

Pushing the Boundaries of Innovation through Community Source

April 25, 2007

Charles Severance (csev@umich.edu)
Executive Director
Sakai Foundation

Introduction

While Community Source seems to be a recent phenomenon, in some ways it is truly the natural way for technology to be developed in Higher Education. This is particularly true during periods of particularly during periods of innovation. Higher Education is rather unique in that Higher Education is both a producer and consumer of technology solutions and innovations. Usually Higher Education does not meet long-term technology needs of broad markets - these are best achieved by commercial entities that make long-term commitments based on the profit motive. However at points of important innovation, you will often see Higher Education involved in creating the new definitions of the future. We have yet to see a situation where a community sourced end-user product dominates its market in the long term - but these efforts have remarkable staying power as they often meet the needs of the community of adopters in unique ways that commercial products find difficult to completely match and exceed.

In many cases the actual technology developed by Higher Education may be less important than the resultant change to the marketplace caused by the Innovation sparked by Higher Education. I take a look at several historical examples and then look at the current landscape in Community Source and Higher Education. Often in higher education, with our inherent orientation towards research, we see technological challenges that might seem mundane as interesting. When higher education finds a challenge interesting, we can rally far more resources to attack problems that those outside the academy campaign. .

But even in the less technically interesting cases the challenging problem to solve - may actually be a problem in the market place. Perhaps the problem is that a market is taking Higher Education for granted or not producing the right solution for Higher Education at the right price.

When there is a combination of these factors, Higher Education can bring great resources to bear on a problem of interest. The Sakai Project[8] was formed in just such a "perfect storm" and as such made very rapid progress in a few short years.

In the rest of this paper, I examine some historical context, explore how Sakai was formed, how Sakai operates, places Sakai in the context of other Community Source efforts, and then look towards the future.

A Brief Look Back

In the 1950's there was little vendor software or hardware - as universities made their own hardware - they also made their own software. As one university would produce a hardware design - other universities would copy and improve/alter the design and as such end up with "compatible" hardware - thus allowing for sharing of software - often this was done by sending paper tapes through the mail. A higher education community was forming around the hardware and software. This community was a natural blend of the research elements exploring new technology and end users using the new technology for new exciting applications.

As hardware issues became better understood, commercial hardware vendors such as IBM were able to produce far better hardware than faculty and graduate students with soldering irons - but the interest in building communities around software continued.

Early Example: Michigan Terminal System (MTS)

In the 1960's and 1970's it was quite common for a college campus to have their own local custom operating system. The vendor operating systems simply were not innovating rapidly enough to meet new designs. At the University of Michigan we used an operating system called the Michigan Terminal System [6]. The very name of the operating system hints at the "interactive" use case that was not being met by the vendor operating systems of the time. MTS ran on IBM hardware that was predominantly used for batch data processing in the commercial sector. In Higher Education often an important focus of applied technology is on people and their interactions.

At the height of the MTS effort there were about 50 developers spread across nine adopting institutions including the University of Michigan, University of British Columbia, and others. The developers of MTS primarily were members of their campus IT organizations and working on MTS was simply seen as meeting the campus user requirements for an interactive operating system.

The most popular program on MTS was a collaborative system called Confer - an interactive conferencing system designed and built by Robert Parnes. This system was used both to help the MTS developers communicate around the world well before the Internet was generally available. In addition to enabling developer communities, Confer and MTS allowed many communities to form around various research and social interests.

Ultimately projects like MTS simply could not keep up with the vendor solutions and the universities involved could no longer justify the large number of staff required to "go their own way". The MTS community was relatively small (nine

universities) and as vendor solutions improved - there was no way for new universities to justify selecting MTS and expand the community.

Ultimately as the use cases for interactive operating systems were understood and provided by vendors at a reasonable price - the money that a University spent on maintaining their "own" operating system could be better spent elsewhere. However the last MTS system was retired at Rensselaer Polytechnic University in 1999. The project and community of developers supporting each other lasted 30 years.

