

THE FUTURE OF SOCIAL NETWORKING

G. CHANDRA SEKHAR,
Assistant Professor,
Institute of Aeronautical Engineering College
Hyderabad, T.S.

Dr. M Naveen Kumar
System Administrator
Telangana University, Nizamabad-503322, T.S.

Introduction

Almost three years ago, I started the Appleseed Project as a way to start experimenting with the idea of distributed social networking. Armed with enough hubris and naivete to think I could build it alone, I coded and coded until I got to the point I am now: With a project that stands at around 75% complete, it functions as a sort of proof-of-concept for distributed social networking in general. I've learned a lot, however, about the technological hurdles and impact that a truly open and distributed social networking platform could provide.

I'm writing this as a way to present my ideas and what I've learned in the hopes that an open and distributed solution becomes the direction that social networking takes. Although I would be happiest to see Appleseed succeed, I'd be excited to see any open and distributed social networking solution be adopted and replace the walled gardens that currently dot the internet landscape.

The Problem with Walled Gardens

The current crop of social networking sites (Myspace, Friendster, Facebook, etc) all operate as "walled gardens", where content is exclusive to the site, and most importantly, the sites provide very little interaction with the outside internet. A user on Myspace has no way to interact with a user on Facebook, despite the design of the internet having historically favored the concept of interaction between internet locations. There is no technological reason for why a user on Myspace wouldn't be able to add a user on Friendster or Facebook to their friends list. At the very least, there is no reason why the same Myspace user couldn't send a message to that same Facebook or Friendster user.

While building Appleseed, I realized just how simple it was to get two "nodes" to communicate. It became a matter of a simple behind-the-scenes request which returned a small XML file. It became apparent to me that the reason that we don't have communication between social networking sites had nothing to do with technological constraints, and was a purely economic decision on the part of existing sites. Almost all major social networking sites business models are centered on having the largest userbase possible, and user lock-in is a major part of that. By restricting your ability to interact with outside sites, they also restrict your ability to choose another site and still maintain your relationships and ability to contact your friends.

The whole situation we're in makes no sense in the context of the way the internet was meant to operate. Concentrating user bases into centralized locations and locking them in seems more like the early 90's and the way CompuServe, AOL, and the like attempted to use walled gardens to monopolize the internet. The natural evolution is an open, distributed standard. A sort of SMTP for social networking, which allows any node to fully connect to any other node. Would people put up with an email address at gmail.com which could only email other gmail.com users? Of course not, and the same standard should be applied to social networking.

Why "glue" doesn't work

One of the solutions proposed is to use aggregation services which act as "glue" to hold together the current systems. This is a fundamentally flawed solution, because it ultimately relies on no single standard for communication between sites. Site A, B, and C might work with the aggregation service, but brand new Site D will require that the aggregation service supports their system. And what happens when sites change their format? The result becomes a constant process of modifying and extending the aggregation service when new sites and formats arise.

Oddly enough, when it comes to making sense of multiple walled gardens, the best we've come up with is to glue them together and hope it holds. But this is hardly a solution, and really shouldn't be advocated by those who want to see social networking properly evolve.

The Distributed Solution

The solution is actually pretty simple. Especially considering things like OpenID already solve the problem of distributed logins. A user has a home node, which provides them with a social networking address, much like an email address.

I'm currently 'michael.chisari@appleseedproject.org'. This also becomes their OpenID identity, and allows them to log in to other Appleseed (and potentially any OpenID site) nodes. Communication between sites exists as a simple http request from one node to another. Friend

requests, messaging, all types of interaction between users becomes easily distributed while functioning seamlessly as far as the end user is concerned.

The result is that instead of having one Myspace or one Friendster, each with millions of users on it, we have thousands of sites which range from a few dozen to hundreds of thousands of users. Except they all exist in a network of nodes which communicate between each other.

The importance of distributed social networking can be outlined in the advantage that the user gains: Freedom to choose between any compatible node without losing access to their contacts.

Privacy and Trust In A Socially Networked World

It's a whole new ball game when you're dealing with a distributed social networking model. No longer provided the protection of a walled garden, it's important to think of the system from the ground up in terms of "Privacy" and "Trust."

