# OSBR.CA

The Open Source Business Resource




DECEMBER 2010


# DECEMBER 2010

**PUBLISHER**

The Open Source Business Resource is a monthly publication of the Talent First Network. Archives are available at: http://www.osbr.ca

**EDITOR**

Chris McPhee
chris.mcphee@osbr.ca

**ISSN**

1913-6102

# Editorial

## Chris McPhee and Leslie Hawthorn

**From the Editor-in-Chief**

***The editorial theme for this*** issue of the OSBR is Humanitarian Open Source.

***In this issue, authors in*** Canada (Ottawa and Toronto), Sri Lanka (Columbo), and the United States (Brunswick, Hartford, Indianapolis, New York, Portland, and Seattle) draw upon their experiences to show the role of the open source approach in meeting humanitarian needs in the past, present, and future.

***We encourage readers to share*** articles of interest with their colleagues, and to provide their comments either online or directly to the authors.

***The editorial theme for the*** upcoming January 2011 issue of the OSBR is The Business of Open Source and the guest editor will be Michael Weiss, Associate Professor in the Technology Innovation Management program at Carleton University. For upcoming issues, we welcome general submissions on the topic of open source business or the growth of early-stage technology companies. Please contact me if you are interested in submitting an article (chris.mcphee@osbr.ca).

**Chris McPhee**

**Editor-in-Chief**

*Chris McPhee is in the Technology Innovation Management program at Carleton University in Ottawa. Chris received his BScH and MSc degrees in Biology from Queen's University in Kingston, following which he worked in a variety of management, design, and content development roles on science education software projects in Canada and Scotland.*

**From the Guest Editor**

***In recent years, our increasingly*** connected world has provided us with a greater understanding of the needs of our fellow global citizens. The devastating worldwide impact of natural disasters, disease, and poverty has been raised in our collective awareness and our ability to collectively alleviate this suffering has been brought to the fore. While many of us are familiar with donating our funds to better the lives of those less fortunate than ourselves, it is often easy to overlook a core component of facing these global challenges: information and communications technology (ICT).

***The humanitarian open source movement*** seeks to ameliorate these sufferings through the creation of ICT infrastructure to support a wide array of goals for the public good, such as providing effective healthcare or microloans to the poorest of the poor. Achieving these goals requires a sophisticated set of software and hardware tools, all of which work to save and improve lives in some of the most difficult of situations where the availability of electricity, data, ICT knowledge, etc. may be low or lacking altogether. It should come as no surprise that the humanitarian open source domain attracts a great deal of attention from software developers, engineers, and others who find that they are able to both solve intense technical challenges while helping to improve the lives of others.

***However, to support ongoing humanitarian*** needs, the communities who produce humanitarian free and open source software (HFOSS) and hardware have increasingly identified the need for business models to support their efforts. While the lower cost of using open source software and hardware solutions means that more funds can be directed to aid and comfort

those in need, the goodwill of developer communities and the funds of grantees alone cannot grow the ecosystem sufficiently to meet ever-growing global needs. To face these challenges – poverty, global health crises, disaster relief, etc. – humanitarian open source projects must fully engage the market and provide cost-effective, efficient solutions to the technical aspects of these challenges.

***In this issue of the*** OSBR, our authors from several open source software and hardware projects explore not only the global need for humanitarian open source projects, but also the business cases for humanitarian-focused ICT.

***Chamindra de Silva, Director and*** CTO of the Sahana Foundation, explores the HFOSS landscape as the incarnation of the concept of "software engineers without borders." His article discusses the synchronicities between humanitarian open source software efforts and the open source software model, while highlighting the specialized requirements of HFOSS development.

***Mark Prutsalis, President and CEO*** of the Sahana Foundation, examines the Sahana project's service to the global community in disaster-relief scenarios and the need to build a service industry based on supporting HFOSS in order to sustain the ecosystem.

***Glenn McKnight and Alfredo Herrera*** from IEEE Canada's Humanitarian Initiatives Committee describe the IEEE's Humanitarian Technology Challenge. This challenge resulted in five open hardware solutions to provide reliable sources of electricity, and the authors explore the creation of these new technologies as part of the broader need to create appropriate and sustainable solutions for the issues facing the world's most vulnerable people.

***Adam Feuer, Director of Engineering*** for the Grameen Foundation's Mifos Initiative, discusses the effort to alleviate global poverty through microfinance. Feuer elaborates on how the development of the Mifos software package and the nurturing of its community can be used as a model to address other global challenges.

***Dawn Smith, Project Coordinator for*** Open MRS, examines the role of the OpenMRS medical record system in anchoring a health information business ecosystem. Smith details the project's success in encouraging the formation of new business opportunities through collaboration and the creation of local capacity for development, maintenance, and service provision.

***Ralph Morelli, Professor of Computer*** Science at Trinity College, Allen Tucker, Professor Emeritus at Bowdoin College, and Trishan de Lanerolle, Project Director for the Humanitarian FOSS Project at Trinity College, discuss the academic Humanitarian Free and Open Source Software Project. In addition to exploring the benefits of using HFOSS development to improve undergraduate computer science education, their article discusses their student's forays into commercializing one of the applications developed under the auspices of the HFOSS program at Trinity College.

***Mike Herrick, Executive Director of*** the Collaborative Software Foundation, details TriSano, a software package designed for surveillance of infectious disease transmission, bioterrorist attacks, and other threats to public health. Herrick's exploration of collaborative processes serves as a model for others who are developing first-responder HFOSS technologies while simultaneously providing insights into building an effective business model to best serve the technical and budgetary needs of clients.

**Leslie Hawthorn**

**Guest Editor**

# Editorial

*Leslie Hawthorn*

*Leslie Hawthorn is the Open Source Outreach Manager for Oregon State University's Open Source Lab (http://osuosl.org). Previously, she was Program Manager for Google's Open Source Programs Office, where she was the Community Manager for the Google Summer of Code community. She conceived, launched, and managed the Google Highly Open Participation Contest (now Google Code-in), the world's first global initiative to get pre-university students involved in all aspects of Open Source software development. Leslie has also organized more than 100 open source conferences and hackathons, most held at Google's Corporate Headquarters in Mountain View, California, USA. When not wrangling FOSS developers, she's usually speaking about Open Source, FOSS in education, and community building. Leslie holds a Honors B.A. in English Language and Literature from U.C. Berkeley. Her personal website is http://hawthornlandings.org.*

# Humanitarian Free and Open Source Software

## Chamindra de Silva

*"What this world needs is a new kind of army – the army of the kind."*

Cleveland Amory

Humanitarian free and open source software (HFOSS) represents the application of free and open source software (FOSS) to the coordination problems faced in the humanitarian and disaster-response domains. FOSS has found a natural home serving the humanitarian domain because of certain problem patterns that promote the use of an open source approach. HFOSS also integrates two volunteer-rich communities that have much in common: the humanitarian community and the open source community. HFOSS is not distinct from the free and open source approach, but is rather a specialization of its principles. This article explores and elaborates on that natural alignment by presenting the concepts of HFOSS and the ecosystem that sustains it.

**The Relevance of IT for Humanitarian Response**

The humanitarian response domain aims to help save lives and alleviate human suffering in responding to: i) rapid-onset natural disasters, such as tsunamis, earthquakes, and pandemics; ii) slow-onset natural disasters, such as global warming, droughts, and famine; and iii) human-instigated disasters, such as civil wars. The humanitarian response domain consists of an ecosystem that is strongly represented by non-governmental organizations (NGOs), civil society organizations, and state-based institutions. The larger organizations in this ecosystem are the non-profit agencies such as Save the Children, the Red Cross, World Vision, Oxfam, and organizations of the United Nations such as the World Food Programme (WFP), the Office for the Coordination of Humanitarian Affairs (OCHA), the World Health Organization (WHO), and the United Nations Children's Fund (UNICEF). There are also tens of thousands of smaller NGOs serving the domain in different regions and areas of specialization. Most of these institutions survive on donations from society, corporate sources, and various states, and they are governed as non-profit bodies. The domain is also complemented by a great deal of voluntarism that peaks during disaster events. Organizations like Doctors Without Borders/Médecins Sans Frontières or Engineers Without Borders are volunteer professional groups that provide capacity to humanitarian response events.

As much as food, shelter, medical aid, and security are important for the affected victims in a disaster, so too is the information needed to identify those needs in the first place and create the essential connections for recovery. Correct and timely information is critical to effective response, especially during the first three days following an event, when there is an opportunity to save lives with timely action. However, as is often the case during a disaster, the scale of the event overwhelms the effectiveness of information gathering and dissemination using traditional approaches. For example, in the 2004 Asian Tsunami, governments found themselves, all of a sudden, dealing with responding to the needs of millions of people. Managing informa-

tion at this scale is a problem for which information technology was built. Typical large-scale data management problems include finding missing persons, tracking displaced people, tracking medical facilities, situation mapping, effective distribution of aid and resources, reporting hazards or violence, and many more.

**Open Source Communities as Software Engineers Without Borders**

In the humanitarian response domain, the free and open source approach has become prevalent for distributing software solutions as public goods. Popular FOSS projects and organizations delivering open source public goods specifically for this domain include:

- Sahana (http://sahanafoundation.org)
- Ushahidi (http://ushahidi.com)
- OpenStreetMap (http://openstreetmap.org)
- InSTEDD (http://instedd.org)
- FrontlineSMS (http://frontlinesms.com)
- OpenMRS (http://openmrs.org)
- RapidSMS (http://rapidsms.org)
- Google (http://google.com)

And there are many more institutions contributing to these projects. These FOSS communities and the global communities that work around them effectively provide "software engineers without borders." Most of these products depend on open source platforms built from open source components such as Apache, Linux, PHP, Python, MySQL, and these projects in turn have become inadvertent donors towards the goal of delivering these essential software public goods.

**Open Source Alignment to Humanitarian Values**

Why does open source find such a natural home here? There are multiple reasons particular to the humanitarian response domain where open source software freedoms and practices align to the principles in the humanitarian domain. Over the years, these principles have been captured in codes such as the Red Cross Code of Conduct (http://ifrc.org/publicat/conduct/). The principles that align to FOSS in particular include:

**1. No discrimination on access:** From the time an open source package is available online, it becomes a global public good that anyone, irrespective of race, station, or creed, can download and use, customize, and apply to serve their respective relief effort.

**2. Ability to leave technology behind:** Irrespective of the support group that brought the technology to help the response effort, open source is a product that can be left behind and maintained by local groups if required.

**3. Empowering local capacity:** Most disaster response activities work around the local capacity on the ground. The local community is the most tuned-in to the priorities of disaster response and thus they are the best suited group to modify the software for the response effort. Local open source community groups, such as local Linux User Groups, are recommended as a support mechanism.

**4. Lower cost:** Open source reduces the cost of software for disaster response, which means more funds are available for essential aid.

**5. Transparency and neutrality:** The software design and mechanism for building FOSS is transparent and neutral. These characteristics are particularly important for countries that mis-

trust the origins and affiliations of technology. With FOSS communities that are mature, global, and diverse, the software has no specific alignment to a particular political agenda.

In addition, FOSS also has the following added advantages in humanitarian response:

**6. Rapid localization and adaptability:** No two disasters are alike; often localizations and customizations are needed for the software before it can be applied effectively to a disaster. Furthermore, no two nations handle the humanitarian response in the same manner. Free access to the code and the freedom to modify it are essential benefits provided by FOSS.

**7. Open standards and data exchange:** With many different systems in operation during a disaster response, it can be a significant issue if they do not share data. To avoid confusion and inefficiencies, it is essential for systems to be able to share information through open standards. Open source has a track record with open standards from TCP/IP and it helps promote the spread and implementation of open standards.

**8. Shared inter-organizational development:** NGOs and humanitarian relief groups all need software tools to be effective and competition is not as rife as in other industries. Sharing the development of their IT infrastructure is a natural way to reduce costs and promote integrated response efforts.

**Does HFOSS Differ from FOSS?**

Rather than differing from FOSS, HFOSS recommends certain best and standard practices due to the critical nature of the humanitarian response environment and the impact solutions can have. These best practices include:

**1. HFOSS software is mission critical:** The HFOSS products that are delivered can have a significant impact on saving lives and alleviating suffering, thus the systems are mission critical and there can be no compromises on quality and stability. Information loss or corruption due to the system is completely unacceptable.

**2. High usability and low learning curve are essential:** The entire disaster response community is diverse and ranges from trained emergency responders to untrained volunteers. Most of these users have not seen or used the HFOSS system before; they should be able to understand how to use it in as little time as possible.

**3. No administrators and superusers are expected:** Chances are there will not be strong linux-savvy administrators around to help set up the software. Even if there were, they probably will not have the time. A normal IT-literate user should be able to install the software and a version that runs on Windows is essential.

**4. Data sharing through import and export:** Data will arrive and be needed in many formats, thus import and export mechanisms, especially using open standards, are important to make the software effective.

**5. High resilience and avoidance of single points of failure:** Humanitarian response solutions have to be fault tolerant to issues like a lack of Internet access, throughput, and even power. Additionally, the community should be diverse enough that there are multiple people to support a particular solution so that no one becomes a bottleneck.

