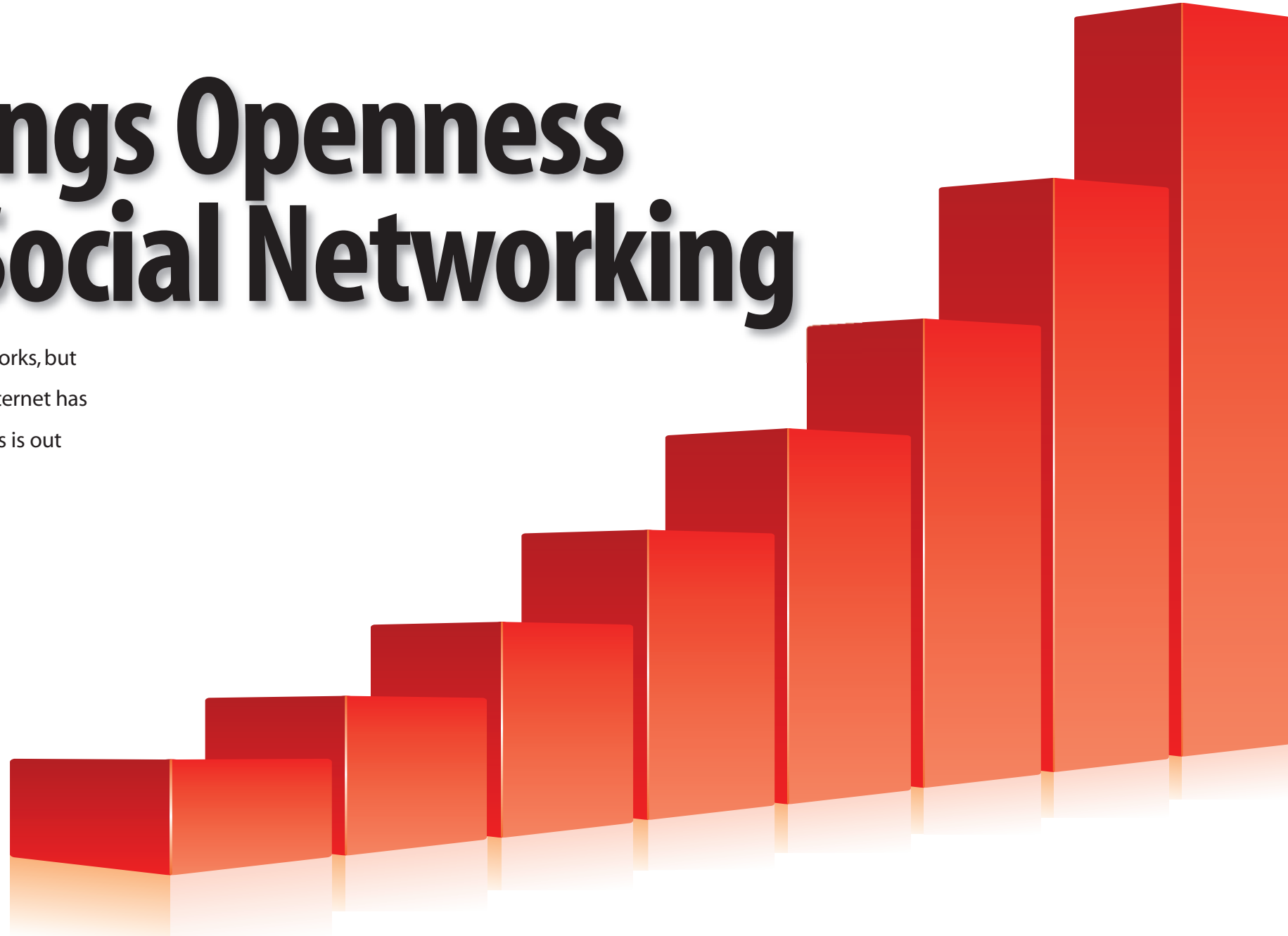


# Musubi Brings Openness to Mobile Social Networking

Facebook may be the 800-pound gorilla of social networks, but it's clinging to a closed service model that the open Internet has destroyed once before. A group of Stanford researchers is out to repeat history.

