



OPEN SOURCES #16 - OPEN – FOR BUSINESS

OPEN SOURCE IS NOT JUST FOR HACKERS – BUT MAKES BUSINESS SENSE, TOO!

Open source is not just for the two-person garage shop hackers and it's not just for the masses of racks in Google's data centers. This month Paul Ramsey, both a business professional and an open source developer, explains that it makes sense for the rest of us, too.

Open source software has had a wild, up-and-down ride over the past five years. In the guise of the Linux operating system, open source has moved from the niche tool of graduate students and basement hobbyists to the backbone of enterprises like Google [1] and Credit Suisse [2].

Just as the corporate and marketing footprint of Linux is small compared to that of Microsoft and Apple in the operating system space, the mindshare of open source software in the geospatial realm remains relatively small. Objections and concerns about open source that were discussed in the general IT world some years ago[3] are only now becoming top-of-mind in the geospatial world.

If you are a business manager, or an open source advocate preparing a brief to management, thinking about the following questions and their answers should help you tailor your open source strategy to fit your organization.

“Is there support for open source software? Am I taking on a big risk by using it?”

This problem feels larger than it is, because in the proprietary world the answer is so simple: just go to the vendor.

In the open source world, there are many answers, and the best solution for your organization will probably be a combination of several of them:

- Product support in the standard fixed-price contractual mold, from specialist companies with expertise in particular components (Refractions for PostGIS, DM Solutions for MapServer, OpenGeo for GeoServer).
- Stack support, from generalist companies putting together mixes of components (WhereGroup, LISAssoft).
- Community support, through mailing lists, web forums, and instant messaging. It is not uncommon to hear users who are new to open source comment on the superiority of the open source community to the contractual support they get from proprietary companies.
- Roll-your-own support, seeking out and contracting with project developers, who are often an unappreciated source of top-notch troubleshooting and knowledge.
- Roll-your-own support, using local consultants who have built previous projects with the tools, and have the in-depth expertise that experience provides.

An important feature of open source support is that, because there are multiple channels, you are not bound to a single vendor or pricing model. You are purchasing your support in a competitive marketplace, not from a monopoly provider.

“Are open source systems compatible? Will using them create closed stovepipes?”

Far more than proprietary systems, open source solutions tend to be



Positioning for Success Today. And Tomorrow.



Trimble
DIMENSIONS 2009

The Mirage, Las Vegas
February 23-25, 2009

Discover a World of Solutions—Attend Trimble Dimensions 2009.

Gain insight into how surveying, engineering, construction, mapping, GIS, geospatial and mobile resource management professionals harness the power of today's technology to help face tomorrow's challenges. At Trimble Dimensions 2009 you will have the opportunity to network with key industry leaders, develop new contacts, build partnerships, discuss opportunities and discover how to overcome obstacles in today's competitive business environment. Positioning you for success today. And tomorrow.

To secure your spot, register now at: www.trimbleevents.com

interoperable, both with existing systems and with international standards.

Interoperability is a reflection of the structure of open source development – there is no strategic marketing reason for open source software to not embrace standards and interoperability. Organizations that want to use open source components will often fund the relatively small development effort necessarily to allow it to interoperate with their unique formats or databases. Over time, these small bits of work add up to extremely cross-platform, cross-format, cross-standard software products.

Through the use of international standards (like OGC and ISO specifications) and de facto standards (like KML and map tiling schemes), open source is becoming even more interoperable, even with vendors who have made little explicit effort to create interoperable software.

“Aren’t staff who can work with open source geospatial in short supply?”

Technical skills in general are in short supply, but that doesn’t mean IT is in crisis. Lack of previously experienced staff is a problem only to the extent that:

- the skills required to understand and maintain a piece of software take a long time to learn, and
- the skills required to understand and maintain a piece of software are in short supply.

Note that you have reason for concern only if both conditions occur – the skills must be both difficult to learn and in short supply.

However, the skills necessary to work with open source geospatial applications are either easy to pick up quickly, or transferable from other domains.