**The HFOSS Consortium**

HFOSS, or the approach of delivering open source products in the humanitarian response domain, is a model that is now widely accepted by many of the key institutions. However, there are often shortcomings, mostly due to the infancy of these volunteer developer communities, that need to be addressed before the model becomes widely adopted. Some of these concerns

can be resolved in partnership. A good example was between InSTEDD, Ushahidi, Sahana, Crowdflower (http://crowdflower.com), Google, and the Crisis Mappers Network (http://crisismapping.ning.com) in the establishment of the "4636" SMS short code as a free aid service for response during the Haiti Crisis Response. Enabling this partnership, however, required system-to-system data exchange between systems, which required some last-moment hacks and rapid enhancements. A reflection of this partnership following the Haiti response initiated an informal grouping of HFOSS projects that includes InSTEDD, OpenStreetMap, Ushahidi, Sahana, One Laptop per Child (OLPC; http://laptop.org), and Google. This group was established to work towards better integration between systems and be better prepared for collaborative response. To help encourage more collaboration, an HFOSS code of conduct (http://humanitarian-ict.org/wiki/h-foss_code_of_conduct) capturing development and deployment best practices has also been drafted and accepted. Recognizing all areas for improvement, an HFOSS maturity model is also being developed as a yardstick for growth to deliver to the demands of disaster response with more rigor, efficiency, and sustainability.

**The HFOSS Ecosystem for Sustainability**

Despite the best of intentions, volunteerism will not easily deliver to standards and best practices in a sustainable and consistent way. What we often find is that there is a peak in capacity in HFOSS projects during a disaster response event, but it soon tapers away, leaving solutions that are sometimes not being maintained actively. There is also an imbalance in the different tasks required from development, documentation, quality assurance, and training based on available volunteer capacity. There needs to be an economic model to sustain a core group of people around the deployed solution to cater to its stability, usability, documentation, and training.

However, at the same time, it is rather inappropriate to be seeking business opportunities in the midst of a disaster response, where most private and public institutions are found donating goods and services. This is why most of the organizations that work in humanitarian response are non-profits, NGOs, and government funded groups. Some of the HFOSS projects themselves are funded non-profits, such as InSTEDD and Ushahidi, and they sustain core teams for disaster response. However, even they cannot manage all events and there needs to be a transition to other providers, especially in building up local capacity for support. Ushahidi, for example, transitioned the maintenance of the Haiti Ushahidi deployment to a private company based in Haiti.

Thus, business and social entrepreneurship opportunities lie in pre-disaster and post-disaster events, when the efforts of volunteer and non-profit organizations are insufficient to maintain tailored solutions for either the recovery or preparedness phases of disaster response.

**A Proposed Public-Private Partnership Model for HFOSS Projects**

An effective economic model has still to evolve that sees non-profits, for-profits, corporate social responsibility programs, and volunteers all working towards a common goal in delivering HFOSS solutions. A business model the author would like to propose for this domain is that of an HFOSS project governed by a funded, central non-profit organization with a paid core team to manage contributions and assure the quality of the core product releases. This same team would be funded for disaster response efforts. However, once the initial support is provided, a suitable transition needs to be made, preferably to a local for-profit or non-profit organization that will maintain the solution in the long term. A certification and corporate sponsorship program can in turn help maintain the essential tenets of quality and fund the core non-profit for its maintenance activities.

# Humanitarian Free and Open Source Software

*Chamindra de Silva*

**Conclusion**

Delivering global public goods in the form of free and open source products has become a popular norm in the humanitarian response domain. HFOSS projects and the communities that surround them have become the natural homes for the "software engineers without borders" of the world. However, these projects need further growth to assure the sustainability and quality of products being deployed due to the mission-critical nature of the applications and their requirements for a high degree of quality, especially aligned to stability, fault-tolerance, and usability. One approach to delivering sustainability can be through a healthy economic ecosystem around an HFOSS project that involves funded non-profits, for-profits, corporate social responsibility programs, and volunteers working together to bring efficiencies to the disaster response efforts that will help save lives and alleviate human suffering.

*Chamindra de Silva is Director of the Sahana Foundation and volunteers as the foundation's CTO. He has been involved with Sahana from its inception in Sri Lanka and was the project lead from 2005. He is the concept founder of HFOSS and is also on the advisory board of the HFOSS academic project. His other involvements in open source include being a charter member of the Open Source Initiative (OSI), an Apache Committer, and a contributor to the One Laptop per Child (OLPC) project. His day job is the Head of Strategic Initiatives in the Global Technology Office at Virtusa Corp.*

# Developing a Service Industry to Support the Sahana Disaster Management System

**Mark Prutsalis**

*"No innovation matters more than that which saves lives."*
Avelino J. Cruz, Jr.
Secretary of National Defense of the Philippines,
on the use of Sahana following disastrous mudslides in 2005

The Sahana Software Foundation (http://sahanafoundation.org) governs humanitarian free and open source software (HFOSS) projects that address the information-coordination challenges of disaster management. Sahana software has been proven effective and is recommended by numerous industry sources from both the emergency management and business sectors. It fulfills critical needs to enable organizations responding to disasters to share information across organizational lines and to track and effectively manage disaster efforts. These needs are substantial and are growing because of climate change and urban population growth, which is leaving increasing numbers of people vulnerable and susceptible to the effects of disasters.

Sahana fills a unique niche in the toolkit of emergency and disaster response agencies because it facilitates critical information sharing and coordination of efforts across all types of organizations and individuals involved, and is readily flexible to the needs that arise from any particular disaster. The software has its roots in the open source business community and has always been strongly supported by that industry. In this article, we describe an industry opportunity in HFOSS and provide evidence that Sahana software can be successfully commercialized. These factors make it a ripe time for open source businesses to engage cooperatively with the Sahana Software Foundation to support the growth of a service industry around Sahana software and HFOSS.

## Introduction

Sudden-onset natural disasters have a devastating political, economic, social, and human impact on a society. The 2004 Indian Ocean Earthquake and Tsunami left almost a quarter of a million dead, 500,000 injured, 5 million homeless, 1 million jobless, and caused at least $7.5 billion in damages across several countries (http://tinyurl.com/3agajer). Hurricane Katrina submerged New Orleans the next summer, and dispersed thousands of the evacuees across dozens of states; it was the costliest storm on record in the United States, causing over $81 billion in damage (http://tinyurl.com/2bmdqxp). Approximately 75,000 persons were killed by the 2005 Kashmir earthquake, mostly in Pakistan, with 3 million left homeless (http://tinyurl.com/2cf988d). The 2010 earthquake that decimated Haiti left 230,000 dead, 300,000 injured, 1 million homeless, and almost 300,000 homes and businesses destroyed (http://tinyurl.com/273xxsr).

# A Service Industry to Support Sahana

*Mark Prutsalis*

Monsoon rains left almost one-fifth of Pakistan under water in 2010, affecting 20 million persons left homeless and destroying infrastructure and businesses (http://tinyurl.com/35zhzx9). As the trends of population growth, urbanization, and global climate change converge, the scale and impact of disasters will only continue to grow.

Disasters of this scale require a corresponding massive relief effort to save lives and help disaster victims recover. For agencies responsible for helping disaster victims survive and recover, some of the biggest challenges involve those of effective coordination and management of the multitude of requests for assistance and information. Lifesaving decisions need to be made quickly: where to dispatch search and rescue teams, where to set up shelters, how to effectively distribute aid, how to manage donations, how to trace missing persons, and how to assure security of persons and property in the areas affected by the disaster. The best decisions are the most informed ones. Without access to information needed to make good decisions, relief may not get to those who need it most, aid dollars may be wasted, and lives may be lost.

Ten days after the 2010 Haiti earthquake, those coordinating the relief effort for the US Government did not know the location of all of the operating hospitals and medical facilities in Haiti, what services they offered, their status, and how many beds were available. This information was needed so relief workers would know where they could send the injured or ill, and for those hospitals already overflowing with patients, where they could transfer non-critical patients so they could continue to receive needed medical care. It is hard to imagine how this is possible. The Sahana Software Foundation stepped in to meet this need and through hard work and research by volunteers, we were able to provide through our public portal the first comprehensive data set of all functioning hospitals in Haiti, including their coordinates such that they could be located on a map, within 24 hours, and provided multiple data feeds so everyone could use this information within their own systems (http://tinyurl.com/38smlxh). This information was used by the Pan American Health Organization (PAHO), responsible for coordinating the international health response to the Haiti earthquake, the medical planners for US Southern Command, and dozens of other relief organizations.

**The Sahana Software Solution**

The experience of the 2010 Haiti earthquake illustrates the mission of the Sahana Software Foundation: to help alleviate human suffering by giving emergency managers, disaster response professionals, and communities access to the information that they need to better prepare for, and respond to, disasters through the development and promotion of free and open source software and open standards. In the aftermath of a massive natural disaster, some degree of chaos is inevitable, but this where Sahana's disaster management systems can play a critical role. It can provide a scalable systematic approach by managing large amounts of information, being able to efficiently distribute that information and make it accessible to those who need it on demand, providing a means to automatically collate, aggregate, and calculate based on all available data, and provide reports that are updated live and in real-time.

Sahana is one of several HFOSS projects that are now revolutionizing information management for international disaster response operations. Among others, this landscape includes the following projects and organizations, many of which provide visualization and maps from static sources of disaster information:

- Ushahidi (http://ushahidi.com)
- InSTEDD (http://instedd.org)
- OpenStreetMap (http://openstreetmap.org)

# A Service Industry to Support Sahana

*Mark Prutsalis*

- FrontlineSMS (http://frontlinesms.com)
- Google Crisis Response (e.g., http://google.com/crisisresponse/pakistan_floods.html)
- Crisis Mappers Network (http://crisismappers.net)
- OpenMRS (http://openmrs.org)

Compared to other projects, Sahana software is unique in two ways. First, Sahana software is the only system designed to provide management tools that allow responding agencies to work with the large amounts of data available to them; it allows them to assign individuals to tasks and facilitates management of people, places, and things that are important in disaster relief. Second, it provides a powerful platform for interoperability such that other tools and users can use Sahana software to share information, for example by pulling data from Ushahidi, OpenStreetMap, Google Person Finder, and other sources, aggregate it in one place, and redistribute it via open data standards such as EDXL, KML, or RSS. This allows Sahana to bridge information gaps across the diverse organizations and individuals responding to an emergency, each seeking to manage maximize the impact of their efforts.

Sahana software ensures responders and recipients have current relevant information to respond effectively. Sahana users can:

- register individuals or families at a shelter, and link requests for aid (e.g., food, water, and blankets) or services (e.g., medical assistance, counseling, and family reunification) at individual or location levels
- manage inventories and update the status and capacity of operating hospitals and medical facilities. Missing persons registries can be referenced against those registered at shelters.
- pull requests for assistance and information from crowdsourcing solutions such as Ushahidi, FrontlineSMS, and Twitter. Requests can be managed by assigning tickets to each one, which are used to track responsibility, validity, priority, and status.

By facilitating sharing of relevant information across the diverse set of individuals and organizations that need to communicate with each other, Sahana software fundamentally transforms the chaos of information from a hindrance to an asset, enabling greater self-sufficiency during responses on behalf of communities.

**The Sahana Software Foundation**

The Sahana Software Foundation is the first and most mature HFOSS project and inspired the movement itself (http://en.wikipedia.org/wiki/HFOSS). Sahana was originally developed by members of the Sri Lankan IT community who wanted to find a way to apply their talents towards helping their country recover in the immediate aftermath of the 2004 Indian Ocean earthquake and tsunami (http://tinyurl.com/37lhncg).

Since 2004, Sahana has grown into a global free and open source software project supported by hundreds of volunteer contributors from dozens of countries and it has supported national and local authorities and relief agencies in their response to numerous large-scale, sudden-onset disasters (http://tinyurl.com/2fnrelt). The effectiveness of Sahana software is demonstrated by a track record of successful deployments:

1. In the 2004 Asian Tsunami, the Sri Lanka Center for National Operations used Sahana to identify the location of all relief camps and the demographic breakdown of over 26,000 victims (by age, gender, and illnesses) to help them target relief.

2. Since 2007, the City of New York's Office of Emergency Management has used Sahana for managing its all-hazards sheltering plan, which involves over 500 shelters capable of housing over 800,000 persons and staffed by over 60,000 city agency employees and volunteers.

3. After the Chendu-Sitzuan Earthquake of 2008 in China, Sahana was used by the Chendu police to track over 40,000 families. Following the earthquake, 42 separated family members were reunited using Sahana.

4. After the Haiti earthquake of 2010, Sahana was used as a registry of nearly 700 organizations involved in the response. The software tracked almost 10,000 requests for assistance and information collected by Project 4636 (http://tinyurl.com/25jqfp7) provided the most accurate and complete registry of the 162 operating hospitals and medical facilities, along with bed availability and status, and aggregated 41 data layers from various sources onto one situation map. More than 230 registered users entered data into the system, which was accessed by over 8,600 visitors. The site was used daily by dozens of responding agencies in the first month of the disaster response.

5. The US National Library of Medicine has developed a set of medical-related tools with Sahana as part of the Bethesda Hospitals Emergency Preparedness Partnership.