**By Kurt Marko**



# CONTENTS

TABLE OF

- 3 Author's Bio
- 4 Executive Summary
- 5 Open Social Networks
- 5 Figure 1: Monitoring of Social Networks
- 6 Figure 2: Full-Time Employees Focused on Social Media
- 6 'Egocentric' Social Platform Architecture
- 7 Figure 3: Stanford's Open Social Software Architecture
- 7 A Mobile, Social Demonstration Project
- 8 Figure 4: We Paint Screen Shot
- 9 Figure 5: Organizational Presence on External Social Networking Sites
- 10 More Than a Client. An App Platform
- 10 Figure 6: Official or Unofficial Organizational Presence on External Social Networking Sites
- 11 Caveats and Conclusions
- 12 Social Networking via Email
- 13 Related Reports



## ABOUT US

**InformationWeek Reports'** analysts arm business technology decision-makers with real-world perspective based on qualitative and quantitative research, business and technology assessment and planning tools, and adoption best practices gleaned from experience. To contact us, write to managing director **Art Wittmann** at [awittmann@techweb.com](mailto:awittmann@techweb.com), content director **Lorna Garey** at [lgarey@techweb.com](mailto:lgarey@techweb.com), editor-at-large **Andrew Conry-Murray** at [acmurray@techweb.com](mailto:acmurray@techweb.com), and research managing editor **Heather Vallis** at [hvallis@techweb.com](mailto:hvallis@techweb.com). Find all of our reports at [reports.informationweek.com](http://reports.informationweek.com).

**Kurt Marko***InformationWeek Reports*

**Kurt Marko** is a technology writer and IT industry veteran, now focused on IT analysis and reporting after a varied career that has spanned virtually the entire high-tech food chain, from chips to systems. Upon graduating from Stanford University with a BS and MS in electrical engineering, Kurt spent several years as a semiconductor device physicist, doing process design, modeling and testing. He then joined AT&T Bell Laboratories as a memory chip designer and CAD and simulation developer.

Moving to Hewlett-Packard, Kurt started in its laser printer R&D lab doing electrophotography research, for which he earned a patent, but his love of computers eventually led him to join HP's nascent technical IT group. He spent 15 years as an IT engineer and was a lead architect for several enterprise-wide infrastructure projects at HP, including its Windows domain infrastructure, remote access service, Exchange email infrastructure and managed Web services. For the past five years, Kurt has been a frequent contributor to several IT trade and consumer technology publications and industry conferences. He is now a regular contributor to *InformationWeek* and *Network Computing*.

**Want More?****Never Miss  
a Report!**

Follow



Follow

# SUMMARY

## EXECUTIVE

**The problem with today's social networks**, and the increasing array of collaborative applications built on top of them, is that they're closed, proprietary systems. That's in (un-flattering) contrast to open, standards-based communication and application protocols, where it doesn't matter if I'm on AT&T and you're on Verizon, we can still talk and text. Social networking and collaboration today require that we all use the same cloud-based system. Not only is this a threat to privacy, since Facebook or Twitter has access to everything users upload, it also leads to lock-in since it's impossible to take an archive of my postings, pictures and social connections from one service to another. Our point is that social networks are today's version of AOL and CompuServe, not the Web and SMTP email—a service model that's in conflict with the open, standards-based Internet. Furthermore, even though they've evolved to support mobile devices, today's social services were designed for the era of PC-based Web browsers, not handheld smartphone apps.

If you're thinking, "There must be a better way," you're right. In this report we'll examine some groundbreaking research aimed at bringing openness to social networking on mobile devices. Technology coming out of Stanford's Mobile and Social Computing Lab is creating a new generation of inherently mobile, open, secure and extensible social software platforms and clients.

## Open Social Networks

A paradox with today’s social networks is that while they encourage sharing of users’ most intimate secrets—often with people they’ve never actually met—they themselves are closed, tightly controlled environments. Facebook, Foursquare, Google+, LinkedIn—they all require people to set up separate accounts on a centrally managed service in which every shred of information passes through the operator’s virtual hands. That’s valuable data, and it’s often mined for its commercial value to advertisers pitching everything from pizza to politicians. This proprietary power is even more worrisome now that the most popular social sites, of which Facebook is the poster child, have morphed from being on-line hangouts to application platforms. Every developer seeking to tap into Facebook’s rich reservoir of users must do so on its terms; right now, that means ponying up 30% of gross receipts for the right to use Mr. Zuckerberg’s services and directory. In fact, Facebook’s own IPO filing notes that Zynga, the online game

developer of FarmVille fame, accounted for 12% of its total revenue last year.

Now, no one begrudges Facebook its revenue stream. But without competition, how long before that 30% becomes 40% or 50%? “You don’t mind [a company] charging for a platform,” says Monica Lam, professor of computer science at Stanford University and director of the Stanford Mobile and Social Comput-