- Spatial SQL is a generic skill that allows staff to use ArcSDE, Oracle Spatial, SQL Server Spatial and also the open source PostGIS spatial database.
- JavaScript is a generic skill that extends from the ESRI JavaScript API through Google Maps to the open source OpenLayers map API.
- Internet cartography is more about understanding the relationship between scale dependency, number of features and performance than the dialect of the configuration file, whether it is the ESRI AXL format, the standard SLD format or the open source MapServer map file format.
- Deploying server-side Java applications like ArcIMS or standalone apps like ArcServer is no more involved than deploying open source applications like GeoServer or MapGuide.

No matter whether you’re building on ESRI or open source, if you are building something complex your staff will have to learn a few new skills. Their prior experience with core IT concepts like programming and systems administration will serve them well in both

domains, and the learning curve will be no worse either way.

“What is the business model behind open source development? If people are making money from open source, I feel safer that it will be around for years to come.”

The people improving open source fall into lots of categories, but here are some broad ones:

- **The altruist/tinkerer.** The most popular media archetype, the altruist/tinkerer probably accounts for the smallest amount of effort on mature open source technologies, but occasionally will create a new technology from scratch that is so good and compelling it gathers other types who then keep it alive and move it into wider utility.
- **The service provider.** The most widely understood business model, running from the one-man consultant to billion-dollar companies like Red Hat. They will sell you support, or custom development, directly on a particular piece of software. Because they are tightly associated with the software, this is the easiest model for “vendor minded” folks to mentally grasp.
- **The systems integrator.** Working “with” – but not necessarily “on” – the software, the system integrator is easily drawn into fixing bugs and adding new functionality on projects as the client requires. Systems integrators love open source because it allows them to meet client needs without being stuck behind a vendor’s development priorities (“we’ve got your bug report on file for the next release” and “that feature will be available in 18 months”).
- **The company man.** Easily the least appreciated member of the open source pantheon, because he is not paid to work on open source. He is paid to work on fish inventories. Or carbon models. Or inventory management. Or tax collection. But he has a bit of discretionary time, and he uses it to make the tools he uses work better.

Notably missing from this list is “the billionaire” and “the venture capitalist”. That is because, while there is money to be made in open source, there are not buckets of money to be made. The companies are smaller, and growing organically; they are not loud venture-backed marketing machines.

“Are decision makers going to be comfortable with this? How can I sell this concept to them?”

It is probable that decision makers will not initially be comfortable with open source, because it represents a change in their way of doing business. More trust must be placed in internal staff to locate and evaluate candidate technology, and less in external vendors who come in and sell solutions directly to management.

For an organization just getting started with open source, it provides advantages at the margins: not in reworking your existing systems, but in giving you flexible options when building new ones.

Existing systems should be left running until they hit a natural end-of-life, either when they become out of date, or so expensive to run and pay maintenance on that the switching cost actually becomes acceptable.

Evaluate the cost of change regularly. Sometimes not changing is the more expensive option, and it is important to know when that time arrives.

“Is that it?”

Hardly. Books [4][5] have been filled on the economics [6] and sociology [7][8] of open source software and communities. Keep an open mind, engage with the communities behind the software, and open source will rarely disappoint.

Links

- [1] http://en.wikipedia.org/wiki/Google_platform
- [2] http://www.redhat.com/about/presscenter/2002/press_suisse.html
- [3] http://www.oreillynet.com/pub/a/oreilly/opensource/news/myths_1199.html
- [4] The Practical Manager’s Guide to Open Source, Maria Winslow, <http://www.lulu.com/practicalguide>
- [5] Succeeding with Open Source, Bernard Golden, <http://safari.oreilly.com/0321268539>
- [6] The Magic Cauldron, Eric Raymond, <http://www.catb.org/~esr/writings/magic-cauldron/>
- [7] Producing Open Source Software, Karl Fogel, <http://producingoss.com/>
- [8] The Cathedral and the Bazaar, Eric Raymond, <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/>

Author Bio

Paul Ramsey started working with open source geospatial software as a practical matter – in 2001, a client needed a spatial database for a project, and the proprietary alternatives of the time were expensive and balky. The solution he and his company developed, PostGIS, was good enough to be useful in other projects for other clients, and was soon adopted as the standard spatial database of the open source community. Today, PostGIS is used globally by organizations differing in scale from graduate research projects and community mapping to DigitalGlobe and NASA. Mr. Ramsey is a Director of the Open Source Geospatial Foundation (OSGeo), a committer in several open source projects and a popular speaker and teacher on open source topics at geospatial conferences and workshops.