6. Internationally, Sahana has been adopted by national and local governments, including Sri Lanka, the Philippines, Bangladesh, India, Indonesia, Pakistan, Peru, Taiwan, and China.

The Sahana Software Foundation is recognized as a leader in humanitarian free and open source disaster information management software. It was established in 2009 as a non-profit organization to serve the needs and requirements of a diverse group of customers: government at the national, provincial or state, and local levels, United Nations (UN) agencies, international and local charitable organizations (non-governmental organizations), communities and disaster victims, and technology companies and software developers.

When considering the future of HFOSS and the Sahana Software Foundation's objectives, we recognize the following trends:

1. The world's urban population will increase to 6.4 billion by 2050, when 70% of the world's population of 9.2 billion will live in urban areas, according to the UN (http://tinyurl.com/2wmt2mm).

2. Urban infrastructure spending is expected to approach $350 trillion over the next 30 years, excluding water and sanitation, based on calculations by Booz & Company (http://tinyurl.com/254roy4).

3. Spending on disasters will triple to $185 billion per year by 2100, according to estimates in a UN and World Bank report (http://tinyurl.com/2a8avmg).

Together, these trends represent an incredible opportunity. The Sahana Software Foundation seeks to leverage its humanitarian free and open source software, its own resources and expertise in disaster management, and an active community of volunteers, to significantly enhance the world's ability to mitigate and respond to disasters that threaten increasingly large numbers of vulnerable persons concentrated in densely populated urban environments. But we can not do this alone.

**The Path Ahead: A Cooperative Business Model**

While Sahana is primarily a global, volunteer-driven, open source project, it also has deep roots in the open source business community and has always been supported by that industry. Google and IBM (http://www.sahanafoundation

.org/donors), along with Accenture (http://tinyurl.com/2ef5lfr), have made investments and contributions to Sahana's development. Also, Virtusa (http://virtusa.com) provided testing help for the Haiti earthquake and Pakistan flood responses and provided a significant contribution at the start of the project in Sri Lanka. Further, Sahana has spurred the formation of two commercial companies that were started by experienced Sahana developers to provide Sahana-based services: Respere (http://respere.com) and AidIQ (http://aidiq.com). Sahana has also been a part of several commercial service offerings and successful bid responses.

Although the software will always remain free, part of creating a sustainable HFOSS project includes the successful and viable commercialization of a product with a service industry surrounding it. However, despite the success and recognition that the Sahana Software Foundation has already received, there has admittedly not been enough user adoption to generate enough demand for widespread industry growth; it is simply not being deployed fast enough. One of the factors holding Sahana back is the lack of a qualified service sector to support it. The Sahana Software Foundation hope to grow the service industry that can provide technical support, while providing:

- core codebase maintenance and support functions
- professional user documentation and training
- help in guiding the roadmap for new features based on stakeholder and user input
- promotional assistance to encourage the adoption of Sahana and the principles of collaborative open source approaches to disaster information management

Despite these challenges, there are many indications that Sahana is on the cusp of widespread global adoption. Sahana has been acknowledged by important stakeholders in emergency and disaster management, a message that will reach those who have the financial resources to invest in disaster risk reduction initiatives. An article in the Bulletin of the International Association of Emergency Managers (http://iaem.com) recently noted the uniqueness of Sahana's capabilities that include many useful features that are not included in commercial Critical Incident Management Systems (CIMS) that emergency management agencies typically use (IAEM Bulletin, Vol 27, No. 5, May 2010). Also, a US State Department White Paper on the Haiti Earthquake response noted, partly in reference to Sahana:

*"Within hours after the report of the Haiti earthquake, a new community of virtually connected volunteers affiliated with ICT consulting companies, private corporations, open source software proponents, academic/research institutions, NGOs, and even the Haitian diaspora community began applying new ICT applications to the earthquake response… This new community needs to be recognized as a new player in the humanitarian information environment..."*
(http://tinyurl.com/2vqzdmj)

The business and open source sector has similarly acknowledged the maturity of Sahana through the following awards:

- 2010 Cool Vendor in Risk Management and Compliance, awarded by Gartner Inc. (http://tinyurl.com/35btdof)
- 2010 Best Practices Award, awarded by Private & Public Businesses, Inc. for our collaborative response to the Haiti earthquake disaster (http://tinyurl.com/2fbm3c2)
- 2006 Free Software Foundation Award for Social Benefit, an award that was itself inspired by Sahana (http://tinyurl.com/2cmx2ja)

Having reached this level of recognition, the challenge is to deliver the support that is now expected. Capacity is needed to respond to new re-

quests to have Sahana adopted at national and local levels as part of disaster risk reduction programs. Some of this demand will be fulfilled by the Sahana Software Foundation, but further capacity is needed. We hope that a robust service sector develops to support the efficient and cost-effective deployment of Sahana globally with customers who need it most. These customers are primarily in the public sector, but international organizations also have substantial needs for support services. Industry can provide critical personnel resources to market, deploy, and support HFOSS projects like Sahana. Through marketing, the private sector can help convince organizations of the value of this solution. Through the deployment of Sahana, private companies can enter commercial contracts to provide hosting, along with customization and enhancements on request. Finally, through their support of Sahana, private companies can enter maintenance contracts with a service level agreement that guarantees a professional level of technical and customer support in exchange for a fee. In these ways, industry can help create enough demand to make Sahana sustainable and commercially viable.

One of the means by which an industry around Sahana can grow cooperatively is through a certification and sponsorship program designed to professionalize the industry. This would provide a means for supporting a strong Sahana Software Foundation as a means to ensure the development of best practices around Sahana deployments, ensure that the codebase is well maintained, and provide a guarantee to Sahana's customers that support is being provided with quality control and competence in the provision of emergency and disaster management services.

The needs of disaster victims are urgent and widespread and organizations providing disaster relief have a corresponding need for Sahana's information-management tools. We hope to inspire the open source business community to join with the Sahana Software Foundation in a cooperative model to grow the capacity to address these widespread and growing needs so that there is an industry-level capability to do good. Doing good by helping people is fundamentally what HFOSS is about and this also fits in with corporate social responsibility programs.

In step with Sahana's own social responsibilities, any engagement with the foundation must be done with the highest of ethical standards. The Sahana Software Foundation's Chief Technical Officer, Chamindra de Silva, has been leading an effort to develop an HFOSS code of conduct (http://tinyurl.com/28tp2v2) designed to address issues of competitiveness and collaboration in the name of humanitarianism. Many other HFOSS projects have already agreed in principle to sign on to such a statement and we would expect the same of any company who seeks a partnership with the Sahana Software Foundation or seeks to use Sahana.

**Conclusion**

The open source industry has an interest in supporting a strong Sahana Software Foundation, whose experience, mission, and non-profit status allows it to serve as a trusted agent between a service industry and its customers. The number and impact of disasters is growing and will continue to grow, along with tremendous spending growth on disaster response and mitigation. Because of HFOSS projects like Sahana, there is an opportunity for open source businesses to contribute and to benefit from increased spending on initiatives that mitigate and reduce the risk of disaster. The challenge will be to develop a cooperative and collaborative industry that meets the needs of its growing customer base, while retaining a commitment to the principles around which HFOSS projects are organized. Most importantly, the solution should be born of a desire to help alleviate human suffering.

# A Service Industry to Support Sahana

*Mark Prutsalis*

*Mark Prutsalis is the President and CEO of the Sahana Software Foundation, a non-profit organization that governs the free and open source Sahana Disaster Management System. He is also the Founder and President of Globaliist Inc., an emergency management and disaster response consulting company. Mark has over 18 years of operational disaster response and emergency management experience following major international and domestic sudden onset natural and man-made disasters. He is a member of the International Association of Emergency Managers (IAEM), the National Emergency Management Association (NEMA), and the Humanitarian Logistics Association (HLA).*

# IEEE Humanitarian Projects: Open Hardware for the Benefit of the Poorest Nations

Glenn McKnight and Alfredo Herrera

*"I shall make electricity so cheap that only the rich can afford to burn candles."*

Thomas Alva Edison

An opportunity to solve some of the pressing needs of impoverished nations may be effectively addressed by the active sharing of open hardware solutions. As other articles in this issue of the OSBR demonstrate, open source software is already being used as an effective technology to address humanitarian needs in developing countries. The adoption of open hardware, as an alternative to commercial off-the-shelf products, may be another effective solution to global development challenges. In this article, we will not discuss the socio-economic aspects inherent with global initiatives; but there is a need for meaningful dialogue to come up with appropriate and long-lasting solutions.

The focus of this article is on one of the recent activities of the Institute of Electrical and Electronic engineers (IEEE): the Humanitarian Technology Challenge. The Challenge identified three humanitarian problems that could be solved through technology and challenged participants to find an open-source approach to tackle them. The challenges were Reliable Electricity, Data Connectivity, and Personal Identification Records. This article describes the work part of the Reliable Electricity challenge from three perspectives: i) that of the initial HTC Reliable Electricity team; ii) its offspring, called Community Solutions Initiatives (CSI); and iii) an IEEE Canada group called the Humanitarian Initiatives Committee. All of these groups are part of the IEEE.

**Introduction**

The United Nations (UN) reports that the lack of modern fuels and electricity in most developing countries entrenches poverty, constrains the delivery of social services, limits opportunities for women, and erodes environmental sustainability. Currently, it estimates that 1.6 billion people lack access to electricity and 2.4 billion people lack access to modern fuels for cooking and heating (http://esa.un.org/un-energy).

These populations are forced to use lighting and heating methods that are associated with potentially deadly health risks, have negative impacts on the environment, and cost 5-100 times more than electricity. The poor are spending a disproportionate share of their income on energy when compared to the developed world. This is making the climb out of poverty even harder.

The kerosene lamp is the most common alternative to lighting a home with electricity. The ker-

osene needed to light a home costs most families between 10% and 40% of their income. These lamps emit hazardous fumes that fill the lungs of the residents, particularly the children when they sit close to the lamps to read. Inhaling kerosene fumes is the equivalent of smoking two packs of cigarettes a day. Worldwide, burns and smoke inhalation are responsible for over 322,000 deaths annually according to estimates from 2002 (http://tinyurl.com/2va9ppt). These numbers are probably grossly underestimated, but they provide a sense of the scope of a problem worth eradicating. Electricity is a safer and less expensive technology than kerosene; and affordable and reliable electricity for basic lighting and low energy appliances can certainly improve the living standards for families in developing countries.

Medical clinics in rural communities are in desperate need of lighting and refrigeration for blood storage. For example, maternal mortality worldwide accounts for more than half a million deaths a year, and 99% of these deaths occur in underdeveloped countries (http://tinyurl.com/33lhbok). The availability of reliable electricity for these clinics will greatly reduce these staggering statistics and the human tragedy they represent.

**IEEE Humanitarian Technology Challenge**

In 2008 the IEEE Humanitarian Technology Challenge (http://ieeehtc.org) was formulated as a joint effort funded by the IEEE Foundation, the Vodafone Foundation, and the UN Foundation. The objective was to have a major impact on the eradication of poverty by supporting the UN Millennium Development Goals (http://un.org/millenniumgoals). The Challenge would contribute to these goals by inventing a range of useful advances in technologies. These advances would rely solely on non-governmental organizations (NGOs) or new for-profit ventures to develop deployment opportunities by means that would be determined through the Challenge's activities. The Challenge was designed to be a collaboration between technologists (led by IEEE) and humanitarians (led by UN Foundation) with the objective to identify humanitarian problems that could be solved through technology. Further, it challenged participants to find an open-source approach to tackle the problems they identified. Through collaboration online and in workshops, the Challenge relied on the altruistic and voluntary participation of technologists, humanitarians, nonprofit organizations, students, and government employees.

In 2009, a Humanitarian Technology Challenge conference brought together over 150 attendees with approximately equal mix of humanitarian and technology representatives. Three selected challenges were documented and ratified by working groups that were tasked to find solutions for them and the first of these challenges will be the focus on this article:

**1. Reliable Electricity:** This includes low-power stationary facilities; rugged, mobile power supplies for emergency settings; mechanical transducers; passive generation devices (e.g., charge as you walk); and renewable energy hubs.

**2. Data Connectivity of Rural District Health Offices:** This includes two-way transmission; daily data batch transfer; emergency and outbreak alerts; more affordable or higher-bandwidth services; mapping of existing global connectivity; and the creation of a data relay network with intermediate field offices.

**3. Individual ID and Tie to Health Records:** This includes secure, confidential ID for patients; emergency response and chronic care applications; and routine care for migrant populations.

A follow-up workshop focused on defining plans for field tests of solution ideas. It brought together about 70 people, with representation from 15 countries. The workshop focused on defining plans for field tests of solution ideas, including

input from representatives of NGOs to address the realities of field test implementations, along with funding and partnership opportunities. From these efforts, three groups have emerged to address the Reliable Electricity challenge:

**1. Reliable Electricity**: The Reliable Electricity team (http://ieeehtc.org/index.php/htc/challenges/electricity) adopted a technology-transfer approach, which was prescribed at the Humanitarian Technology Challenge workshop. With this approach, the team would define a system that addresses an identifiable set of needs. For example, instead of focusing on metropolitan areas that have sporadic access to electricity, the team would focus on rural areas that lack access to electricity. This type of solution requires the help of local institutions, such as health services, to support the adoption and diffusion of the new technology.