Recent Example: JA-Sig and uPortal

Sometimes the issue is not a lack of a capability but simply a market place that is not well suited to the needs of Higher Education. The JA-Sig project was formed in just such a situation. During 2001-2003, portals were all the rage and every organization needed a portal. Some commercial vendors like Plumtree came out with highly extensible portals that were marketed to high-end commercial customers at very high prices. These commercial-oriented portals were not well suited for Higher Education deployment - neither the features nor the price were attractive. In this environment Dr. Carl Jacobsen of the University of Delaware was give Andrew W. Mellon Foundation[5] funding to develop a free and open source portal focused on the use cases of higher education. The foundation funding allowed the hiring of professional project management and a small core of professional developers to "jump-start" the community and provide a single point of contact for three years. This was enough to form a volunteer community around the project for its long-term sustainability. The uPortal project was very successful and was used to deploy nearly 1000 campus portals over the life of the project.

uPortal is now six years old and the landscape and market place has changed very much for the better. uPortal continues to be successful with 200-300 people attending their twice-yearly conferences and the JA-Sig board is building a non-profit foundation for the financial sustainability of uPortal.

The marketplace has produced several important standards during this time period: (a) JSR-168 is a Java API for portlets and (b) WRSP (Web Services for Remote Portals) is a protocol to allow a portlet to be hosted remotely and integrated into a portal. The market is rich with open source portals and commercial portals - uPortal remains the only widely used open source portal aimed squarely at the particular needs of higher education. The long-term sustainability of uPortal is centered on the institutions that have adopted the software helping one another and sharing improvements and bug fixes.

Forming Sakai: The Perfect Storm

As Sakai was being considered and being formed there were a number of factors

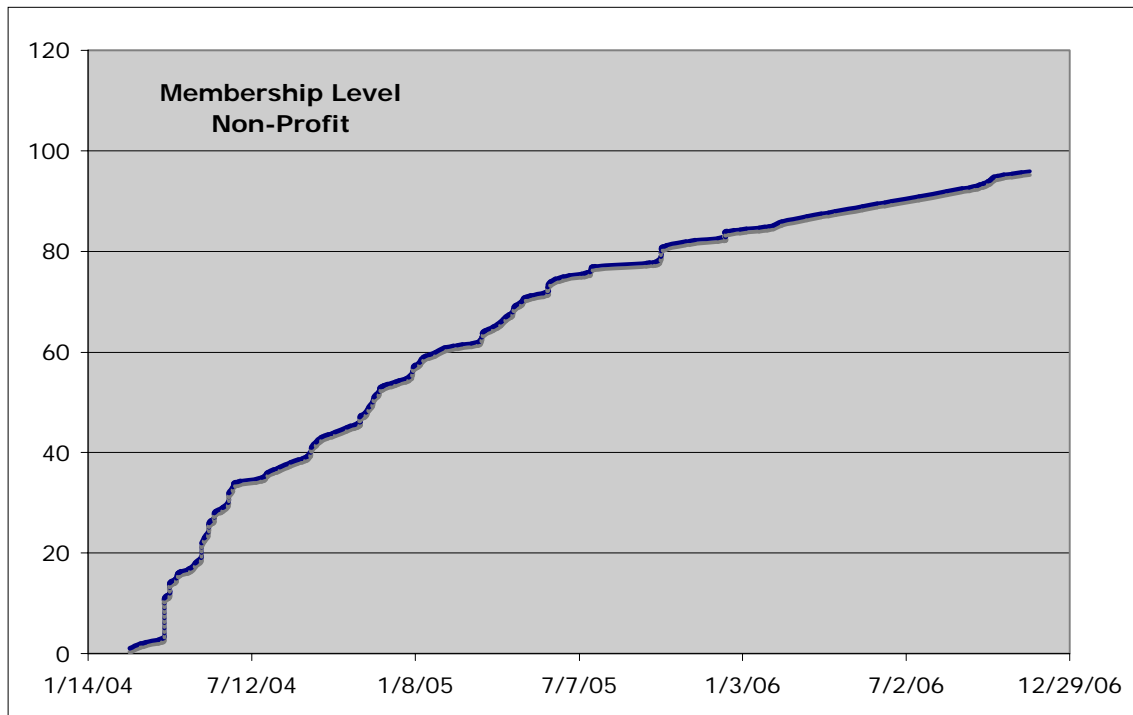
that helped bring significant resources to bear in Sakai and to rapidly grow the Sakai community. The Learning Management System market was divided into two basic camps: (a) school that used commercial solutions and (b) schools that wrote and maintained their own local learning management systems. While the market seemed simple on the surface, there were a number of activities underway:

