer their new cloud capabilities only under commercial licenses (at least for some period of time). The third would investigate different OSS licensing options. All three agreed that openly published APIs were essential in any case.

Not surprising, the group that was bullish on new revenue platforms would expand their monetization strategy and packaging options the most adding dual licensing, direct SaaS offerings and professional services to their existing support subscription offerings. They also intend to offer usage based pricing in addition to subscription pricing where there is demand. The group concerned with graceful transition planned to offer their cloud capabilities under a subscription pricing model, similar to their curren support subscription packaging, in order to allow graceful transitions within their accounts. The group looking to reposition itself up the value chain is seeking to change their pricing model to something more delivery oriented such as managed resources or per report.

Event Summary

The Spring 2010 Open Source Think Tank was a resounding success with more enthusiasm and active and diversified participation than any previous year. The change in format to more structured, real-world case studies was received extremely well. As a result we intend to focus on the real-world business case format in all future Thin Tank meetings. The panel presentations by end users, cloud computing experts and senior executives discussing exit strategies and experiences were also clear hits with the audience.

The European Think Tank to be held September 28 and 29 in Paris, will follow the same basic structure with case study participants from Europe focusing on the intersection of cloud computing and open source.