The Reliable Electricity team is focused on a 250 Watt power generation and load management system to provide a scalable, renewable energy hub for the Reliable Electricity challenge. The objectives for this project were to firstly produce a working system that was configured to gather data on user behavior and field stresses, and secondly to test a first cut of the design of a power system that could be totally self-managing and operable by people that were totally unfamiliar with electrical technology and were perhaps even illiterate. These two issues were the main drivers for a test site selection.

**2. Community Solutions Initiative (CSI):** The CSI team (http://communitysolutionsinitiative.org) adopted a human-needs approach based in part on the philosophy of Paul Polak (http://paulpolak.com). This approach builds on community-based solutions created with organizations in the field. The team consists of a core group of professional engineers plus a spectrum of multidisciplinary professionals from medical, industrial, educational, legal, business, marketing and research fields. Membership is not restricted to IEEE members or to the engineering professions. The CSI group was formed specifically to address the key transition problems of: i) how to initially deploy such technologies; and ii) how to partner with NGOs to create local business models that empower local entrepreneurs to maximize growth of local economies. NGOs are key to the solution because of their established community relationships, but not all NGOs have the needed entrepreneurial business skills to make good entrepreneurial partnerships "on the ground."

The core principles of the CSI are:

1. Dedication to developing indigenous entrepreneur opportunity and reinvestment in the local community. This is business development, not charity.

2. All technology specifications, designs, business, and operations plans are non-exclusive and open source.

3. All CSI partners work on a pro bono basis as a contribution to their professional organizations.

4. All individual and business partners of CSI, both non-profit and for-profit, are dedicated to a success metric that sees their efforts helping millions of people emerging from extreme poverty.

**3. IEEE Canada Humanitarian Initiatives Committee:** This committee (http://ieee.ca/hic) was created to support IEEE members participating in humanitarian initiatives or building relationships. Its mission is to enable members to fulfill IEEE's strategic vision of being "universally recognized for the contributions of technology and of technical professionals in improving global conditions" (http://www.ieee.org/about/corporate/strategy/index.html). The committee

promotes IEEE's core value of advancing technology for the benefit of humanity by raising awareness of how IEEE Canada can best use its strengths and relevant technologies to address societal problems. In practice, this means that the committee will support the work of IEEE members involved in:

- relief assistance during natural disasters, such as ice storms and floods
- socio-economic development abroad, such as improving affordable electricity access in developing countries
- awareness initiatives in our communities, such as student design competitions with humanitarian applications

As an example of these activities, the committee recently launched its first student design competition. The competition consists on improving, extending or innovating WE CARE Solar's system (http://wecaresolar.org). For this project, WE CARE and the committee are collaborating in defining a version of WE CARE's system that the committee will make available as open hardware to participants in the competition; students will thus be able to contribute back their ideas as open hardware. This work is a reference model being presented by the committee to the other IEEE groups mentioned in this article, to enable their work.

The committee is more of a facilitator than an agency. It adopted principles from the appropriate-technology approach (http://wikipedia.org/wiki/Appropriate_technology), and it seeks to understand the socio-economic context of the communities it is working with before selecting a suitable strategy. For example, it is following ideas from structurist development to build the capacity to address future humanitarian projects by leveraging the IEEE culture of conferences and workshops for its current student competition.

**Open Source and Open Hardware**

The projects described in this article are committed to following an open source, collaborative model. The Humanitarian Initiatives Committee helps define and promote the details of what open source software and open hardware means in the context of the IEEE humanitarian initiatives.

Negash and colleagues (http://aisel.aisnet.org/sais2007/33) explored the differences in the adoption behaviour of open source software between economically developing countries and industrialized countries. Their case studies lead them to identify four factors to consider in developing countries: reward and compensation, local competency, piracy, and intellectual property laws. If we adapt their conclusion to open hardware, we expect to find similar behaviour: unless there is an economic incentive for individuals and organizations in developing countries to adopt open hardware, adoption will be slow; and piracy practice in developing countries may have the viral effect intended by the open source model, but it will probably work against the creation of a sustainable ecosystem. Many developing countries depend on the government to pay for ICT projects; without this support, it is often too costly to adopt unfamiliar open source software (and hardware, by extension). Also, it has been observed that user adoption is influenced by access to a network of collaborators. Our interpretation is that open hardware, even if made widely available by unlicensed imitation, may become victim to this apparent success if there does not exist a network of volunteers committed to supporting the open hardware.

Because of the inherent openness of the volunteer work of the IEEE, patents have not been an option on any of these humanitarian projects. During the development phase, there has been no need to work on the reward and compensation factor referenced in the paragraph above. Volunteers have been working for altruistic reas-

ons, but now that the first field trials are being rolled out, questions about liability, intellectual property protection, and meeting reliability expectations are being raised. The use of open source clearly becomes fundamental to the sustainability of the IEEE humanitarian projects. A license can probably address all of the issues that are part of the design process, while a contract can probably best tackle issues that are part of the manufacturing and procurement transactions of the assembled open hardware systems. In both instances, the open source model is the most natural to adopt. Depending on the project, open hardware can be free or commercial. For example, the sustainability of the CSI project is based on the expectation of an operating profit from the use of its open hardware mobile charging station. There is a growing number of companies making millions from open hardware (http://tinyurl.com/2ehovx8). This model has tremendous potential and that is why it has been considered by the IEEE in at least one instance.

**IEEE Open Hardware Projects**

The following section summarizes five open hardware projects and their contributions to the overall goal of providing electricity to those in need:

**1. HTC Reliable Electricity:** This project is producing a 250W, integrated power-on-demand electricity supply and management system for a community centre, medical clinic or individual dwelling. This stand-alone hybrid power system that uses a 24VDC bus to minimize wire size and cost. It provides power for fans, lighting, water pumping, refrigeration, radio, and cell phone charging. All inputs and outputs have current monitoring, programmatic disconnect, and auto-resetting circuit protection. The built in data logger can collect user and environmental information for months in non-volatile memory. There is also a trial load shedding algorithm meant to optimize battery usage. The initial design is over-instrumented to provide field study feedback, later versions will be simpler.

**2. CSI Mobile Charging Station:** The Sirona Cares Foundation (http://sironacares.org) and the CSI are jointly developing and deploying sustainable businesses in Haiti to provide renewable electricity to 1 million Haitian people. This program is centered on giving Haitians the ability to earn a living by providing electricity to their community. CSI has brought together highly trained engineers to design a generating system that is easy to deploy, operate, and maintain. Together, CSI and Sirona have developed sustainable business models around this generating system. Sirona is now working with its partners in Haiti to deploy these models and integrate sustainable businesses into communities. The businesses will use a combination of renewable electricity generators (principally, solar) to provide a battery-charging service.

**3. CSI WindTurbine Task Force:** CSI is collaborating with professors and students at Seattle University, University of Washington, and the Seattle chapter of Engineers Without Borders to build a small-scale wind turbine. The deployment of the wind turbine into rural Africa is projected for the summer of 2011. The wind turbine is designed to be made out of materials that could be assembled in country with minimal instruction and training. The output of electricity would be stored in "community charging stations," where members of local and surrounding villages could come to refill their home batteries.

**4. CSI Pedal Power:** CSI is teaming with IncSys (http://incsys.com) to build a pedal-powered generator that is affordable and easy to operate. The project is called Power2Light and will allow any community member to assemble and run a micro-generation business out of their home. With 10 hours of pedaling, energy can be created to light over 100 homes.

# IEEE Humanitarian Open Hardware Projects

*Glenn McKnight and Alfredo Herrera*

**5. Humanitarian Initiatives Committee Student Design Competition:** Participants in the student design competition (http://tinyurl.com/2uj4evv) will be asked to propose ideas that will improve or extend the WE CARE solar suitcase system (http://www.wecaresolar.org), which is available under open hardware licensing. Starting points for potential projects include simplifying the installation, creating a collection of direct current (12V DC) medical devices, enabling the use of Li-Ion batteries, enabling the use of electric hand-held tool batteries, optimizing the charge controller, improving the system's serviceability and cost, improving the system's enclosure, innovating the connectors and cabling, and enabling recycling and sustainability aspects. The student design ideas will be released as open hardware.

**Conclusion**

Through the Humanitarian Technology Challenge and the related humanitarian projects of the IEEE, open hardware projects hold great potential for tackling the challenges facing the most vulnerable. Several of these projects are just beginning to gain traction and we look forward to seeing real impacts on this significant humanitarian problem. The IEEE aims to make a tangible impact by advancing technology for humanity.

*Glenn McKnight is the Humanitarian Technology Challenge Reliable Connectivity Liaison of IEEE Canada's Humanitarian Initiatives Committee. He has worked extensively with Canadian and international private and public sector organizations to promote cost effective IT and non-IT projects. His work included Baygen Radio of South Africa, environmental technologies in China, and IT strategies in India. His certification experience includes developing apprenticeship programs, operating IT schools, and promoting the Linux Professional Institute as an international standard.*

*Alfredo Herrera is a senior member of the IEEE and the Chair of IEEE Canada's Humanitarian Initiatives Committee since its creation. He has over 12 years of experience in telecom system validation and digital hardware design and verification. He currently works for Ericsson in Ottawa as a Radio Hardware Systems Integration and Verification Engineer and previously, he worked for 10 years at Nortel Networks. He has been a member of the IEEE since 1996 and has been Vice-Chair of the Ottawa chapter of the IEEE Technology Management Council and the IEEE Society on Social Implications of Technology since 2005. He is also a Master's student in the University of Ottawa's Systems Science program, and his research topic is on using technology for development.*

## Recommended Resources

- CSI Open Hardware Blog
  http://csiopensource.wordpress.com
- IEEE Humanitarian Technology Network
  http://www.ieeehtn.org/
- ICT 4 Peace
  http://wiki.ict4peace.org/FrontPage/
- Value of ICT in Humanitarian Relief Efforts
  http://www.inveneo.org/ict-value-haiti/
- The IEEE partnership and ASME Engineering for Change Initiative
  http://www.engineeringforchange.info/
- Free Charge Controller
  http://www.freechargecontroller.com
- Yahoo Discussion Group, Charge Controller
  http://groups.yahoo.com/group/charge-controller/

# Mifos: Ending Poverty with Software

Adam Feuer

*"My vision for the future? A world completely free from poverty."*
Muhammad Yunus
Winner of the 2006 Nobel Peace Prize

Free and open source software (FOSS) and business are key elements for solving the largest problems facing humanity. In this article, we focus on the challenge of eradicating poverty, a significant economic condition that affects billions of people worldwide. We outline the role of microfinance – the use of small loans to help poor people establish businesses – in eliminating poverty. We then describe the Mifos Initiative, a FOSS project to provide banking software to support microfinance institutions. Although we focus on the specific case where software is being used to address the problem of poverty, the approach and lessons learned can be applied to other great problems facing humanity.

**Introduction**

Poverty, ecological devastation, war, sickness, and the survival of our species: these are solvable problems. Even though they have been with us for thousands of years, they do not have to be with humanity forever. To understand the potential we have for overcoming these problems, we should examine other great problems that have been solved throughout history. For instance, legalized slavery was a constant feature of human societies and human suffering for thousands of years, but in 1787 a team started working to eliminate it. In 1833, it was ended in the British Empire (http://tinyurl.com/kmduvh) and that achievement provided enough momentum to end legalized slavery worldwide soon afterward. A second example is the worldwide eradication of smallpox. By the 1950s, the disease was killing 2 million people per year and, over the course of history, it killed hundreds of millions of people. In 1967, the World Health Organization formed a team to work on the elimination of smallpox, and it was successfully eradicated by 1980 (http://wikipedia.org/wiki/Smallpox).

Both of these efforts used global teams that collaborated with each other by openly sharing and improving the technologies they used to engage with their problems. Both efforts are among the pinnacles of human achievement, relieving tremendous suffering and creating great positive economic results. We consider these efforts, and others like them, the virtual ancestors of open source business.

**The Poverty Problem**

Poverty has been with human society since the beginning of civilization, but unlike legalized slavery and smallpox, poverty has not yet been eradicated. Billions of people still suffer from this curable economic condition. There are between 2 and 3 billion people living in poverty now, depending on how you count (http://wikipedia.org/wiki/Poverty#Demographics). The UN Millenium Development Goals call for poverty's elimination by the year 2050 (http://un.org/millenniumgoals/poverty.shtml).

Poverty is not just an abstract measure – Grameen Foundation (http://grameenfound

ation.org) has concrete measures for describing poverty, and these are immediately understandable:

- Do the family's children eat three good meals a day?
- Do they go to school?
- Does the family have a house with a roof that will withstand a severe storm?
- Does the family have healthcare?

Another way to describe poverty is by purchasing power. If the family earns less than the equivalent of $1.25 USD per day, they are *extremely poor*; if the family earns $2 or less per day, they are *poor* (http://wikipedia.org/wiki/Poverty#Demographics). These and other questions can reliably measure whether a family is in poverty.