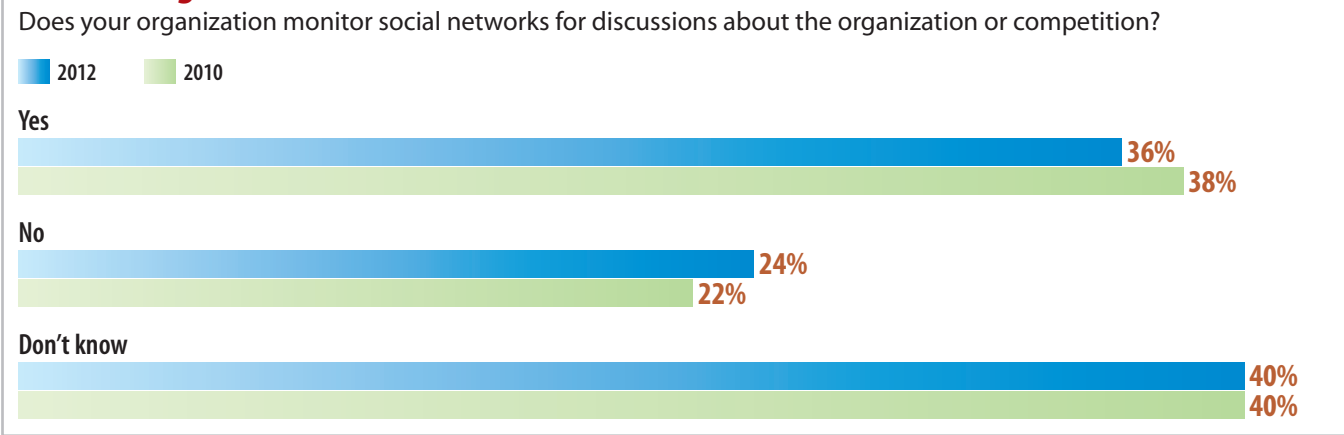
ing Laboratory, “but when it’s a monopoly, that’s a bad thing.”

### Smartphones, the Perfect Social Client?

Professor Lam and her graduate students are out to tear down proprietary walls via the open social software that serves as a key piece of Stanford’s ambitious Programmable Open Mobile Internet 2020 project. Unlike today’s

Figure 1

### Monitoring of Social Networks



Base: 394 respondents in October 2011 and 624 in August 2010 at organizations using one or more internal social networking systems

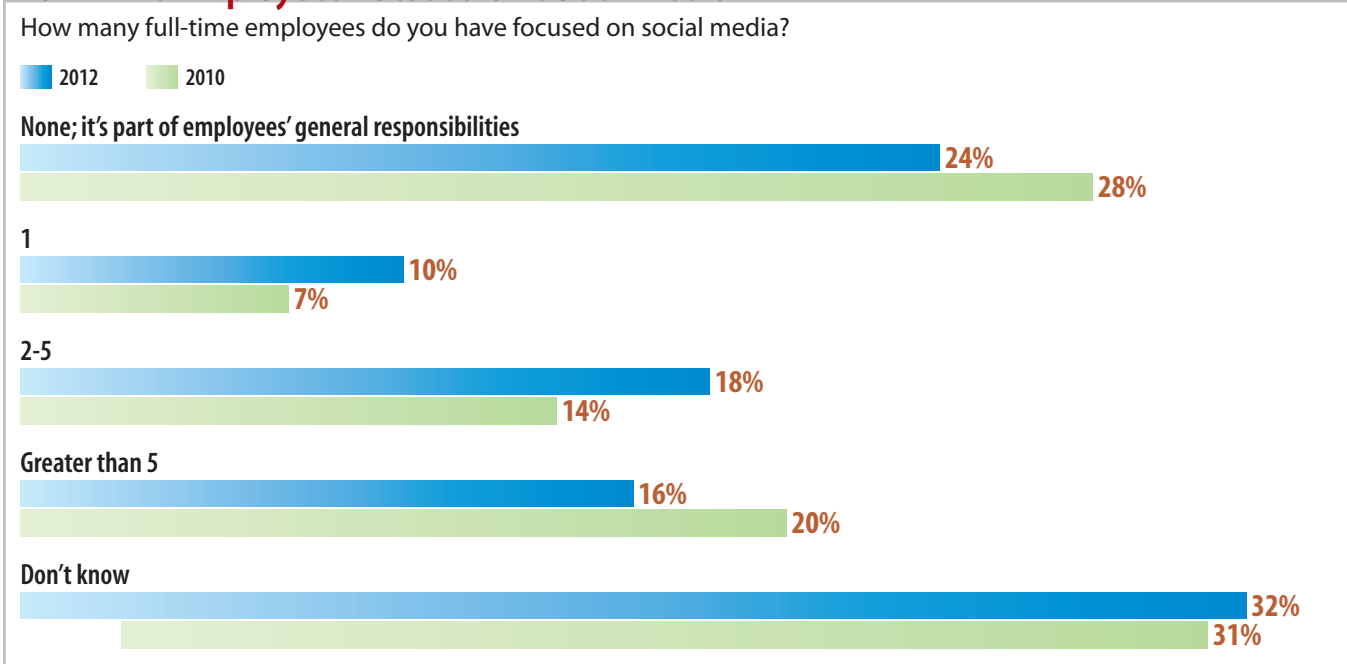
Data: InformationWeek Social Networking in the Enterprise Survey of business technology professionals

R3611111/19

cloud-based social platforms, with their hub-and-spoke architecture channeling all user activity through the provider’s infrastructure, the MobiSocial approach is disintermediated—more like SMS messaging than AIM. In fact, the team’s initial effort, which I described in a [recent column](#), actually used the SMTP infrastructure as its communications fabric ([see story, p. 12](#)). Regrettably, Lam says, email protocols proved not to be as ubiquitous and flexible as the team had hoped, primarily because the most extensible client protocol, IMAP, isn’t universally available. For example, Hotmail (now Windows Live) still doesn’t support it. This called for a change in strategy, but a fortuitous one since it caused the team to shift its focus from PCs to mobile clients, for which email has always been a secondary communication channel.