- A number of open source enterprise learning collaborative learning environments were being explored around the world. Schools were experimenting with Stanford's CourseWork system and others were experimenting with University of Michigan's CHEF system. These systems had been developed by a single school and as such were not well-suited to general-purpose deployment. These systems had patchy functionality and lacked the ability to scale to very large user populations.
- The Mellon Funded OKI[7] project led by Massachusetts Institute of Technology had explored the issues around interoperability between learning management systems and had produced a specification for Open System Interface Definitions (OSIDs). Because the OKI process involved a number of higher education institutions, it created a group of schools that had become used to the notion of working together. The OKI project was coming to an end of its funding and had not yet produced an open source LMS.
- The commercial market place for learning management systems was maturing. Blackboard and WebCT both were efforts that started out in higher education that were commercialized and were becoming quite successful in gaining market share. Other efforts that came from higher education such as Prometheus were acquired by the commercial entities and gently shut down.
- The pace of innovation in the commercial products slowed as they gained market share and installed user base. Consistency, reliability and profitability all were more important to the companies than innovation.
- There was significant research interest around the world in virtual research environments that included collaboration capabilities. This provided a significant source of additional funds and talent to the Sakai effort.
- The Mellon-funded uPortal effort was very successful as the first "Community Source" effort lead by Higher Education producing and consuming end-user technology.

All of these activities were "priming" the market for the emergence of a Sakai or Sakai-like activity. The Andrew W. Mellon Foundation saw this market at a

crossroads and funded the formation of the Sakai Project consisting of MIT, Indiana University, Stanford University, and University of Michigan to develop and distribute an open source collaboration and learning environment.

The Hewlett Foundation funded the formation of the Sakai Partners program to help build a base of financial support for the long-term sustainability of Sakai. Each organization was to agree to a three-year commitment for \$10,000 per year. The partner program experienced strong growth from the moment it was announced, validating the broad market interest in having a general purpose open source enterprise level learning management system available and sustained. Interestingly the Sakai partners were evenly split between those schools who were developing their own systems and schools who were using commercial systems. Those organizations that had commercial systems were interested in seeing Sakai as a force for "market fairness".



When the Sakai Project was finished in December 2005, the coordination of the Sakai community was transferred to the Sakai Foundation and the Sakai Partners became the founding members of the Sakai Foundation. With over 100 members, the Sakai Foundation has revenue of a million dollars per year from its members.

During the same time, Sakai had been significantly rewritten from its CHEF roots to be for more general purpose and far more scalable. The Sakai 2.1 release was a very solid release and was running in production for over 50 schools in December 2005.

A strong Sakai Foundation coupled with the strong software base in two short

years shows how rapidly a community can form, organize itself, fund itself, and accomplish a very significant task in a short period of time.

The Community as the Engine of Innovation

We see Community Source as an improvement on the well-understood patterns of open source as practiced by the Apache Foundation. Open Source has been shown to be very successful in developing infrastructure such as web servers and even operating systems. Open Source governance is very developer-centric.

Community Source Governance is exploring two important areas beyond Apache-style developer-centric governance.

- Since applications such as Sakai have user interfaces - the end-users will have opinions, needs, and requirements around the software. These needs must be recorded and communicated across the community.
- Community Source projects often end up with dedicated professional full-time developers that work for institutions that adopt the software. When an institution is forgoing its local priorities and needs and contributing resources to the commons, those institutions will need to see tangible return on investment over time or they will pull their resources back from the commons to focus on local priorities and needs.

The essential value proposition of community source is "sticking together" to share overall risk. It is not critical for all of the adopters to "stick together" - but it is necessary to maintain a critical mass of the community comprised of staff from the organizations that are "sticking together".