**What Is Needed**

We believe that software and business are two key elements for solving poverty. First, we do not assert that software *alone* is enough to end poverty – poverty has many factors, and eradicating it will require many innovations. Rather, we assert that any solution to ending poverty will need software in order to reach all poor people in the time given. Second, we assert that business must play a role in the effort to eliminate poverty as an effective social technology for generating wealth. Eliminating poverty for 2 or 3 billion people is something charity cannot do – there simply is not enough wealth to go around, even if people were willing to part with it. And while governments have had some success redistributing wealth, they have not been very good at generating it. Furthermore, redistributing wealth has never alleviated poverty.

**Microfinance**

Where software and business come together in the service of poverty is in the field of microfinance. Microfinance is the business of providing financial services to poor people. To demonstrate how it can eliminate poverty, the process of microfinance is summarized below.

People are generally smart and entrepreneurial. However, poor people often lack the capital to better their situation. Traditionally, the only sources of capital available to poor people are those in their community, village moneylenders and the like, who often charged usurious interest rates, made arbitrary investment decisions, and did not have the well-being of the community in mind.

Microfinance institutions (MFIs) are for-profit, social businesses (http://wikipedia.org/wiki/Social_business) that lend money in small increments at reasonable interest rates, in amounts often less than $100 for a one-year loan. The loans are made mainly to women. Men pay back the loans just as reliably as women, but lenders have found that men mainly use the money on themselves. Women generally use the money on their children, their family, and their parents. Given scarce resources, loaning mainly to women is a more effective way to help people raise themselves out of poverty.

Women use the money to start businesses that earn them income, for instance, to buy a goat or a sheep, to buy material to make furniture for sale, to stock a small store, to start a restaurant, or to buy a mobile phone they can rent to their fellow villagers. These small businesses bring in more income for their family than the women could otherwise provide. This is the difference between their children going hungry and having

enough food, between no school for the children and paying their school fees, between having a roof that blows off in a big storm and having a storm-proof roof. Over time, these businesses are the difference between poverty and having enough.

Microfinance also refers to savings, not just loans. Poor people often have erratic or seasonal income, and access to a safe savings account allows them to save money to even out their income over time, reducing the impact of emergencies and allowing them to put their energy toward their family and work rather than worrying about how to get by through other means, like buying and selling assets. It also gives them security compared to alternatives like stashing money in their mattresses where it can be stolen.

Microfinance has a strong track record. For instance, in Bangladesh microfinance institutions have helped 30 million people raise themselves out of poverty. Since microfinance banks charge a fair interest rate, the institutions earn money and can operate as a business, expanding if they are good at serving their customers and providing a fair return to their investors.

**Microfinance Software**

While the benefits of microfinance are great, there are several constraints on the growth of MFIs. Two important barriers are lack of capital and lack of software. The lack of capital is related to a lack of transparency; investors have a hard time finding and monitoring MFIs that are financially sound and socially successful. Without investors, MFIs cannot grow quickly. Lack of software is important for the operation of MFIs, enabling them to grow beyond small-scale operations that can be managed on paper or spreadsheets. MFIs with centralized software can operate hundreds of branches, reaching hundreds of thousands, or even millions, of poor clients.

These two major barriers are related. Centralized portfolio management and accounting software help MFIs know what is happening in their own business on a day-to-day basis, and can also be used to give their investors a view of the same information. However, most MFIs do not have access to good, centralized software that can be used to operate a socially and financially successful business.

**The Mifos Initiative**

The Mifos Initiative (http://mifos.com) is a humanitarian FOSS project that aims to remove constraints on MFI growth. We make banking software that MFIs can use to solve their operational and transparency needs. MFIs can download Mifos Enterprise software for free to run on their own premises or they can run the software on the web with a paid subscription to Mifos Cloud, our software-as-a-service application.

One of the software's main strengths is centralized management via a web-based user interface that MFI branch offices can access using a web browser. This allows MFIs to instantly know what is happening in all their branches, at the same time as minimizing the IT complexity and costs, since all they need at each branch is a computer with a web browser. This lower complexity enables MFIs to open new branches rapidly when they want to expand to meet market demand for their services.

The other main strength is business intelligence. We incorporate the Pentaho Business Intelligence Suite (http://pentaho.com), a leading business intelligence tool chain that is also FOSS. This gives Mifos customers unparalleled reporting and data analysis capability, allowing MFI executives see their financial and social results clearly so they can steer their business toward profitability and effective poverty reduction.

Mifos also includes Grameen Foundation's Progress out of Poverty Index (PPI; http://progressoutofpoverty.org), a statistical methodology for measuring an MFI's social performance with the same rigor as it measures its financial performance. This allows MFIs and investors to know whether the MFIs products and services are actually helping people raise themselves out of poverty.

**Successes and Challenges**

We have had some great successes with Mifos: our software runs MFIs with hundreds of thousands of poor clients and its specialty is helping MFIs expand rapidly. Our customers are in Africa, India, the Middle East, the Philippines, and Mexico. We have won awards for our software, including the Java's Duke Award in 2009 (http://tinyurl.com/lvltlb).

However, with a project of this sort, the challenges are large. One particular challenge with Mifos is that developers cannot "scratch their own itch" by contributing to the project to solve a personal challenge. While people may use software run by a bank, no one ever runs banking software for their own personal use. And since Mifos is specialized around microfinance, which is used primarily by poor clients and has no online banking functionality, it is unlikely volunteer programmers would ever encounter this software in their daily life.

To work on this project, people must have a completely altruistic motive or else be paid to work on the project - there are no hobbyists using Mifos themselves. This presents a barrier to growth, since it is hard to find capable volunteer programmers with a solely altruistic motivation. Despite this challenge, we have found a handful of very capable, dedicated volunteers and are seeking additional people who wish to contribute directly to eradicating poverty from the Earth by helping MFIs be more effective.

We can use additional volunteer programmers, test automation developers, manual testers, translators, technical writers, user experience experts, product managers, and product managers. To help meet this challenge, we make a point of offering projects to match a volunteer's desired level of involvement, from developing entire features, to improving user interface design, to fixing bugs that prevent new MFIs from using Mifos in their region of the world.

Volunteers are encouraged to contribute to Mifos because they are interested in helping others as a means to make themselves happy. Many people, especially open source enthusiasts, enjoy programming and helping others. Many open source projects are helping people in the industrialized West, people who have easy access to technology. We encourage open source and business enthusiasts to work to help people who have less access to technology, or who have bigger, more difficult problems. We can help our entire world society, our species, or even our planet. We believe it is a direct path to happiness for oneself and others.

Another challenge that we have, that we share with many FOSS projects, is that we are a small, but global project. This means that our programmers are in many different time zones, in the USA, India, Africa, and Europe. Making decisions in such an environment can be difficult, particularly because we want supple, flexible software that meets customer needs and is fast and fun to develop on.

A key idea that we have come to believe is that "Team = Product." If the goal is for the software to exhibit certain characteristics, the team must also embody these same qualities. Because of this, we use the Core Protocols (http://liveingreatness.com/) methodology and put a lot of care into our team's shared vision. A team with shared vision can trust its sub-teams to make good decisions on their own without consulting

with the whole group beforehand. Consistently making good decisions at all levels of a project leads to supple, effective, and fun software.

We also follow Scrum (http://wikipedia.org/wiki/Scrum_(development)) and Extreme Programming (XP; http://wikipedia.org/wiki/Extreme_programming) methodologies, which rely on high-quality decisions and shared code ownership to encourage innovation. These methods also promote heavy use of automated tests and continuous integration to detect problems right away; these are techniques that we have found helpful on a distributed team.

**Why Are We Open Source?**

The Mifos software is still a small project, comprised of about 40 developers on four continents. Some of our developers are paid by Grameen Foundation, others by MFIs, and still others volunteering their time. Our corporate partners include ThoughtWorks (http://thoughtworks.com) and SunGard (http://sungard.com), as well as our customer MFIs worldwide.

Grameen Foundation started out believing that just having a FOSS project would bring many individual volunteers to help us build and maintain our software. We labored under this misguided idea for several years, not attracting many contributors, until we started to understand the "scratch your own itch" problem discussed above. Eventually we came to learn that MFIs would contribute to Mifos, or pay others to do so, because they wanted to minimize their ongoing software maintenance costs. If they built their own software, or even made private modifications to Mifos themselves, they would be responsible for keeping the software additions running as Grameen Foundation released new versions. This would be difficult and costly for them. If they contributed the changes back to Mifos, there would still be some effort to ensure the software worked at each release, but the cost could be shared among many other MFIs who also wanted the same features. This reduces each MFI's ongoing maintenance costs, and in our case, is one of the main value propositions to our customers.

Another way open source helps our business is that we can effectively use the same technique that MFIs use on our own software – incorporating work from other FOSS projects to enhance Mifos' capabilities, and sharing the maintenance burden with other "customers" using the same projects. Several examples of this are the main components that Mifos is based on: Spring Framework, Hibernate, MySQL, Tomcat, and more recently, Pentaho. Like the MFIs that contribute money and work to us, we contribute money and work to these projects too, and by using this model explicitly we can develop it as a strength.

Lastly, from a philosophical point of view, Grameen Foundation believes strongly in transparency and freedom. FOSS embodies both of these values as well, so it is a natural match.

**Conclusion**

In this article, we have shown how Mifos is contributing to the eradication of poverty through an open source project and discussed some of the challenges and how they are being met. However, this approach and its lesson can be generalized to other problem areas, such as war, sickness, ecological devastation, and the survival of our species. Each of these areas is so large that businesses are vital to quickly developing the innovations, scale, and speed sufficient to reach a solution. In this issue of the OSBR, the approaches taken by various projects have been described and we encourage others to take these insights and apply them to further humanitarian challenges.

# Mifos: Ending Poverty with Software

*Adam Feuer*

*Adam Feuer is Director of Engineering at Grameen Foundation's Mifos Initiative. He founded the network scalability solution provider F5 Networks, the internet service provider IXA (now part of Savvis Networks), and helped launch Amazon's grocery delivery business Amazon Fresh. He specializes in mission-critical business applications built using open source software and Agile methodologies. His passion is creating great teams to solve difficult problems.*

## Recommended Resources

- *Creating A World Without Poverty*
  by Muhammad Yunus
  http://amazon.com/dp/1586486675

- *Banker to the Poor*
  by Muhammad Yunus
  http://amazon.com/dp/1586481983

- *Software For Your Head*
  by Jim McCarthy and Michele McCarthy
  http://liveingreatness.com/software-for-your-head.html

- Progress out of Poverty Index
  http://www.progressoutofpoverty.org/

- MFI Transparency
  http://www.mftransparency.org/

# Emerging Open Source Health Information Business Ecosystems in Resource-Poor Environments: the OpenMRS Experience

**Dawn Smith**

*"It self-organizes and evolves. It creates diversity, not uniformity. That's what makes the world interesting... and that's what makes it work."*

Donella Meadows

OpenMRS is a rapidly growing open source community that supports emerging healthcare environments around the world through the collaborative development of an open source medical record system platform. This article provides a brief history of the project, including how the humble beginnings of the collaborative have grown to unlock business opportunities for indigenous populations and also provide sustainable, scalable, locally supported health information infrastructures.

**Background**

Healthcare is a distinctly information-centric business. The care environments that effectively capture, process, and re-use clinical information are typically more efficient and produce better healthcare outcomes. These functions are even more critically important in the resource-poor areas of the world, which are most often forced to support higher disease burdens per capita with fewer human and infrastructural resources. The onset of the HIV/AIDS pandemic in the 1980s spotlighted the disparity of resources between developed and developing countries. Resources such as basic medical care, human capacity, and physical infrastructure were absent from the countries that also carried the heaviest burden of infectious diseases, which often led to dramatic drops in life expectancy.

As a response, initiatives such as the U.S. President's Emergency Plan for AIDS Relief (PEPFAR; http://pepfar.gov) and The Global Fund to Fight AIDS, Tuberculosis, and Malaria (http://theglobalfund.org) provided aid to help establish health systems in environments without infrastructure. Consistent with developing health systems, there grew an increasing focus on the development of health information technology (HIT) which supported routine clinical care processes such as clinical and hospital visits.

It was within this context that the collaboration that led to OpenMRS began. In 2004, early independent work by the Regenstrief Institute in Indianapolis (http://regenstrief.org) and Partners in Health (PIH; http://pih.org) in Boston encouraged both groups to find collaborative partners for developing patient record systems. In each team's work with health-system-strengthening initiatives (Regenstrief's support of the Academic Model Providing Access to Healthcare [http://iukenya.org/hiv.aids.html] in Western Kenya and PIH's support of environments in Rwanda and Haiti), there was a shared recognition that an information system's ability to grow

would only come with the sharing of expertise, experience, and limited resources. This commitment to each other's success was manifest in a deep health informatics collaboration between the two groups that led to shared designs and software development which unknowingly served as the formation of an open source community. The decision to use freely available online tools to facilitate collaboration between the two groups made it easier for other organizations to learn more about the work and participate at all levels in the community.