Privacy means that the home node of a user should never give out any information that isn't explicitly stated to be publicly available. Because nodes are querying other nodes about information about users, the assumption is always on the side of privacy, lacking any further data otherwise. Although you cannot achieve full privacy in a socially networked world, most social networks today function as unlocked glass houses, providing very few, if any, privacy options for the user. In a world that is increasingly socially networked, where social networks will very likely include one's boss, parents, landlord, and more, it's important to be able to provide strong mechanisms for restricting access. The user should have as much granular control over who in their immediate network can see what as practically possible.

Trust is the concept of utilizing social networks to determine relationships with strangers, much like in the real world. If you sit alone in a public place, and a stranger comes up and begins talking to you, you have very little trust between you and that person. However, if you sit with a good friend, and that good friend introduces you to that same stranger, a trust relationship has been established. Social networking must recognize this basic aspect of human social interaction, and find strong and intuitive methods for implementing it on a software level. This not only can help with the random friend requests that many on MySpace get, but also can help to eliminate spam, and even possible provide for a social network where everyone can participate as safely as real life, using their trust networks to evaluate new relationships.

Technological Hurdles and Considerations

Most importantly, it must be stated that social networking works best when the concerns of the users are considered to be greater than the concerns of the developers. Many open source social networking solutions break too much from the current social networking convention in what may be interesting technologically, but can be off-putting to users who have grown accustomed to sites such as MySpace. Being at the birth of a new paradigm provides a lot of opportunities to experiment technologically, but when this is done without usability kept in mind, or when it solves a problem that only really exists for the developer, it serves as a deterrent to building a large userbase, which is the only real purpose of social networking.

Balancing this consideration with the technological problems that distributed social networking creates can be very difficult. For instance, the issue of spam messages, while easily guarded against from within a walled garden with active administration, becomes a much more difficult task during decentralization. One of the solutions that Appleseed has used is a simplified version of the IM2000 protocol (<http://www.im2000.org/>), which is a sender-stores system, as opposed to the receiver-stores system of SMTP. This provides the most important protection against spam: Accountability. In order to send a message, your node must maintain its existence and identity on the internet until the message is "read" (downloaded). Without going too far into this idea, the balance between this system and the previous concern of usability is that users will be reluctant to adopt this new system unless it feels incredibly similar to a standard email application.

Other hurdles include scaling outward (as opposed to upwards, like current walled gardens do), and how to perform searches in such a distributed environment. Maintaining that distributed system, from a development and API standpoint, is probably the most difficult task. Error checking for downed nodes, allowing for variable speeds in response times, etc., all become huge issues, and unlike a walled garden, a social networking node cannot control anything outside of its own server. Fortunately, these are issues inherent in the distributed nature of the internet, and have been solved in varying degrees plenty of times before. The issue is simply to apply those solutions in the best manner possible.

Conclusion

Eventually, Myspace and Facebook, no matter what fancy features they may add, will seem as archaic as CompuServe and Prodigy do now. The acceptance of a distributed social networking model is, as the internet has shown, an inevitability. All proprietary walled gardens have given way to distributed models, and social networking is the next frontier. And just like open, distributed protocols before it, social networking requires an open API in order to function

properly. Since many walled gardens are based on amassing as many users as possible, in order to maximize ad revenue, adopting a distributed model goes against their business plan. Therefore, it's up to the open source community to come up with a real distributed social networking solution.

References

- [1] Milan Stankovic: Modeling Online Presence. In: Proceedings of the First Social Data on the Web Workshop, Karlsruhe, Germany, October 27, 2008, CEUR Workshop Proceedings, ISSN 1613-0073, online: <http://sunsite.informatik.rwth-aachen.de/Publications/CEURWS/Vol-405/paper9.pdf>
- [2] Brickley, D., Miller, L., 2006. Foaf vocabulary specification. Available at <http://xmlns.com/foaf/spec/>
- [3] Bojars, U. and Breslin, J. SIOC Core Ontology Specification, Available at <http://rdfs.org/sioc/spec/>
- [4] Berners-Lee, T., Hendler, J. and Lassila, O., 2001. The Semantic Web. Scientific American, May 2001: 29-37. See [<http://www.sciam.com/article.cfm?id=the-semanticweb>].
- [5] Carr, N., 2008. The Big Switch: Rewiring the World, from Edison to Google. W. W Norton.
- [6] Ceccaroni, L., Codina, V., Palau, M. and Pous, M., 2009. PaTac: Urban, ubiquitous, personalized services for citizens and tourists. To appear in the 3rd International Conference on Digital Society.