The Sakai Foundation is the "glue" that works to keep the Community together. There are a number of important services provided by the Sakai Foundation:

- A requirements process is used to insure that all the voices of the community are heard, prioritized, and retained. Since the Sakai developers are volunteers (i.e. they are not paid by the Foundation) the purpose of the requirements is to inform the developers of the priorities as seen by the community. Interestingly this process seems to work well in the medium-term because developers generally are interested in making their software meet the user needs as much as possible.
- Project Coordination staff that track and report on the activities of the volunteer community. Sakai is made up of many small self-directed projects. Ultimately these small projects must come together to produce a release of the Sakai "bundle". The project coordination effort helps get all

of the "incoming projects" aligned so they can come into the release in an orderly manner. Also with a widely distributed community it is important to enhance communication between teams - particularly when one team is depending on the output of another team.

- Quality Assurance is a very important community-wide service. The Foundation staff coordinates several QA cycles per year with up to 100 volunteers participating in each QA effort. The QA volunteers usually are drawn from the staff of adopting institutions. Over time schools have learned that they need to volunteer and participate in the Foundation's QA process before the software is released to make sure that each release is bug free when it is released. If schools wait to test the Sakai release right before they are placing it in production - they find that there may not be enough time or developer attention to fix issues that arise.
- Community Development - It is important for the community to continue to expand because there will always be some attrition within the community. Community development is a wide-ranging activity and includes putting on Sakai meetings and conferences and working closely with prospective members.

Sakai is already mature enough that there has been some gentle turnover in the membership, staff, and leadership. This turnover is very healthy and makes members of the community far more confident that the Sakai effort is sustainable in the long term and not simply dependent on a few core individuals.

The overall goal of the management of the community is to assure all-important stakeholders that they will be rewarded for their involvement in the community. This is the primary purpose of the Sakai Foundation - making joining the Sakai Community an attractive value proposition for each member.

Evolving Community Source Efforts

Community source is still an evolving concepts - each project can build on the successes of the previous project and learn from the challenges of previous projects and hopefully improve the model as we go along. In a sense there is a community of people who are generally interested in the governance of Community Source and they mutually help and support each other.

A common thread of these projects is funding and/or encouragement from the Andrew W. Mellon Foundation. This funding has been essential because it has helped cover the initial costs of collaboration and also helps bring attention to these efforts.

As I go through this timeline I identify challenges and issues with each project and note how the succeeding projects changed their approaches slightly to

address issues.

- uPortal[2] is seen by many as the first higher-education community source project. It established the value of funded central resources and a focal point around which to develop the community. For uPortal consulting organizations funded by the grant provided the central technical resources. This dependence on a single commercial organization led to some challenges when the grant funding was removed.
- Sakai imitated much of the uPortal approach and community values. Jim Farmer and Carl Jacobsen were brought from the uPortal project into the Sakai project to help insure that the ideas and culture from uPortal were well represented in the Sakai Project. Sakai evolved in several ways from uPortal. Sakai added the notion of a partners program - an effort to build a funding base for post-grant sustainability from the very moment of project inception. Sakai also kept the governance and technical leadership in Higher Education with technical resources contributed to the commons by the member institutions. Sakai encouraged commercial partners involvement and has commercial partner staff as part of the community but technical leadership and governance has been higher education focused. Sakai's primary challenges primarily arose from its rapid pace of expansion. During 2004-2005, Sakai was building software, processes, and community culture all the while adding a new organization per week to the mix. This resulted in much of Sakai governance to be "done on the fly". Because of this rapid pace, it was difficult to centrally control and manage the community so Sakai had to fall back to open source patterns where institutions made their own choices with their own resources - informed by the global requirements and priorities.
- Kualifinancial Services[3] started a year after Sakai and was well informed by the challenges of Sakai. Indiana University took the lead in Kualifinancials and Brad Wheeler who was a founding member of the Sakai Project, brought many of the best Sakai ideas into KFS. KFS evolved beyond Sakai in several important ways: (a) KFS built their community more slowly - they met with each new potential member and made sure that they would be a good cultural fit for the community, (b) each new member was to make a more significant commitment to the commons, (c) KFS is driven by a functional council which makes software priority decisions across the effort, and (d) the KFS project has strong central project management which manages the formally seconded resources. The KFS effort is a very exciting development in Community Source governance because it is more predictable with a slower pace when compared with Sakai. Like Sakai the Kualifinancial Foundation was created as the grant funds ran out to insure the long-term availability of central financial resources for the Kualifinancial effort.