As a result, since 2006, OpenMRS grew from utilization in Kenya, Rwanda, and South Africa into a community-driven medical record system platform implemented in over forty countries throughout the world. This platform is comprised of a growing number of software components and collaboratively developed designs that can be mixed and matched to develop a patient record system application. The community involved in this work consists of hundreds of people who work together collaboratively to meet the massive demands for basic HIT in resource-poor settings. They come from all walks of life: from clinicians who work directly with patients in Africa to technologists who have no direct healthcare experience but want to do something philanthropic with their time. OpenMRS in many ways is a vibrant example of what is now more commonly known as a humanitarian free and open source software (HFOSS) organization. What started out as a partnership formed out of need has evolved into its own not-for-profit, which has a community-coordinating mission in support of sustainable enterprise patient record system deployments.

**Health Information Business Ecosystems**

The OpenMRS community believes deeply that the key to sustainable patient record systems in poor environments is indigenous local health informatics capacity. All too often, aid-supported environments do not have health information workers readily available, which leads to externally implemented and supported information technology, much of which is proprietary in nature. In turn, this leads to all sorts of problems as the healthcare practices naturally evolve within these environments, necessitating the technology's evolution as well. "Western" organizations struggle to understand the cultural contexts that make information systems fit within clinical work flows, and health systems have rapidly evolving demands as they become increasingly information fluent. These tensions lead to high levels of failure and health systems that are poorly empowered to take advantage of their own health data. Additionally, this model of business is not consistent with the goals of humanitarian aid, which seek to empower self-reliance in emerging health systems and support the sovereignty of countries as they emerge into middle and higher-income economies.

We see humanitarian FOSS as a remarkably positive disruptive market force as it relates to HIT in aid-supported environments. Using the OpenMRS example, the community proactively makes all of its proceeds freely available through the Internet and other means and puts considerable effort into capacity development. So not only does it serve free food, but it also gives away the recipes alongside all of the ingredients as well. It also holds regular cooking classes. By helping establish local informatics "cooking" capacity, new forms of business opportunities form. These newly skilled workers can then leverage an international community for help and peer mentorship while participating directly in the massive infrastructure building activities that are taking place within their environments.

HFOSS empowers health systems to collaboratively participate in the construction and maintenance of their own medical record system, whether through hiring local health information workers directly or in collaboration with newly

formed HIT-based service organizations. In a vibrant community built upon an open platform, individuals and organizations "compete" with one another based on the quality of service or knowledge they bring to an environment, as consumers can choose from a variety of workers who all work from the same technical means. It also gives these health systems more control of their own health information.

We are seeing these very patterns emerge in multiple countries throughout Africa. For example, in Rwanda, the local Ministry of Health has chosen to begin a project to implement OpenMRS broadly as the electronic medical record used in outpatient clinical settings. As a result, they have developed their own functional requirements specifications, and have requested external aid to help build capacity in implementing and developing OpenMRS-based applications. This training takes many forms, the most exciting of which is a formal training partnership between PIH and the Kigali Institute of Science and Technology (http://kist.ac.rw). Graduates from these training opportunities are then either hired directly by the Ministry of Health or are increasingly being paired with business-savvy individuals and seed capital from the Rwandan Development Board to help them establish their own HIT-implementation organizations. These workers will then be called upon to carry out both the initial implementation and long term support of the more than 400 clinics located throughout the country.

The perhaps more entrepreneurial nature of Nigeria is encouraging a more emergent business ecosystem. We are aware of more than a half dozen new, self-identified HIT companies that have formed in support of local health systems that are using OpenMRS as a foundational piece of their strategy. Additionally, self-forming local collaboratives in Nigeria, such as the one led by the Institute of Human Virology, are creating the demand side of the business by bringing health systems to the HIT workers. In Kenya, a local enthusiast group is forming virtually, in response to a recent publication of electronic medical record functional requirements by the Ministry of Health, and it is their intention to create a Kenyan-specific "distribution" of the OpenMRS software consistent with those requirements. There are other examples of such pro-business ecosystems, each forming in a way that is consistent with the socio-cultural contexts of the environment.

**Lessons Learned**

The cases in Rwanda, Nigeria, and Kenya represent only a few examples of the emerging successes OpenMRS has seen where a community, even in a resource-poor environment, can harness and build local health-informatics capacity to create a sustainable system. Yet we have to ask ourselves: What are the lessons we can learn from the success of others to create a sustainable model for all communities? From the perspective and experiences of OpenMRS, much of the growth seen within these environments extends from a focus on the following lessons:

**1. Enable community ownership:** If the community does not have ownership of the tools needed to succeed, then the community sees little, if any value, in the project. Volunteers in the community contribute their best ideas, time, and talents to the organization and development of the software. Given that OpenMRS asks these contributions to be freely shared with the rest of the world, we believe in the importance maintaining a culture of shared ownership.

**2. Focus on real, not perceived needs:** Potential solutions come from individuals who have an idea of the problem; sustainable solutions come from individuals who *know* the problem. In resource-poor environments burdened heavily by infectious diseases, it is vital to respond to the real, not perceived, needs of that community.

End users and developers collaborate through various forums to prioritize design features of the software. This healthy design tension ensures that end users receive the features they need for daily workflows and that developers can scale the code appropriately.

**3. Re-use what works:** Redundant efforts are hazardous in resource-constrained environments. Prior to FOSS, many organizations re-created ideas and code that already existed but were inaccessible due to copyright and licensing restrictions. Redundancy and inefficient responses in many HFOSS environments cost more than time; they also cost patients' lives. OpenMRS built its software upon the underlying design developed by the Regenstrief Institute. Blending this design with the work from open source software projects allowed OpenMRS to develop efficiently and quickly, and in turn, OpenMRS hopes to serve as a foundation for other open source projects in the future.

**4. Promote active transparency:** Active transparency establishes earned trust within the community. As an organization, OpenMRS openly shares both its successes and failures. In doing so, the response is two-fold. Community members actively work to build upon and improve the success of the software while preventing the reoccurrence of the same failure, and in turn, the same individuals share their own stories and encourage others to do so as well. What we have seen is the community connecting through various forms of communication (face-to-face meetings, weekly conference calls, online forums such as Internet Relay Chat, and message boards), which has taken on a movement of self-organization.

**Conclusion**

The lessons described in this article represent some of the core values and experiences of OpenMRS, but they certainly do not represent the only lessons that have been learned. Whether in the foundations of an emerging business ecosystem or in the daily workflow of an HFOSS organization, these lessons contribute not only to the success and sustainability of those entities, but to the health informatics capacity in resource-poor environments as well. The success experienced by such systems represents a tide that raises all ships; when the community succeeds, we all succeed.

*Dawn Smith is the Project Coordinator for OpenMRS. She completed her Master of Public Health (MPH) degree at Indiana University with a focus in both social and behavioral sciences and international health. In 2007, she worked with AMPATH's Psychosocial Department in Eldoret, Kenya, and in 2008, worked at the Institute of Development Management in Gaborone, Botswana. Dawn is currently in the process of completing the necessary steps to formalize OpenMRS into a non-profit organization.*

# The Humanitarian FOSS Project

## Ralph Morelli, Allen Tucker, and Trishan R. de Lanerolle

*"Therefore search and see if there is not some place where you may invest your humanity."*

Albert Schweitzer

The Humanitarian Free and Open Source Software (Humanitarian FOSS) Project is primarily an educational project whose goal is to engage more undergraduates in building free and open source software (FOSS) that benefits their community. Over the past four years, increasing numbers of undergraduates and computer science programs have been inspired by the Humanitarian FOSS project to make significant contributions to several active open source software development projects that have benefited organizations such as the Portland, Maine Ronald McDonald House, and the New York City Salvation Army. This article provides examples of several Humanitarian FOSS projects and describes other initiatives aimed at promoting undergraduate education about FOSS and its application within the community.

**Introduction**

The Humanitarian FOSS Project is primarily an educational project whose goal is to engage more undergraduates in building FOSS that benefits their community. It began in 2006 as a collaborative effort among computer science faculty at Trinity College, Wesleyan University, and Connecticut College in Connecticut.

The word "humanitarian" in Humanitarian FOSS is meant in the broadest possible sense to mean all contributions that benefit the public good. This includes contributions to global and international software projects as well as those within one's local community.

Over the past four years, increasing numbers of undergraduates and computer science programs have been inspired by the Humanitarian FOSS project to make significant contributions to several active open source software development projects. In this article we provide examples of projects in which these students and programs have been engaged. We also describe other initiatives that we have taken to help ensure the sustainability and future growth of Humanitarian FOSS in undergraduate computer science education.

**The Sahana Project**

As described in the April 9, 2010 OSBR column (http://tinyurl.com/2f5mf8x), the Humanitarian FOSS project was born following an independent study project involving a small group of students and faculty at Trinity College. In January 2006, they downloaded the open source Sahana disaster management system, installed it on their server, and began studying the source code. As featured in this issue of the OSBR, Sahana (http://sahana.lk) was developed in Sri Lanka by a group of volunteer programmers in the immediate aftermath of the 2004/5 Asian tsunami. Over the next several months, the Trinity group designed and implemented a Volun-

teer Management module that was incorporated into the Sahana code base in December 2006. This effort also gave birth to the Humanitarian FOSS project itself.

The Humanitarian FOSS project continues to work closely with the Sahana project, and our students have made a number of contributions to the Sahana effort over the years. In 2008, Humanitarian FOSS students worked with developers in Sri Lanka and with users and developers in China to help deploy Sahana in Chengdu, China following the devastating earthquake there. Students have also worked with Sahana team members in a number of other disaster management efforts organized by the Naval Postgraduate School at Camp Roberts in California.

Students in software engineering courses have worked on Sahana issues as part of their class assignments. Recently, students in introductory courses have helped test and report bugs for a new Python-based version of Sahana in conjunction with its redeployment following the Haiti earthquake in early 2010. Humanitarian FOSS students are currently working on efforts to develop Android-based software tools that interface with Sahana.

Students who have engaged with Sahana in these ways have gained experience that is not normally available through the traditional undergraduate computer science curriculum. In addition to learning how to manage and use the tools and techniques of an open source development environment, such as Eclipse, Sourceforge, SVN, and Mercurial, Humanitarian FOSS students have also learned how large-scale, distributed FOSS development projects are organized.

These students have learned how to collaborate with programmers and developers in the Sahana community, most of whom are based in Sri Lanka. They have learned the importance of well-designed and well-documented code by dealing with complex software systems that have been written by others. And they have seen first hand that what matters in a FOSS meritocracy is whether or not your code solves problems that the development community considers important. So far two Humanitarian FOSS students have earned committer status in the Sahana project, thus becoming full-fledged members of the Sahana project team.

Sahana is just one example of the many global FOSS projects in which Humanitarian FOSS students have participated. Others include OpenMRS (http://openmrs.org), an electronic medical record system developed by Partners in Health and the Regenstrief Institute at Indiana University that is finding growing use in Rwanda and other African nations; the GNOME Accessibility project (http://projects.gnome.org/accessibility/), an effort to make the GNOME desktop accessible to persons with hearing, sight, or physical limitations that prevent them from being able to use a computer; and the Tor project (http://torproject.org), whose software is used to help protect the identities of human rights activists.

Source code from these projects has been studied and used in courses, independent studies, capstone projects, and summer research internships. These computer science students have learned about the FOSS movement and the FOSS development and distribution model.

Reactions from students have been overwhelmingly enthusiastic and positive. A typical sentiment expressed in course evaluations and questionnaires is: "After taking this independent study I realized that I can be in the lab, doing what I am interested in, and still make a humanitarian impact and help society."

**The Ronald McDonald Project**

In addition to participating in global FOSS projects, Humanitarian FOSS students have parti-

cipated in a number projects situated within their local or regional communities. For example, students at Trinity College in Hartford are currently developing a kiosk system for the public computing lab at the Hartford Public Library. In 2008, students at Bowdoin College developed a volunteer scheduling system for the Ronald McDonald House in Portland, Maine

Inspired by the success of the Humanitarian FOSS project with Sahana, an instructor and a group of four computer science students at Bowdoin College decided to try to replicate that experience by applying the same principles to the needs of a local nonprofit organization. During a one-semester software project course, they developed a software artifact called "RMH Homebase" (http://sourceforge.net/projects/rmhhomebase) for the Ronald McDonald House in Portland, Maine. This contribution allows the House manager and volunteers to use an online system to recruit and schedule volunteers who perform various duties at the House on a daily basis.

The RMH Homebase system was completed and installed at the House in May 2008 after a three-month development period, has been updated and improved several times, and is still in productive use today. Moreover, other Ronald McDonald Houses have inquired about adapting RMH Homebase to help with their scheduling needs. This is entirely possible since the software is open source and can be freely adapted by other developers to suit other related scheduling needs. Since 2008, RMH Homebase source code has been downloaded over 500 times from its Sourceforge repository.

When humanitarian FOSS is developed in this way, everyone wins. Students gain a course credit by making a meaningful contribution to open source software, a local non-profit gains a valuable software artifact, and the computer science program adds a socially-relevant dimension to its curriculum that can arguably make it an attractive to a wider and more diverse range of students.

