Towards a Free and Open Source (FOSS) Spatial Data Infrastructure

Chris HOLMES and Allan DOYLE, USA, and Mick WILSON, Kenya

Key words: Open Source, Spatial Data Infrastructure, Developing Nations

SUMMARY

This paper presents the progress made by several grassroots communities of Open Source developers and users that forms pieces of usable and affordable decentralized Spatial Data Infrastructures. The paper shows that such freely available and modifiable software can play a key role in making a truly Global Spatial Data Infrastructure. Money in developing countries can go exclusively towards developing skills and local capacity, instead of paying license fees that tie customers to a single vendor.

Drawing upon the models that have made Linux and Apache such successes in recent years the rapidly emerging Open Source GIS community has already made huge strides towards software that can truly compete with proprietary offerings. We show that a full stack of tools is being built, interoperably and in a number of programming languages, so that new projects can quickly stand on the shoulders of those that have come before. This will ensure that the acceleration in development that the past few years have seen will only increase. Many of the most important technical pieces needed to implement a Spatial Data Infrastructure indeed are already stable, moving towards even more maturity.

The Open Source movement additionally represents a new model for financing and implementing Spatial Data Infrastructures by taking a truly collaborative and iterative approach. Funding for projects comes from a variety of sources, for a number of different ends. We present a case study of the GeoServer project, to demonstrate how disparate actors deciding to work together can create high quality and usable software. We also show how this collaborative and iterative approach can potentially open the doors for new economic opportunities in the countries that employ them. Money spent on tailoring and translating the software can go towards developing local software industry, since no large company in some other country owns the code. This also allows more culturally sensitive solutions, as they are developed by people in the country, attuned to their real needs, instead of a one size fits all answer imposed from the outside. We believe this Open Source GIS movement is of utmost significance in developing affordable, high quality, usable software for building SDI's. We conclude that the Open Source GIS community is much more mature than one might initially think, and that it is well positioned to contribute to a truly Global Spatial Data Infrastructure.

Towards a Free and Open Source (FOSS) Spatial Data Infrastructure

Chris HOLMES and Allan DOYLE, USA, and Mick WILSON, Kenya

The term *Open Source (OS)* refers to a set of licenses that require unfettered access to the human-readable *source code* from which all computer programs are made. There are many published explanations of Open Source methods[1], this paper will highlight the collaborative spirit of the Open Source process. A helpful metaphor is to think of source code as a number of LEGO®[2] pieces. This is not far from reality, as source code is not just a long stream of text, instead it is a number of small files that work together and build something more complex than the sum of their individual parts. One can imagine software as a very complex LEGO house, a functional unit built up from individual pieces that is used by consumers. Most commercial software is sold already built into its final form – similar to an already built house, (or a car, a firetruck, or a school). Unlike LEGO sets, which include a detailed instruction booklet, proprietary software does not include instructions for its inner workings. This is satisfactory for most people, since they just want a house for their LEGO people to live in. But it's antithetical for anyone who plays with LEGOs, or who might want to modify their house after they buy it, or use the parts to build an entirely new type of structure.

Open Source software requires that the instructions to build the house are always included. This may seem like a very minor point, but for people who build with LEGOs it is quite important. A key property of all software is that it can be infinitely copied. In economics terms this is called a *non-rival* good – one which can be used by two people with no loss to one another. This makes software a good type of thing to share, since no one loses anything when someone else is using it.

For a long time Open Source software was something that no one except the LEGO builders cared about. They formed communities and built all sorts of LEGO structures, freely sharing them with one another. They built a culture around sharing, and formed governing structures to organize themselves to build even larger houses, which anyone could then freely copy and use. If someone built a nice extra room on their house, then they could contribute it to the large house that everyone was working on, and all would then gain the benefits of his innovation. As more people started copying and building better houses, the builders came up with even more innovative ways to organize and coordinate everyone's efforts.

Soon the world started noticing that they were building some really nice houses. Not always the best looking houses, but generally some of the strongest. The Open Source community stipulated that these houses should be freely copyable, modifiable, and able to be used as pieces of even larger buildings[3]. Commercial software, on the other hand, continues to not

[1] See Cathedral and the Bazaar [Raymond, 1997], The Political Economy of Open Source Software [Weber, 2000], The Free Software Definition [Stallman, 2004], and Open Source Baselines: Compared to What? [Lessig, 2003]

[2] LEGO® blocks are a child's toy, a set of connecting blocks which can be combined into larger and larger structures. See http://www.LEGO.com/. LEGO® is a registered trademark of the LEGO Company.