- Kuali Research Administration[4] is a new effort that has adopted the Kuali Financials governance patterns. The interesting effect of KRA will be to see how the Kuali model evolves with a slightly different set of stakeholders. In the financial area - the requirements are very well understood and most schools have a similar approach to debits and credits. In Research Administration many schools have developed their own highly localized and customized software solutions. So it will be more challenging to understand the scope and requirements of a general purpose KRA system deployable across many universities. The complexity of the KRA requirements space of KRA is quite similar to the complexity of the Sakai requirements space. It will be interesting to see if KRA finds a need to alter the KFS governance model slightly to meet the needs of the community stakeholders.
- Kuali Student is being led by the University of British Columbia and is adding the notion of Service Oriented Architecture at its core. The idea is to allow many components to be independently built and put together. Kuali student is building a solid commitment from its founding members and is taking a relatively long view of the effort. The first years of Kuali Student will be general-purpose design and modeling efforts to insure that the architecture and scope of KS is appropriate in the long term. KS is an example of a very highly evolved Community Source effort - the founding partners are so confident of the success of the effort that they are willing to commit financial and staff resources to support a multi-year planning process.
- The Fedora [1] project is collaboration between Cornell University Information Science and the University of Virginia Library and has been supported by the Andrew F. Mellon Foundation and National Science Foundation for many years and has resulted in a mature well-adopted product. Fedora is in the process of creating the Fedora Commons that will act to collect resources for the long-term sustainability of Fedora beyond its grant-funded efforts. The Fedora Commons is a new wrinkle in Community Source because Fedora is a very mature community with well-established leadership and a well-established community culture and a very mature and well-adopted product that has very strong market presence. While it is an oversimplification, Fedora is effectively past many of the challenges of other younger Community Source projects and is truly focused on the long-term sustainability of a mature product. While the Fedora Commons is still in development, it will likely be a good indicator of what Community Source will look like in steady state.

A clear pattern in Community Source is rapidly evolving governance models where each project is acutely aware of the challenges of the projects that came before and adjusts their model to improve governance. This has resulted in a very rapid exploration of the Community Source Governance space. It will still

be some time before we understand the ideal governance model (or models) for community source.

Looking Forward

Thanks to visionary investments but the Andrew W. Mellon Foundation, the Hewlett Foundation, and National Science Foundation, Community Source has experienced rapid growth and is currently seen as a credible way to produce Open Source software.

Each successful project which uses Community Source approaches validates the Community Source model and helps potential investors "take the risk" on Community Source efforts

If we look at the overall amount of money and staff being invested in providing information technology solutions in Higher Education across the board - there is no question that we have sufficient resources to write nearly all of the software needed by Higher Education using Community Source approaches. There are two basic questions that must be answered before we understand how far Community Source will drive forward:

- The cost of coordinating is significant. In the short-term costs of those institutions that take leadership roles in Community source increase rather than decrease. It is important that schools that take leadership in community source maintain their commitment until the community evolves to be self-sustaining at which point leadership of the effort can move between multiple organizations.
- It is challenging to build general-purpose software that is flexible enough to meet the needs of many different adopting institutions. Even when an institution adopts a large commercial product, there is often a significant customization cost. In Community Source - some of this customization cost is moved to earlier in the process as the software is designed and developed. Community members must understand that the up-front coordination effort is ultimately saving them on customization costs. So institutions must be patient to get the features they need and often they must invest in the needs of others before they can satisfy their own needs.

The essence of a community is that we all make some compromises so that we can all enjoy the benefits of the commons.

References

[1] - Digital Content Repository - www.fedora.info

[2] www.ja-sig.org

[3] - Kualifinancial System - <http://kuali.org/communities/kfs/>

[4] - Kuali Research Administration - <http://kuali.org/communities/kra/>

[5] - Research in Information Technology -
http://www.mellon.org/grant_programs/programs/copy_of_research

[6] Michigan Terminal System - <http://www.clock.org/~jss/work/mts/30years.html>

[7] - The Open Knowledge Initiative www.okipproject.org

[8] www.sakaiproject.org