In watching her students use social networks, Lam noted they were generally on smartphones, not PCs. She came to believe that social networking more closely matches the transitory, succinct, hit-or-miss communication style of phones rather than the more

Figure 2  
Full-Time Employees Focused on Social Media



Base: 394 respondents in October 2011 and 624 in August 2010 at organizations using one or more internal social networking systems  
Data: InformationWeek Social Networking in the Enterprise Survey of business technology professionals  
R3611111/22

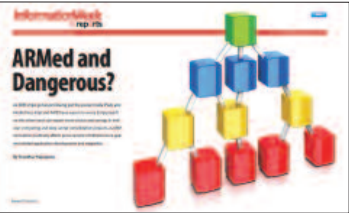
structured, uninterrupted and lengthy style typical of PC-based applications. “Social matches your mobile usage habits,” she says. Thus, she sees the social network being primarily a mobile one.

Enter [Musubi](#), Stanford’s open social com-

munication *and* application platform for phones.

‘Egocentric’ Social Platform Architecture

Strictly speaking, Musubi is just a client demonstration project for the MobiSocial



Fundamentals: ARMed and Dangerous?

As ARM chips go beyond being just the power inside iPads and Kindle Fires, Intel and AMD have reason to worry. Enterprise IT, on the other hand, can expect more choice and savings in end-user computing and data center consolidation projects, as ARM innovation positively affects procurement of infrastructure gear and related application development and migration.

Download

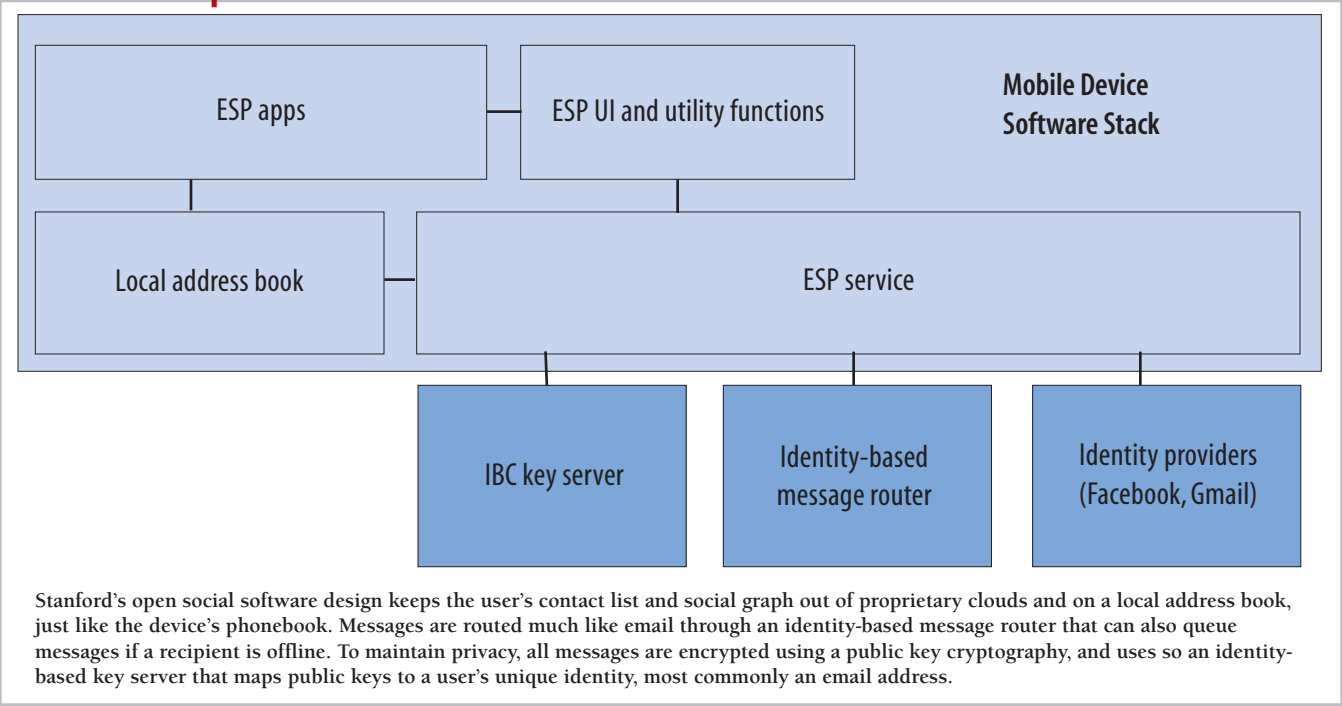
Lab’s “Egocentric Social Platform,” with “ego” being the operative word since ESP is designed to wrest control from the grips of social network operators and return it to social network users, app developers, and even device manufacturers and carriers.

