

Collaboration the Open Source Way

Nicole C. Engard, MLIS (nengard@gmail.com)
Director of Open Source Education, ByWater Solutions

As you may already know, I am extremely active in the open source community. I recently was given the opportunity to work for not one, but two amazing library open source support companies at the same time. These two companies, ByWater Solutions (<http://bywatersolutions.com>) and BibLibre (<http://biblibre.com>) decided to have me work in collaboration with the two of them simply because that's what open source is all about – collaboration.

When I teach librarians about open source software I make it perfectly clear that you can't separate the code from the community. For those who are unsure of what open source software is, it is basically software that is packaged with its source code so that those who download it are free to view, modify, and redistribute the code for any purpose¹.

That definition doesn't give you the whole picture though. In this column I'd like to explore the heart of open source – the community and the collaboration of members across the world.

A global community

When I talk to librarians about open source, I go through many different definitions, but I always make it clear that you should look at the size of the community behind the software before making the switch. If there is an active community behind the open source application then you can expect constant improvement and excellent support, just look at Firefox for an example of a successful open source product.

Since open source software is most often developed over the Internet in the public eye, learning to collaborate with others is essential to keep the project alive. Reading

this you may think, well that's easy enough, we work in groups every day, but this type of collaboration is very different from what many of us are used to in our daily work lives.

Many open source projects include community members from all over the globe, speaking many different languages, using the software for varied purposes and they all have to work together or the product they produce will end up being a hodge podge of incomprehensible code. This is why for every successful open source project there are at least two that end up being abandoned.

Working together

One way that open source communities get around the barriers of language and culture is to elect someone to help facilitate decision making. Every community has at least one person who is either the creator of the project or has been elected to approve the code that is submitted to the application. In the community I am most active, the one that surrounds the Koha ILS (<http://koha.org>), this person is called the Release Manager.

The Release Manager reviews all code that is submitted to the project² and decides if that code is necessary, if it improves the project, and if enough people would want to use it. In a well-run community, it isn't often that a patch is ignored, because the developer communicated with everyone else about his/her desires and received feedback and suggestions while writing his/her addition.

I have had the pleasure of working in several different types of libraries, but one thing was always the same: meetings never ended when they were supposed to and when they did, we always left with more questions

than when we entered. In open source, the meetings are held virtually and while they tend to get off track (as any meeting does), they are attended by a group of people who all want to be there. Those who participate in open source communities know that if they don't collaborate with others the application will fail. So meetings usually end with a decision being made.

Building together

It may surprise some to find out that open source software, while a new hot topic in libraries, is not something new; in fact, in the beginning all software was open source. One of the most well known success stories in the open source world is the Linux operating system. Linux was started by Linus Torvalds because the operating system that his first computer came with didn't meet his needs. This is a common story among open source projects; they were started to scratch an itch, basically to fix an annoyance. When Linus had the application to where he needed it he sent out a message to his colleagues (by colleague I mean other developers) and offered to share the code if anyone was interested – and boy were they!

Linux has taken off and runs computers worldwide, all because developers worldwide have learned to collaborate together on one of the most complicated types of software – an operating system.

So the question becomes – how do they do it? How do they collaborate over the Internet when they don't speak the same language or share the same time zone? The answer is simple: the developers love their products. You may think that seems rather naïve of me, but I stand by it.

In many cases you have a community of people who are writing code for a project because they want to improve the product for themselves. As a web developer I can tell that the feeling you get when you see that a few lines of code that you wrote make a difference in the way someone completes a simple task is amazing. This feeling is part

of what drives developers in open source communities to work together, to improve the product.

Don't get me wrong, open source projects all have their conflicts; members of the community don't always agree or get along, but behind the frustration and arguments is love. The only reason community members fight with each other is because they all want what's best for the product in the end.

Bringing it into the library

So how then can we learn from the open source world? What should we take away from what we see in the open source world? I think there are two things to take away that will improve collaboration in and between our libraries. The first is empowerment.

I mentioned before the feeling you get when you write a snippet of code that makes a product better. That feeling is power. With open source the power is put into the hands of the community and that makes all the difference. We have to remember in our libraries to give our staff a chance to have a little bit of freedom and along with that comes power.

I am not suggesting we throw out the hierarchy and allow everyone to do whatever they want; I'm suggesting giving everyone a chance to feel like a project is theirs, like the project will live or die by their hands. Give this feeling to more than one project member and you suddenly have a group of people who all have a reason to work together and create something awesome.

Along with that feeling of power will come the second thing I think we can take away from the open source world – love of the project. I have worked on so many projects in my day where either I or someone around me didn't care what the outcome was. This apathy for the project is what leads to a lack of collaboration. If the project is only important to one person, that one person is going to put his/her whole heart into it and the

others are going to do the bare minimum – making the project less of a collaboration and more of a solo project.

Participating

If you'd like a chance to experience the sense of community and collaboration

around an open source application, just pick one and jump right in. If you love using Firefox, check out the community around it and see how you can help. If you want to support open source in the library world there are tons of products out there that will welcome you with open arms.