**Incubating FOSS: The Collabbit Project**

Collabbit (http://collabbit.org) is an open source web-based application that aims to increase emergency management efficiency through distributed asynchronous information sharing. The software is targeted to serve the needs of loosely coupled non-profit disaster relief agencies that coordinate responses to disasters. Disaster relief agencies create a common operating picture of an emergency incident through remotely posted incident updates. Individual users subscribe to topics of interest and receive near-instantaneous updates on those topics. Where information is lacking, users may access a topically organized contact registry.

What is particularly interesting about Collabbit is that it provides the first example of the Humanitarian FOSS project's role as an incubator for a new FOSS product. The Collabbit project was begun during the 2009 Humanitarian FOSS Summer Institute when a member of the New York City office of Volunteers Active in Disaster (VOAD; http://nvoad.org), familiar with the Humanitarian FOSS Project's involvement in the Sahana effort, requested that a simple collaboration system be developed for a table-top disaster recovery exercise. A prototype was developed in three weeks and used successfully at the exercise, providing a proof-of-concept that such software would be useful to VOAD and similar organizations.

During the remainder of that summer, Humanitarian FOSS students worked closely with users from VOAD and the Red Cross and the Salvation Army to develop a full-fledged collaboration system. Collabbit is currently hosted on an Humanitarian FOSS server has been used by VOAD for similar table-top exercises. During the U.S.

Thanksgiving holiday in 2009, it was used by the Salvation Army to help coordinate the distribution of 10,000 turkey dinners throughout the NYC metropolitan area.

Currently, two of Collabbit's lead developers, seniors at Wesleyan University, are in the process of creating a company to market and support Collabbit. As in many commercial FOSS enterprises, the software will remain licensed under its current LGPL license, and the development project will continue to be supported at http://collabbit.org. Humanitarian FOSS students and others will continue to participate in Collabbit's development community. At the same time, the students together with one of the principal designers from the VOAD community, will form a LLC associated with the collabbit.com domain and will oversee Collabbit's continuing development.

While still in its infancy, Collabbit exhibits a completely unanticipated but welcome side effect of the Humanitarian FOSS project's educational effort. That is, a new project originating as an academic exercise can lead to the development of a mature and on-going open source software entity whose lifetime and impact are far larger than that for which it was originally developed.

**New Initiatives: Chapters and Certificates**

In addition to its efforts to get students engaged in open source projects, the other main goals of the Humanitarian FOSS project are:

- to get faculty and students at other colleges and universities involved in similar activities
- to develop a certificate program to recognize student achievement in open source software development

The Bowdoin activity described above was the first example of the potential for growing the Humanitarian FOSS community at schools outside Trinity College, Wesleyan University, and Connecticut College. During this past summer, the Humanitarian FOSS Project provided seed funding to start chapters at several new schools, including Mount Holyoke College in Massachusetts, Bergen Community College in New Jersey, and Oregon State University in Oregon. Our goal is to provide support to three additional colleges and universities during the summer 2011.

**The FOSS Certificate Program**

The FOSS Certificate is a credential by which a student can demonstrate mastery of FOSS concepts and practice (http://cert.hfoss.org). The purpose of the FOSS Certificate is to recognize student achievement in FOSS development. It recognizes students who have achieved a significant level of mastery of FOSS concepts and practices.

The Certificate also helps identify the core software development curricular elements that belong in a modern undergraduate computing education. Thus, the Certificate can also raise awareness and interest in FOSS principles and practice among a wider range of computing faculty, students, and programs nationwide.

Applicants may prepare for FOSS certification by satisfactorily completing one or two college-level courses (or their equivalent) that have significant FOSS curriculum content and by having achieved contributor status in one or more FOSS projects. For the purpose of Certification, contributor status means any active contribution to the code base, documentation, or other part of the software that affects the software's productive use. Certificate applications will be reviewed

by three-person teams consisting of academic computer scientists, FOSS project leaders, and professional software developers.

During the current academic year, the Humanitarian FOSS project will conduct a pilot experiment by making the certificate program available to students affiliated with the Humanitarian FOSS project's participating schools. Following the successful completion of that phase, the program will be made more widely available. We are currently in the process of recruiting certificate reviewers from the academic and FOSS development communities.

In addition to serving as a way to recognize student achievement, we are hoping the FOSS Certificate can also serve to stimulate thinking among academic computing departments about the place of FOSS in the undergraduate curriculum. The FOSS Certificate may also provide an additional credential for computer science graduates seeking employment in the software industry.

**The App Inventor Experiment**

App Inventor (http://appinventor.googlelabs.com/about/) is a new visual programming platform for creating mobile application for Android-based smart phones. During the summer of 2010, the Humanitarian FOSS Project participated in an experiment that is addressing the question: Can App Inventor be a suitable platform for bringing a better awareness of computer science to K-12 students?

The App Inventor experiment was a collaboration among two high school teachers, two novice undergraduate students in computer science, a community outreach leader, and a computer science instructor. The Humanitarian FOSS component of this experiment explored the utility of App Inventor as a tool for engaging high school and early college students in the study of computer science by having them develop socially beneficial applications. This particular focus can be particularly useful in attracting student interest from underrepresented groups – especially females, African Americans, and Hispanics – in computer science as a field of study.

In the fall of 2010, the application called "Work it Off," developed by the two students during this experiment, won a competition designed to promote nutritious food choices and physical activity for children as part of First Lady Michelle Obama's *Let's Move!* initiative (http://tinyurl.com/2nd3nl). In the future, the Humanitarian FOSS Project hopes to initiate more experiments like this one, aiming to extend to the K-12 cohort its effort to get students engaged in learning about and employing FOSS principles to benefit their communities. We believe that the potential for App Inventor to help improve public understanding of computer science is significant.

**Conclusion**

The Humanitarian FOSS Project has grown significantly since its inception in 2006. We are encouraged by the progress that the project has made to engage undergraduates in Humanitarian FOSS developments that are serving the needs of global and local humanitarian organizations.

As valuable byproducts, students gain knowledge of the open source development process, local and global organizations gain valuable software tools. Further, colleges and universities gain new and practical dimensions that help connect their computer science programs to the professional and humanitarian world around them.

We hope that the future will see further development of the Humanitarian FOSS model in undergraduate computer science programs throughout the nation and beyond.

*Ralph Morelli is a professor of computer science at Trinity College in Hartford, Connecticut and Principal Investigator of the Humanitarian FOSS Project. He has a BA in mathematics from the University of Connecticut and a PhD in philosophy and an MS in computer science from the University of Hawaii. He is the author of a textbook on Java programming and articles on artificial intelligence and computer science education.*

*Allen Tucker is the Anne T. and Robert M. Bass Professor Emeritus at Bowdoin College in Brunswick, Maine. He has a BA from Wesleyan University and a PhD from Northwestern University. He is the author of several books and articles on programming languages, software development, natural language processing, and computer science education. He is a Fellow of the Association for Computing Machinery (ACM), an ACM Distinguished Lecturer, an open source software developer, and a member of the Humanitarian FOSS Project's Executive Committee.*

*Trishan R. de Lanerolle is the Project Director for the Humanitarian FOSS Project at Trinity College in Hartford, Connecticut. He has a BS in Computer Science from Trinity College, and a MS in Management of Innovation and Technology from Rensselaer Polytechnic Institute (RPI). He is a founding member of the Sahana Software Foundation and community development committee member. He has published and co-authored papers on topics from Computer Science education to FOSS disaster management applications for several international conferences and journals.*

# TriSano: Collaborating for the Good of Global Public Health

Mike Herrick

> *"Protecting the public against health threats ranging from the recent* Salmonella *outbreak linked to peanut butter, to an influenza pandemic requires epidemiologists be able to track, investigate and respond to diseases across jurisdictional boundaries in real time. We now have the advanced technology that enables us to do that."*
>
> Dr. Robert Rolfs, Director
> Division of Disease Control and Prevention
> Utah Department of Health

Healthcare and public health continue to experience significant changes, driven primarily due to new legislation and economic challenges. These changes mean that those in healthcare and public health are under pressure to respond differently in order to meet the health needs of the population. Additionally, public health is collaborative, by nature. Epidemiologists, informaticians, public health officials, nurses, and doctors fluidly come together to prevent disease and protect the health of the global citizens.

This article explores how collaboration and open source software helps healthcare and public health address their challenges across the globe. It is based on the experiences of the TriSano project from the past three years. It traces the history of the project and the refinement of its business model and product offerings to illustrate how collaboration based on a shared business vision with industry thought leaders leads to sustainable software and communities. Finally, insights are shared from working at the intersection of a humanitarian open source project and the business world.

**Introduction**

The TriSano family of products (http://trisano.org) is a surveillance, case management, and outbreak management application for global public health and healthcare. It allows local, state, federal, and international agencies to identify, investigate, and mitigate communicable and chronic diseases, environmental hazards, and bioterrorism events. TriSano supports secure data exchange with laboratories, clinicians, hospitals, vital records, immunization registries, and health information exchanges, as well as offers sophisticated analysis, visualization, and reporting of contact and case information.

TriSano is a registered trademark of Collaborative Software Initiative (CSI; http://csinitiative.com), a company that collaborates with community members, customers, and partners to develop enterprise software products. TriSano

products are highly configurable and comprehensive, and are available in the following editions:

1. CSI TriSano Enterprise Edition is delivered with a commercial source license and is intended for public health organizations in need of an enterprise-class software solution.

2. TriSano Community Edition is freely downloadable under an open source license (AGPL V3; http://wikipedia.org/wiki/Affero_General_Public_License), which is a good fit for those public health organizations needing fundamental case management and surveillance and have strong IT skills available. TriSano Community Edition is backed by an online community of epidemiologists, informaticians, and developers that are passionate about making a difference in protecting and promoting the health of people in local communities and around the world. The community provides this edition to hundreds of nonprofit organizations and others wanting to deploy flexible and transparent applications designed to meet the public health challenges of the 21st century.

**How did we get here?**

The mission of public health is to protect the health of the population. In order to achieve that mission, public health officials, epidemiologists, informaticians, doctors, and nurses collect and analyze data on diseases and outbreaks, then build and execute programs to prevent them.

Public health data across the United States and the world is often collected on paper forms. The data is entered in various file formats into a local, state, or federal database. In 1999, the US Centers for Disease Control and Prevention initiated the National Electronic Disease Surveillance System (NEDSS; http://cdc.gov/nedss/), a national program that promotes the use of data and information system standards to advance the development of efficient, integrated, and interoperable surveillance systems. The Center for Disease Control also built a message-based reporting application called NEDSS Base System, which was provided free to US states. Additionally, many states, counties, and local health departments built their own versions of the NEDSS Base System and proprietary commercial offerings became available.

Over time, relevant issues became apparent with the development and deployment of software systems by government-driven hierarchies or commercial entities. but by global communities that structure themselves in meritocracies based on contribution, such as:

- how to maintain and sustain these systems
- how to exchange data of different formats
- how to meet the business needs spanning local, state, federal, and international organizations
- how to address the variation of reporting requirements across local, state, federal, and international organizations

The State of Utah, its 29 counties, and its 12 local health departments, after a couple of failed attempts to use these offerings, engaged the CSI utilizing the Collaborative Approach. The Collaborative Approach is based upon the premise that a community of industry experts and users and passionate developers can build great software. The result is the TriSano family of products.

**The Collaborative Approach**

The Collaborative Approach begins with a shared business vision (rather then a shared development vision) by industry thought leaders and ends with sustainable software and communities. CSI partners with thought leaders from like-minded entities who come together within a circle of trust and mutual respect to cre-

ate a shared vision for a solution that addresses an industry problem. CSI leads a collaborative approach that embraces openly sharing expertise across groups and boundaries, resulting in a set of mutually defined expectations. CSI and the collaborating members enter into an agreement to work together, with CSI providing developers. The collaborating members share funding of the development of the code and provide subject matter expertise. The advantages to this approach are many, including:

1. The collaborating members share the cost of code development rather than each funding a separate development to solve the same problem.

2. CSI funds productization and manages the intellectual property in partnership with the Collaborative Software Foundation (CSF), a non-profit 501(c)(3) corporation affiliated with the CSI.

3. The collaborating members define the roadmap and directly influence the features, ensuring the code meets their needs.

4. The collaborating members have immediate access to the code and first access to the product.

5. The product is often released under an open source license by the CSF, allowing the collaborating members all of the flexibility that open source licensing provides.

In fall of 2007, the State of Utah Department of Health and Department of Technology Services, representatives from both county and local health departments, the University of Utah, and CSI aligned behind a new, improved vision for NEDSS, one that would meet the needs of federal, state, and local governments. Two key points are particularly relevant about the Collaborative Approach with respect to the development of Trisano:

**1. The Collaborative Approach results in software.** CSI further advances the Collaborative Approach by forming a core team of industry subject-matter experts as well as CSI developers committed to delivering software that addresses an industry problem. The very nature of the Collaborative Approach fosters rapid innovation, resulting in both shared risk and shared reward. Given the insight-driven, closed-loop process, the software is delivered with the benefits of meeting or exceeding the mutually defined expectations of the thought leaders. Beginning in early 2008, the core team delivered semi-monthly iterations, resulting in the availability of TriSano Community Edition in August 2008. CSI TriSano Enterprise Edition went into production across the State of Utah and its local governments in early 2009.