ESP is based on three key principles. First, that users own their “social graphs,” or contact networks, and that graphs reside on their address books, not Facebook’s servers. Second, that data is accessible only on users’ mobile devices, not persistently stored on central servers. Finally, all data is encrypted in transit using public key cryptography; keys are managed through distributed, identity-based encryption, a system that obviates the need for a central certificate-based PKI.

Much like using the phone or SMS, Lam says the basic idea behind ESP is that users should be able to collaborate with anyone in their address books. “We collectively have the social graph in our phone book,” she says. “We only need [tools] to collect it.”

A simple concept, but deceptively complex to implement.

Figure 3  
Stanford’s Open Social Software Architecture



Data: Stanford

Unlike her team’s earlier work using the mature and well-established email system as a communications platform, moving to mobile devices meant developing an entirely new protocol and API. These come together in the MobiSocial lab’s flagship project, Musubi.

A Mobile, Social Demonstration Project

On its face, Musubi is an Android client that will look familiar to anyone who’s used Twitter or iMessage on a smartphone. But Musubi actually targets two audiences: end users, as a vehicle for getting them accli-



mated to the idea of disintermediated social networking, and developers, as a platform for building social applications unencumbered by the restrictions and licensing terms of the big proprietary sites. For the first constituency, Lam says her team resolved to make the out-of-box experience quick and easy. Unlike fledgling efforts, which, well, took a Stanford engineering student to install and configure, [Musubi's first public beta](#), released in mid-March, appeared on Google

**Early users may be underwhelmed by Musubi's functionality, but developers and researchers will find plenty of innovation under the covers.**

Play (the rechristened Android Market) and installs like any other Android app.

Lam says her team has adopted an agile development methodology of releasing early

and often, a strategy driven by a belief that potential users want to actually play with new software, not just read about it. Although the first release is a bit Spartan, it's intuitive and well integrated into both the Android platform and users' existing contact

lists on Gmail and Facebook. That's a good thing, too, since you can collaborate only with other Musubi users. To bootstrap Musubi's user base, the app sends email messages with download instructions to anyone you message who doesn't already have the app installed—which, at this point, is just about everyone in your address book. Lam's goal is to get Musubi into as many hands as possible. "If you have users, all kinds of things can happen," she says.

But "Spartan" doesn't mean "useless." Musubi can already tap into many of Android's native capabilities, so it's easy to insert photos from the gallery or camera, voice notes via the speech recorder, or doodles from one of the Musubi-aware apps into your messages.

While early users may be a bit underwhelmed by Musubi's functionality, developers and researchers will find plenty of innovation under the covers. The foundation of Musubi's ESP design is a new Trusted Group Communication Protocol (TGCP) for peer-to-peer messaging designed for mobile smart-

Figure 4



WePaint is a Musubi-compatible collaborative white board. Here, the red lines might have been drawn by one user and the black by another.

phones that, while always on the network, don't typically accept incoming IP connections.