[3] This metaphor slightly misses the extensible nature of source code. A more precise description would be that you could walk in to someone's house, notice that they have a very nice light fixture, and simply remove it from their wall and integrate it into your own house (without, of course, them actually losing the fixture).

only hide the instructions, but also is usually bound by a license to prevent people from copying the LEGO house they bought. Open Source communities, however, establish the copying of software as a right, leading to a conception of property "configured fundamentally around the right to distribute, not to exclude"[4].

Open Source GIS is not yet as visible as Open Source superstar projects such as Linux and Apache, but there is a sizable community that has been working away, building quite solid software. Recently, the OS GIS community has been growing incredibly fast, which is due in no small part to the great work of the Open Geospatial Consortium (OGC). In the metaphor of the LEGO blocks, the OGC has essentially provided a number of useful blueprints for many Open Source LEGO builders. Programming GIS is very difficult, requiring not only a full understanding of computer programming, but also a wealth of field specific knowledge about geospatial information. The OGC does not create Open Source software but rather specifies open standards, which is like giving someone a picture of a large LEGO house, and some basic rules about how the house can integrate with other houses. Anyone can then choose to implement the standards, to build a house like the one they specify. And to build it they can choose to buy a private company's LEGO blocks, to completely build their own blocks, or to use and build upon Open Source blocks. It is to everyone's benefit to implement OGC standards, even in commercial software, because if everyone follows the basic rules then the software will be able to *interoperate*, that is, it will be possible to easily combine their various large scale pieces together into even bigger structures.

This makes house building easier, since the smartest people have already gathered together and written up the building codes. A complete overview of the full OS GIS community is beyond the scope of this paper. We strongly encourage readers to follow Paul Ramsey's *The State of Open Source GIS* paper[5], updated periodically with a great overview of the various communities that make up the wider OS GIS community. Ramsey recognizes that "successful OSS [Open Source Software] projects are not created by releasing free source code – they are created through the growth of communities of shared interest."[6] The paper examines all the major OS GIS projects and many of the smaller ones, and puts them in a community context. He examines the basic building blocks that are shared between projects, and the users and developers who can tutor new users. The focus on sharing and community is the true strength of the Open Source development process, and Ramsey utilizes this focus to give a great overview of the state of OS GIS.

For those evaluating an Open Source alternative, while the free licenses of Open Source are an obvious appeal, the benefits go far beyond being able to download software at no cost. Steven Weber, in his *Open Source Software in Developing Economies* observes:

> OSFS [Open Source/Free Software] advocates have observed the negative balance of trade impact of a country's reliance on a few major software suppliers located in other countries. A South African government council responsible for formulating the

Software is simply made up of progressively larger and larger pieces, and if the source code is open then *any* of those pieces can be reused by other projects.

[4]Castells, 2005
[5]Ramsey, 2005
[6]Ramsey, 2005

> government's open source policy expressed foreign currency savings as an explicit rationale for considering OSFS deployment. The Taiwanese government estimates that it could save nearly $300 million in royalty payments through a strategic open source project that encourages research and development in office software and the opening of source code for government agencies and private enterprises as well.[7]

There is still much debate about the Total Cost of Ownership (TCO) for Open Source versus commercial software[8]. But a key benefit for developing countries is that they do not remain completely dependent on single software provider. After a commercial software license is bought, only the owner of the source code is able to be contracted for support and maintenance. With Open Source licensing the code is available to all, so a number of companies can all compete to provide low cost and quality support, installation, training and maintenance.

This can not only save money on upkeep, but also gives governments or NGOs more control over where their money goes. They can avoid reliance on a single supplier who is then able to fix their terms in a non-competitive environment. There is also the potential to keep money inside the country by starting and supporting a local software industry. Expenditures can then stay at home, instead of constantly being sent to software providers overseas.