**2. The Collaborative Approach is a model for sustainable products and communities.** As a result of the Collaborative Approach, CSI delivers software products at a fraction of the cost of traditional methods. The Collaborative Approach allows customers and partners predictability and economies of scale. The community edition, released under an open source license, accepts contributions from the community, reducing the cost of development and support. The enterprise edition, released under an annual subscription, generates revenue. Employing best practices, CSI promotes sustainability by balancing a strong enterprise edition with a robust community edition. Today, the TriSano family of products is enabling epidemiologists and informaticians to meet the goal of public health: protect the health of the citizens of the world.

**Where the Humanitarian and Business Worlds Meet**

The following insights have been gleaned from working at the intersection of a humanitarian open source project and the business world:

# Trisano: Collaborating for Global Public Health

*Mike Herrick*

**1. Public health and open source are truly connatural.** Public health is notoriously underfunded and constantly scrambling for resources. At the same time, the problem space is extremely complex and software solutions are very expensive to build and maintain. The only economic model that is sustainable in a domain like public health is a collaborative one like CSI offers where numerous jurisdictions share resources and collaborate to create and sustain these complex systems. Public health has a lot of mistrust of vendors due to the failure of many large-scale, vendor-led software projects over the years. At the same time, the software talent embedded directly in public health organizations and government IT departments is on average ill-equipped to deal with the level of complexity due to resource shortage and talent gaps. An approach based on open collaboration goes a long way to address these issues.

**2. Use the scientific method with the business model.** The original scalable, recurring revenue part of the business model for TriSano was completely focused on selling annual subscriptions to CSI TriSano Enterprise Edition based on a population-based pricing model (i.e., a model based on value) with a commercial source license. The hypothesis was that there were just two types of customers in the market and offerings were provided for each type: i) TriSano Community Edition, which was free and was targeted at developers and highly technical jurisdictions who had the time and resources to work with the code; and ii) TriSano Enterprise Edition, which used an annual subscription model based on population and was targeted at corporate and government customers looking for an enterprise-scale, production-ready solution.

CSI tested its original hypothesis in the market and concluded that customers wanted more offerings around TriSano. It responded with the following range of offerings:

1. CSI TriSano Enterprise Edition with annual subscriptions

2. CSI TriSano Enterprise Edition with up-front perpetual licensing (to accommodate customers who receive a one-time grant)

3. CSI TriSano Enterprise Edition with population or program-based pricing. (The original offering is still appealing to some customers.)

4. TriSano Community Edition with support and maintenance (for customers who want support and maintenance for TriSano but are committed to running the pure open source version of TriSano)

5. TriSano features à la carte (for customers who want TriSano Community Edition plus one of the TriSano Enterprise Edition modules)

6. TriSano SaaS (Software as a Service), offered for both TriSano Editions

By embracing more offerings, the TriSano business model is more complicated, but CSI has seen more product adoption, community contributions, and interest in sponsoring core teams to extend TriSano and create other products in public health as a result of using the scientific method with the business model.

**3. Real community contributions take a lot of nurturing and time to materialize.** People mean well and will volunteer, but there is a high drop-out rate among contributors. The keys to success with contributors are to ensure that the project documentation is good, give new contributors opportunities for simple tasks to get started, and make contributers feel very welcome. Even when these basics are addressed, people will express interest, start, and then just go away. People have busy lives and they mean well, but it takes a lot of time to contribute to a

project and they may not follow through unless it is part of their job.

In the past year, TriSano has made a lot of progress in this area with Southern Nevada Health District (SNHD) in particular. This has now lead to several other community contributors. The story of how SNHD became contributors to TriSano is an educational one. Jeffrey Kriseman, Public Health Informatics Scientist is the original key contributor to TriSano from SNHD. He started by asking seemingly obscure questions directly of TriSano committers. The team did not really know what to make of it, but answered the questions while gently steering him to the mailing list where others in the community could learn from the questions. Little did the team know, but Jeffrey had been looking at TriSano for months prior to asking the questions. The way the team responded to Jeffrey was key to making him believe in TriSano and become a dedicated contributor. Over time, a very good cooperative relationship developed, SNHD deployed TriSano, and continues to be a great contributor to TriSano and advocate for it to other public health agencies.

**4. Open source does not by itself create product success or skip the procurement process.** Open source alone does not differentiate a product enough to make it a winner. The product needs to compete on its own merits through features and by meeting the business needs of customers. Commercial offerings for the product that address all customer needs discussed above help, but do not obviate the need to compete through the procurement process.

The sales cycle is not shorter for open source products. There are legitimate reasons why governments follow a procurement process, but it is often slow and inefficient for everyone, including companies producing open source products. Procurement agents and buyers still are fairly unsophisticated in their understanding of open source products. This can lead to misunderstandings and delays.

Since open source is so transparent, the bar is much higher for open source products because potential customers can assess a lot more about open source products anonymously than they can from a proprietary vendor who shares minimal publicly verifiable information on their websites and only responds to requests for proposals.

**Conclusion**

In late 2007, public health experts and CSI had a vision to deliver an application that would help epidemiologists and informaticians better protect the citizens of the world. TriSano is succeeding in that goal. CSI supports the further development of the TriSano family of products by facilitating the TriSano community and engaging the power of community-building and open source technologies to solve complex public health and healthcare challenges.

*Mr. Mike Herrick is the Executive Director of the Collaborative Software Foundation, a nonprofit organization dedicated to the management of open source software including TriSano. Until recently, Mike was the Vice President of Products for Collaborative Software Initiative. In that role, he was responsible for CSI Core Teams, product development, product management, and technical support. Mike has a broad background in enterprise software with management and individual contributor experience from software companies (eXcelon, Mentor Graphics, and several failed startups), consulting firms (Andersen Consulting and C-bridge Internet Solutions), and IT development and operations (Liberty Mutual). Mike graduated with a BS in Management Information Systems from the University of Dayton.*

# Upcoming Events

**December 4**

International Open Data Hackathon

**Global, including 11 Canadian cities**

The International Open Data Hackathon is a gathering of citizens in cities around the world to write applications using open public data to show support for and encourage the adoption open data policies by the world's local, regional, and national governments. This event is for all citizens, including developers, designers, librarians, and statisticians.

http://www.opendataday.org/

---

**December 4 and 5**

Random Hacks of Kindness Hackathon

**Toronto, ON**

Random Hacks of Kindness (RHoK) is a community of developers, geeks and tech-savvy do-gooders around the world, working to develop software solutions that respond to the challenges facing humanity today. RHoK is all about using technology to make the world a better place by building a community of innovation. RHoK brings software engineers together with disaster relief experts to identify critical global challenges, and develop software to respond to them. A RHoK Hackathon event brings together the best and the brightest hackers from around the world, who volunteer their time to solve real-world problems.

http://www.rhok.org/events/rhok-2/toronto-canada/

**January 25**

Innovation Night

**Burlington, ON**

From successful entrepreneurs, to knowledgeable professors, to experienced investors, and anyone else passionate about innovation - Innovation Night provides access to the region's thought leaders in start-up strategy, providing invaluable input and support on those critical first steps in transitioning your idea into a business. Innovation Night provides an opportunity to showcase your idea and share your passion with a captive audience.

http://www.innovationnight.ca/

---

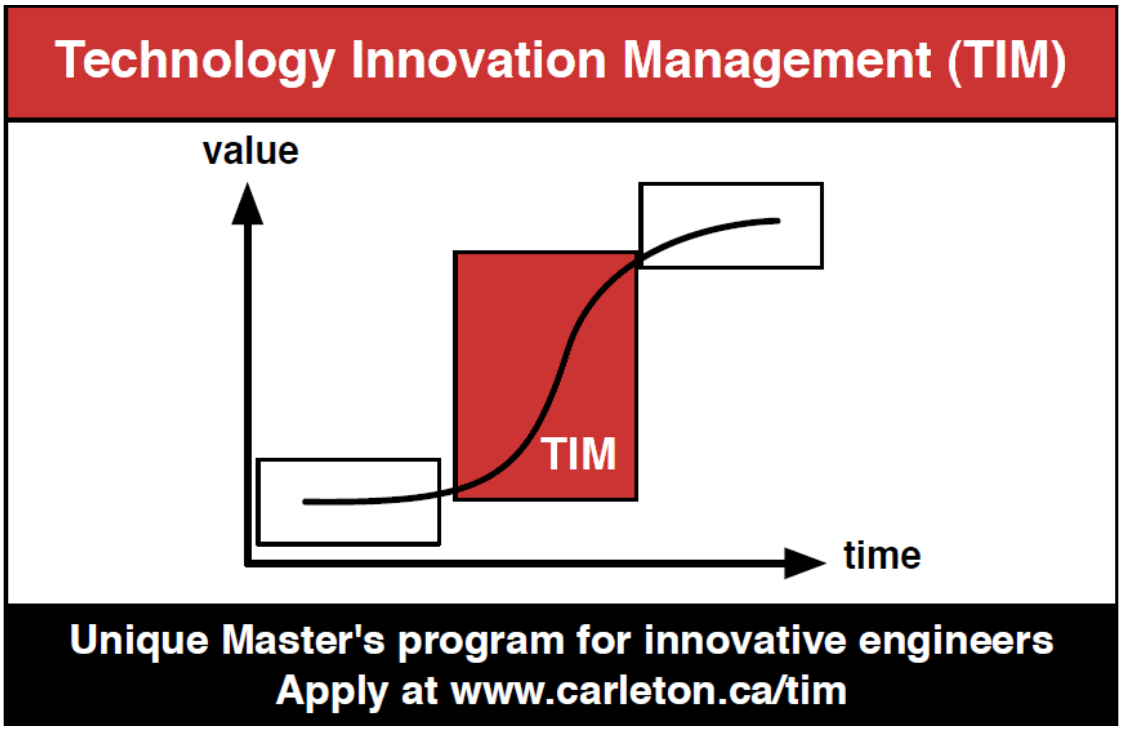
**January 26**

NetGain 5.0

**Toronto, ON**

Is social media a game changer? Social media needs to be monitored, measured and then analysed in order to be actionable business intelligence. To ensure competitive advantage, you need to stay ahead of rapidly evolving trends in research technologies, best practices and business strategies in this growing area. Net Gain 5.0 addresses this need.

http://www.mria-arim.ca/NetGain5/NEWS/default.asp




TIM is a unique Master's program for innovative engineers that focuses on creating wealth at the early stages of company or opportunity life cycles. It is offered by Carleton University's Department of Systems and Computer Engineering. The program provides benefits to aspiring entrepreneurs, engineers seeking more senior leadership roles in their companies, and engineers building credentials and expertise for their next career move.

Carleton UNIVERSITY

# Contribute

The goal of the Open Source Business Resource is to provide quality and insightful content regarding the issues relevant to the development and commercialization of open source assets. We believe the best way to achieve this goal is through the contributions and feedback from experts within the business and open source communities.

OSBR readers are looking for practical ideas they can apply within their own organizations. They also appreciate a thorough exploration of the issues and emerging trends surrounding the business of open source. If you are considering contributing an article, start by asking yourself:

1. Does my research or experience provide any new insights or perspectives?

2. Do I often find myself having to explain this topic when I meet people as they are unaware of its relevance?

3. Do I believe that I could have saved myself time, money, and frustration if someone had explained to me the issues surrounding this topic?

4. Am I constantly correcting misconceptions regarding this topic?

5. Am I considered to be an expert in this field? For example, do I present my research or experience at conferences?

If your answer to any of these questions is "yes," then your topic is probably of interest to OSBR readers.

When writing your article, keep the following points in mind:

1. Thoroughly examine the topic; don't leave the reader wishing for more.

2. Know your central theme and stick to it.

3. Demonstrate your depth of understanding for the topic, and that you have considered its benefits, possible outcomes, and applicability.

4. Write in third-person formal style. Formal first-person style (we only) may also be acceptable.

These guidelines should assist in the process of translating your expertise into a focused article which adds to the knowledgable resources available through the OSBR.

## Upcoming Editorial Themes

**January 2011:** **The Business of Open Source**
Guest Ed.: Michael Weiss
Carleton University

**February 2011:** **Recent Research**

**March 2011:** **Co-creation**
Guest Eds.: Stoyan Tanev, Univ. of Southern Denmark; Marko Seppä, Univ. of Jyväskylä

# Contribute

**Formatting Guidelines**:

All contributions are to be submitted in .txt or .rtf format.

Indicate if your submission has been previously published elsewhere.

Do not send articles shorter than 1500 words or longer than 3000 words.

Begin with a thought-provoking quotation that matches the spirit of the article. Research the source of your quotation in order to provide proper attribution.

Include a 2-3 paragraph abstract that provides the key messages you will be presenting in the article.

Any quotations or references within the article text need attribution. The URL to an online reference is preferred; where no online reference exists, include the name of the person and the full title of the article or book containing the referenced text. If the reference is from a personal communication, ensure that you have permission to use the quote and include a comment to that effect.

Provide a 2-3 paragraph conclusion that summarizes the article's main points and leaves the reader with the most important messages.

If this is your first article, include a 75-150 word biography.

If there are any additional texts that would be of interest to readers, include their full title and location URL.

Include 5 keywords for the article's metadata to assist search engines in finding your article.