Key design goals of TGCP are privacy and se-

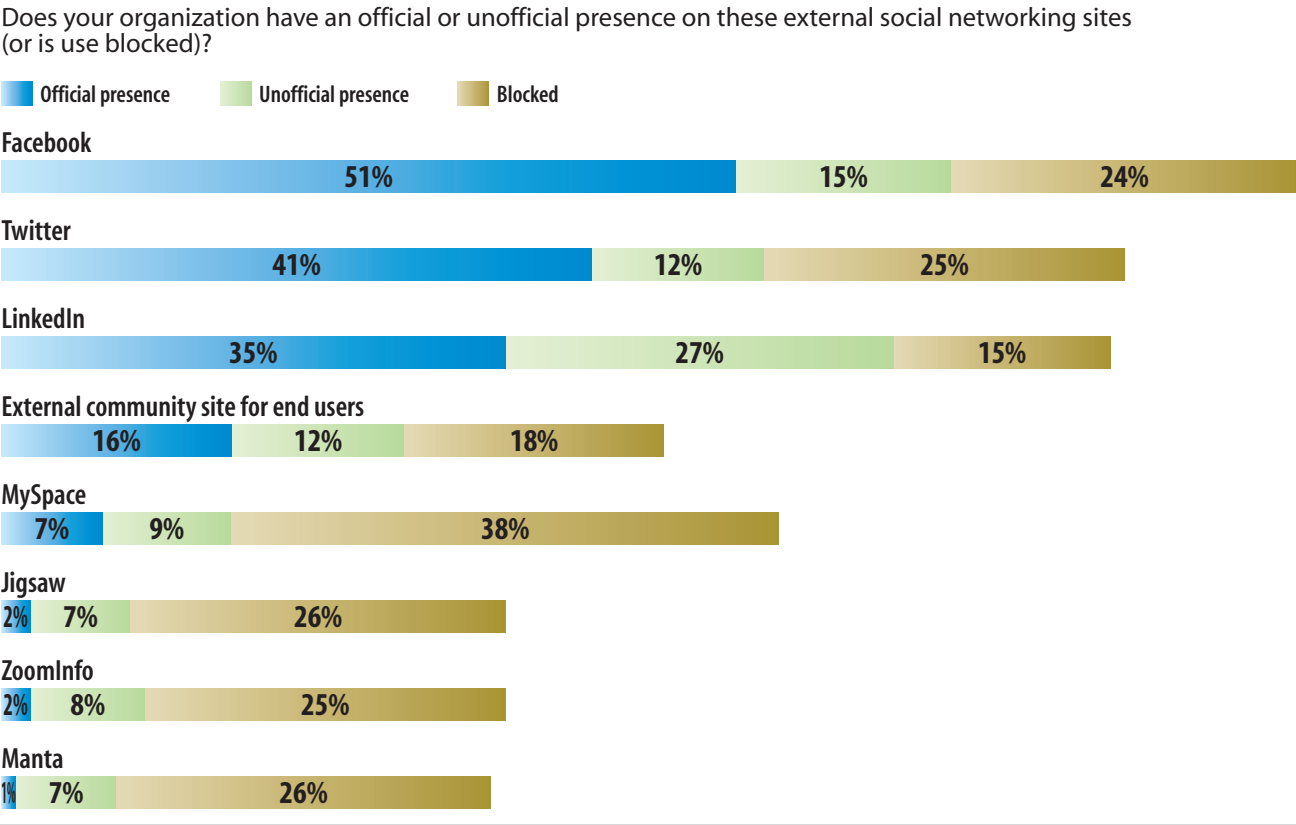


curity. Hence, the protocol encrypts all traffic and routes messages based on identity rather than IP, using the recipient’s public key as the globally unique identifier (with the sender’s private key serving as the other half of the encryption key pair). Bowing to the fact that mobile devices aren’t always on, TGCP also includes an email-like store-and-forward routing mechanism over what are assumed to be semitrusted networks and routing servers, such as those that might be supplied by telecom carriers. Since the messages are encrypted, the routers can’t see any content or even the user’s key; rather, the route server stores an encrypted hash of the key and user identity, masking the user’s true identity from any intermediaries (like the route server operators) while ensuring that only the designated recipient can download and decrypt a queued message.

The ESP architecture also includes the notion of social data feeds, which allow users to share not only status updates but photos, voice messages and indeed any arbitrary data type, such as that from a social application,

Figure 5

Organizational Presence on External Social Networking Sites



Base: 394 respondents at organizations using one or more internal social networking systems  
Data: InformationWeek 2012 Social Networking in the Enterprise Survey 452 of business technology professionals, October 2011

R3611111/15

with any individual or group in their social graphs. Only individuals of a designated group can submit items to the feed, and updates are replicated to every group member. And the feed data structure is extensible, meaning social apps can adapt it to their needs.

### More Than a Client. An App Platform

As Lam points out, writing multiparty social apps for mobile devices is a difficult undertaking. Not only must developers deal with a new API for connecting users and pushing notifications (commonly [Facebook Connect](#)), but they typically need a server to manage connections/notifications and must, of course, write the mobile client code.