Unfettered access to source code and to the communities surrounding Open Source projects can greatly help to develop the talents of domestic programmers. An investigation by the Swedish International Development Cooperation Agency (SIDA) entitled *Open Source in Developing Countries* explains that "one of the biggest advantages of open source is that it gives the opportunity for arbitrary localization and customization."[9] Countries can establish a program to localize software to the native language. This can not happen with proprietary software, because the source code must be accessed to perform localization tasks. Many languages simply do not get translated in commercial software since there is not a big enough market to justify the costs associated with localization. A group in Tanzania recently translated OpenOffice, a complete office suite that is similar to Microsoft Office, into Swahili. The technical leader of the project, Alberto Escudero-Pascual, said after completion of the first version, "we clearly show that with free and open source software we can do in four months what proprietary software has never done for the Swahili speakers."[10]

This process of localization can have benefits beyond just the concrete results of a program in a country's native language. Since the code is accessed during the process, the work done to localize it can function as a apprenticeship for the native speaker. This person becomes acquainted with the community and gains a deeper understanding of the programming techniques of key Open Source projects. This experience gives participants the ability to make deeper contributions to the project, and before long, they can become core members of the Open Source community with marketable programming talents. A country can seed a

[7]Weber, 2005

[8]Many of the TCO debates miss out on a key point - the market power of consumers. As consumers start demanding OS solutions the market will supply them, at increased efficiency and less cost in the most efficient manner, as is the nature of our market economy.

[9]Weerawarana and Weeratunga, 2004

[10]Jambo Open Office website - http://www.o.ne.tz/pressrelease_en.php

local software industry by starting with localization tasks. Developing countries would benefit immensely by having their own software industry, both by keeping government and NGO funds in their own economy, and by giving the country more input into the solutions developed for its citizens.

The process of building a local software industry is also encouraged simply by providing Open Source products to the citizens of the country. The free diffusion of software tools leads to an environment where "the degree to which a software tool can be utilized and expanded becomes limited only by the knowledge, learning, and innovative energy of the potential users; not by exclusionary property rights, prices, or the power of countries and corporations."[11] As common citizens use open software and demand new tools suited to their needs, a local software industry can grow provide them. This in turn leads to far more than just a local software industry – it leads to a culture of innovation. A culture which does not merely take hand-me-down technological solutions from the first world, but one that allows developing countries to chart their own course. In Kenya there is an informal business sector known as *jua kali* – in short, give a *jua kali* worker a flattened fruit tin and a piece of pipe to use as hammer and he will fashion you a new car, a hurricane lamp, a suitcase or whatever you require. He is amazingly innovative, and gears his solutions towards the situation at hand. Open Source software opens the door for a *jua kali* sector in software – a local business that can take pieces from other OS projects, and fashion exactly what is wanted. With Open Source methods the great innovations that the *jua kali* performs not only perfectly gear software to the local needs, but can also become a source of value in the world market.

The Open Source model has a further benefit that is especially relevant for financing and implementing the software components of a Spatial Data Infrastructure. The Open Source process is already built around communities of collaboration and governance structures that enable sharing, which means a shared funding model is already in place. In an Open Source project the pieces are built iteratively, by those who have the time and money to lend support. For an individual country building an SDI this means it can leverage the work that has gone before. And since the licenses are free, it means that one can try out a solution before having to buy it, or can pick and choose which pieces make sense to use.

But the power of Open Source is not just using existing software, it is that multiple countries can easily collaborate on the software pieces needed to implement their Spatial Data Infrastructures. The great thing about the OS model is that countries need not explicitly coordinate funding through writing memoranda of understanding and complex contracts about who pays for what and who gets what. Instead initiatives need only to leverage and to give back to the same communities of Open Source software. These communities are usually the most adept and experienced in coordinating the various requirements and work to be done. Incremental OS GIS, developed with a local focus and demonstrating success at each step, not only progresses towards getting the job done but *also* establishes political and institutional rationales for more formal funded collaborations.

Most major OS GIS projects get contributions from a variety of sources, and fluidly coordinate the efforts. An example is useful to illustrate this process. One of the authors is the lead developer of The GeoServer Project, a fairly successful OS GIS endeavor. A small non-

[11]Weber, 2005

profit called The Open Planning Project (TOPP), started GeoServer in 2001 out of a desire to promote OGC-based standards to build up the Spatial Data Infrastructure of the United States. TOPP's interest was in being able to access traffic data for New York City, but the organization felt that an Open Source implementation of the Web Feature Service (WFS) specification would have value for others as well. One of the first sources of funding was the Cooperative Agreements Program (CAP) of the US's Federal Geographic Data Committee (FGDC). This grant helped to build the basics and to add a protocol that advertised the data on the FGDC's clearinghouse. Following this, the OGC selected GeoServer as the WFS reference implementation, providing funding for development that would ensure compliance with all parts of the WFS specification. This took GeoServer to 1.0. The next large phase of development was implementing the Web Map Service (WMS) specification. Some loosely affiliated Open Source developers had previously built a WMS based on a common toolkit called GeoTools, which was the same building blocks that GeoServer was built upon, for an academic mapping project in Britain. The design to glue the two projects together was done by an Argentinian programmer working for a Spanish commercial company on a project for the Basque government. He decided that GeoServer's functionality was close enough to his needs that it made more sense for him to spend the time improving it than to buy a commercial solution.

GeoServer 1.2 was primarily the work of yet another company, Refractions Research of British Columbia. Refractions applied for matching funding to collaborate with TOPP on GeoServer, in order to build an innovative validation engine and to make the server far easier to use with a web-based administration tool. This funding came from GeoConnections[12], an amazing program in Canada whose mission is to help build their SDI. In 2004, GeoServer saw new contributions from an Australian company called Social Change Online, to execute cutting edge research for GeoScience Australia (the national agency for "geoscience research and geospatial information."[13]) The latest funding is coming from British Columbia's Ministry of Sustainable Resource Management (MSRM), to make improvements to GeoServer's ArcSDE module, as that is the additional piece that the agency needed.

A key thing to note is that none of these funding sources supplied over $60,000 US, which is enough to purchase between 0 and 6 single licenses of some leading commercial web GIS servers. No single organization needs to take a major risk of funding an Open Source project in isolation. By finding a project that is close to meeting their needs, they can simply fund the additional pieces that they require. This will in turn get the project closer to the needs of the next potential funder. The great part is that after a few years, an earlier funder can continue to reap the benefits of the seeds they sowed.

By strategic funding of Open Source GIS, governments can obtain not only great software for themselves, but they can also freely share it with the rest of the world. Many times very large contracts will be used by commercial software firms to fund further development, perfecting their own LEGO blocks, which they can then sell at even higher prices, since they are the only ones to have them. Instead, in the Open Source model, a government can fund builders who will share their results freely. The LEGO blocks themselves become free, and the only development that this and other governments have to pay for is the new, specialized blocks

[12]http://www.geoconnections.org/CGDI.cfm
[13]http://www.ga.gov.au/

that no one else has yet built. As Castells says, "Indeed, the Open Source model undercuts the conventional business logic. Power in this market shifts away from software suppliers and toward software customers." He is quick to point out that it does not eliminate business, there is still a market for training, support, upkeep, and leading new innovations. This puts the power back in the hands of software customers, by giving a choice based on who will meet their needs best, not who has the largest collection of private LEGO blocks.

Although OS GIS still lives in relatively obscurity compared to the major GIS vendors, it's growth is truly snowballing, since more and more can be built once the basic building blocks are established. Michael Gould recounts how in 1987 his company contemplated building their own GIS, and were told it just was not possible, since they would have to build every basic piece themselves. 17 years later he was advising a regional government to make GIS software tailored to their needs by hiring a small IT company. But this time around, "the IT company was able to produce a viable working prototype in under four calendar months (something in the order of 20 person-months)." He emphasizes that "the key to making this possible is free and Open Source software (FOSS). The GIS-related FOSS community has grown in leaps and bounds over the past two years, creating a substantial collection of ready-to-use components for programmers."[14] He too points to the future, that "in 2005 more and better examples of solid, reliable, open GIS components will be revealed."[15] Since OS GIS has no marketing budget, it may be a while until the news breaks into the mainstream press. But soon, for many GIS-related tasks, the freely available components will be just as good if not better than their commercial counterparts. For example, if one needs a spatial database there is PostGIS and MySQL. For a OGC web services there is MapServer, GeoServer, and Deegree. A catalog server and online metadata editor is available from GeoNetwork. There are web-based WMS clients like Intermap, Mapbuilder, Chameleon, and more. Powerful desktop GIS clients are emerging with uDIG and gvSIG, adding to offerings by Thuban, JUMP and TerraView. And for developers there is JTS for topology, Batik for SVG, and generic toolkits like GeoTools, GDAL/OGR, and Terralib for more general GIS programming. A complete list of Open Source GIS projects can be found at freegis.org[16].