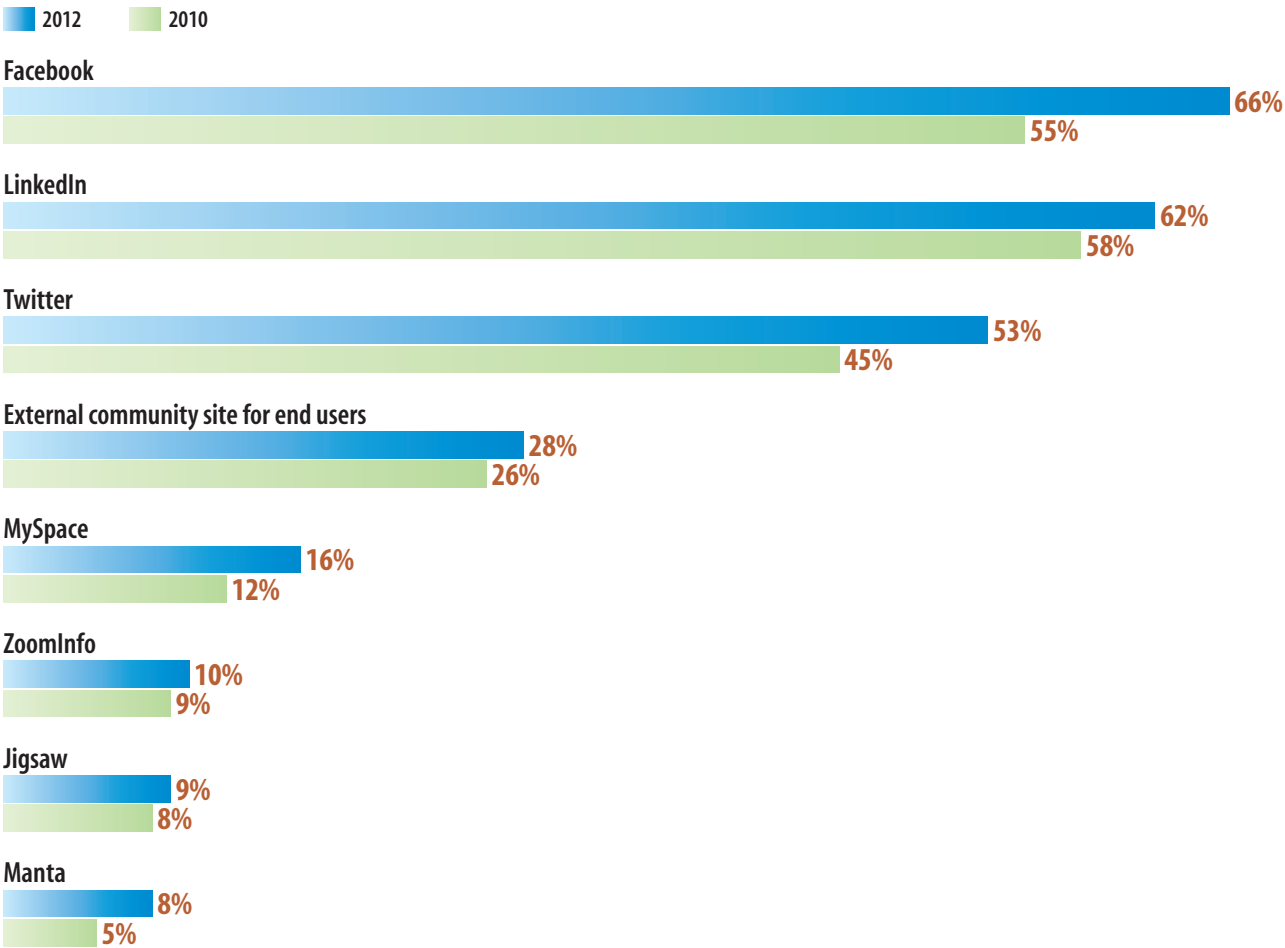
Musubi's SocialKit API streamlines this. Developers are on the hook for the first and last steps, eliminating all the server-side effort. Not only does SocialKit handle the encrypted messaging, identity management and feed distribution, it helps manage distributed state tracking—meaning when Bob adds a letter to his work group's Scrabble game, it's automatically reflected on Mary's, Ted's and Alice's screens.

So far, the Stanford lab has developed several Musubi apps in three broad categories: Interactive sharing apps like TadPoll, a group poll-taker, and TodoBento, a collective to-do list; turn-based apps, in which players take turns moving, like tic-tac-toe, weHold'em

Figure 6

### Official or Unofficial Organizational Presence on External Social Networking Sites

On which of these external social networking sites does your organization have an official or unofficial presence?



Note: Percentages reflect a response of “official presence” or “unofficial presence”  
Base: 394 respondents in October 2011 and 624 in August 2010 at organizations using one or more internal social networking systems  
Data: InformationWeek Social Networking in the Enterprise Survey of business technology professionals

(poker) and WordPlay (Scrabble); and real-time collaboration apps such as WeTube (collaborative music and video playlists) and WePaint (a shared whiteboard).

### Caveats and Conclusions

Much—perhaps even most—of the MobiSocial Lab’s work has been on the plumbing for an open, secure mobile collaboration platform, and here the results, at least to a nondeveloper using and studying the platform, are solid. On the other hand, the Android client, while serviceable, is still clearly more proof of concept than polished product.

But this highlights a notable weakness in Musubi’s story: iPhone support, or rather, the lack thereof.

This is understandable since, unlike Android, the iOS development environment is not amenable to low-level protocol additions and is generally more difficult to extend than Android, but Lam promises that an iPhone port is in the works. While Musubi itself will be a native app, since iOS doesn’t support some of the application messaging features found in

Android, external Musubi-compatible apps like PicSay or Scrabble will need to either be rewritten in HTML5 or accessed via URL-handlers. “It’s not as cool [as a native app], but it’s doable,” she says, adding that HTML5 as an application platform is improving. “You can do quite a bit with it.”