Perhaps the most important thing to take away from this paper is that Open Source GIS components are available for all to use, and you too can help the effort. Those who can influence funding and procurement decisions are greatly encouraged to consider OS GIS projects. It can be a bit daunting to find and determine which project to use. Indeed, many times the most useful contribution can be funding a comparison of different commercial and Open Source offerings, so that others can then access the information, and be better informed as to what solutions exist. The British Columbia MSRM did exactly that, which led them to fund improvements on GeoServer so it could meet their needs while also providing the community with their findings. There are also a number of companies and non-profits[17] that are available to assist with both advice on which Open Source projects to use, and on how funds can make the biggest impact on the OS GIS world. And most of these companies will

[14]Gould, 2004
[15]Gould, 2004
[16]http://freegis.org
[17]EOGEO and The Open Planning Project in the United States, Refractions Research, Vivid, and DM Solutions in Canada, Axios Engineering and IVER in Spain, CCGIS and Deegree in Germany, and Social Change Online in Australia to name some of the most prominant ones.

happily provide commercial-quality installation, support, and training on a variety of Open Source GIS products. The more OS GIS is demanded, the more companies will spring up to meet what the market requires.

Open Source GIS software is obviously not a cure all solution to implementing a Global Spatial Data Infrastructure – no amount of technology alone thrown at a problem will solve it, and SDI's are no different. The important thing behind Open Source is not the products, but the inclusive process used to develop and build upon those results. If the tools that implement the pieces needed for the Global Spatial Data Infrastructure are built on a spirit of cooperation and collaboration, then we are optimistic that such a spirit will carry over to the whole process of sharing geographic data. We hope empower participants in an SDI as builders – just because people write software should not mean they are the only ones who get to play with LEGOs. In this information age it becomes obvious that geographic data, models, and analyses have the exact same properties of our source code LEGO blocks – they are pieces of information that lose nothing by being copied, and can gain infinitely by being shared openly. It is only with the spirit of collaboration that a truly Global Spatial Data Infrastructure can emerge. The OS GIS community has started to build the links from software to SDI's, and we hope that others will start to utilize the tools and run with it.

REFERENCES

Castells, M. 2005 Innovation, Information Technology and the Culture of Freedom, at World Social Forum in Porto Alegre, http://openflows.org/article.pl?sid=05/01/31/2028221

Gould, M. 2005 'Letter to the Editor' in *Geoinformatic*, at http://www.geoinformatics.com/paginas/letterstoeditor.html?subject=Letters%20to%20the%20Editor#Anchor-11-49575

Lessig, L., 2003 Open Source Baselines: Compared to What? In *Government Policy toward Open Source Software*, edited by Robert W. Hahn (Washington, DC: AEI-Brookings Joint Center), http://www.aei.brookings.org/publications/abstract.php?pid=296.

Ramsey, P., 2005 *The State of Open Source GIS*, http://www.refractions.net/whitepapers/2005-02-OSS-Briefing.pdf

Raymond, E.S., 2000 *The Cathedral and the Bazaar*, http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/

Stallman, R., 2001 "The Free Software Definition," at http://www.fsf.org/philosophy/free-sw.html

Weber, S., 2000 *The Political Economy of Open Source Software*, at http://e-conomy.berkeley.edu/publications/wp/wp140.pdf

Weber, S., 2004 *The Success of Open Source* (Cambridge, MA Harvard University Press).

Weber, S., 2005 *Open Source Software in Developing Economies*, at http://www.ssrc.org/programs/itic/publications/ITST_materials/webernote2.pdf

Weerawarana, S. and Weeratunga, J., 2004 *Open Source in Developing Countries* (Sweden, SIDA), at http://www.sida.se/Sida/jsp/polopoly.jsp?d=1265&a=23955

CONTACTS

Mr. Chris Holmes
2005 US Fulbright Scholar to Zambia
c/o EOGEO
36 Norwich Road Needham, MA 02492
USA
cholmes@openplans.org

Mr. Allan Doyle
EOGEO
36 Norwich Road Needham. MA 02492
USA
adoyle@eogeo.org

Mick Wilson
UNEP DEWA
PO Box 30552 Nairobi
KENYA
Mick.Wilson@unep.org