Her hope is that as Musubi catches on, developers will release compelling social apps that virally attract more users, and that the combination of an expanding user and developer community will break down some barriers preventing native ports to other hardware platforms.

For users, the value of Stanford’s MobiSocial effort is that it could lead to an open social platform where users are in complete control of their information and social graphs. For developers, it promises to eliminate the Facebook tax and provide an easier, service-independent way of creating and distributing mobile, social applications without the need for server-side components. And while the lab’s efforts haven’t targeted enterprises, the ESP framework and Musubi-like clients could

enable a future of highly secure, private social networking in which an employee’s personal and work personas can coexist on the same device without worrying about data leakage between the two—or to a nosey third-party social network provider.

Lam says her team’s goal is to combine security and usability with an open development platform that’s fun to use. “We want to create a better ecosystem,” she says, but “it’s the [user] experience that wins people over, not the privacy.”

As the Musubi client matures, its convenient, compelling UI could be just enough of a sweetener to attract the masses, upset the world of closed social software and, in a reprise of the Web’s evisceration of walled gardens like AOL, the dawn of a new generation of open, extensible mobile social software.

### Like This Report?

## Rate It!

Something we could do better? Let us know.

Rate

## MR. PRIVACY

## Social Networking via Email

**T**he first **MobiSocial** project was [Mr. Privacy: Open and Federated Social Networking Using Email](#) (PDF).

Unlike SaaS-based social networking, Mr. Privacy uses email as a federated online identity management system and data transport. As the four Stanford computer scientists working on the project write, “For the sake of adoptability, Mr. Privacy is built upon email, which is itself a mature, scalable, open, and federated infrastructure supporting over 1 billion users. We can socialize with anybody as long as we know his or her email address. We need not sign up to join the same social network. All the shared information is stored as email messages.” The Mr. Privacy APIs essentially turn social applications into email clients, using the same time-tested, reliable protocols, SMTP and IMAP.

To demonstrate the feasibility and evaluate the ramifications of using email-based plumb-

ing for social software, the researchers have developed an API and three Mr. Privacy applications on three platforms: Android, iPhone, and the Firefox browser. It’s the browser app, a [Firefox extension called SocialBar](#), that illustrates the potential for an open Facebook alternative. In a way, SocialBar and its email foundation are quite reminiscent of [Google’s now-defunct Buzz](#), in that they use a Web UI with a specially crafted message and data schema to enable a familiar threaded conversation view. Unlike Buzz, SocialBar isn’t tied to a particular mail service. Also in contrast to all the existing social sites, SocialBar keeps local copies of all messages and a local friend database (that can be integrated with an existing Mozilla Contacts file), giving users greater control over their data, faster response when doing data-intensive actions like message filtering, and the potential for offline access.

Stanford’s researchers rightly conclude

that network effects, i.e., [Metcalfe’s Law](#), create the environment for one or two proprietary companies (read: Facebook and Google) to own the online social ecosystem. Facebook arguably already does and may have reached such critical mass that resistance is futile. However, the most effective way to counteract such natural monopolies isn’t through regulation but openness; namely, a new generation of social software built on open APIs and protocols and using the pervasive, inherently distributed and federated email system. The Stanford team says that because of email’s ubiquity, Mr. Privacy could quickly, almost virally spread, concluding, “We believe that Mr. Privacy is the first proposal that has a chance to challenge the status quo within the next few years. We need one or more killer applications developed using Mr. Privacy to help jump start this model.”

—Kurt Marko

Like This Report?

**Share it!**



Like



Tweet



Share

MORE  
LIKE THIS

## Want More Like This?

**InformationWeek** creates more than 150 reports like this each year, and they're all [free to registered users](#). We'll help you sort through vendor claims, justify IT projects and implement new systems by providing analysis and advice from IT professionals. Right now on our site you'll find:

**Fundamentals: 5 Options for Mobile App Development:** HTML5 and native aren't the only choices when building mobile applications. Hybrid approaches may serve you better. Here's how to decide when to go native, when to go Web and when to split the difference.

**Fundamentals: Mobile App Development:** Smartphones and tablets are everywhere. If you need to reach your customers or employees on these devices, now's the time to start developing mobile apps. We provide essential information to help you get started, including development basics for Apple iOS, Google Android, RIM BlackBerry and Microsoft Windows. We also look at the pros and cons of cross-platform development.

**Rebooting the Antisocial Network:** While corporate social networks starve for attention, users flock to Facebook and other social sites. We share exclusive research and a real-world business case to help you create—or relaunch—a compelling internal site that boosts employee collaboration and productivity by leveraging the habits your users are learning on the Web.

**PLUS:** Find signature reports, such as the *InformationWeek* Salary Survey, *InformationWeek 500* and the annual State of Security report; full issues; and much more.

## Newsletter

Want to stay current on all new *InformationWeek Reports*? Subscribe to our weekly newsletter and never miss a beat.

[Subscribe](#)