Nine IM Accounts and Counting

The key word with instant
messaging today is
interoperability.
Various standards are
in contention.

JOE HILDEBRAND, JABBER

Instant messaging (IM) has become nearly as ubiquitous as e-mail, in some cases—on your teenager's computer, for example—far surpassing e-mail in popularity. But it has gone far beyond teenagers' insular world to business, where it is becoming a useful communication tool.

The problem, unlike e-mail, is that no common standard exists for IM, so users feel compelled to maintain multiple accounts—for example, AOL, Jabber, Yahoo, and MSN.

This decision makes no sense from the end-user perspective, but unfortunately it is an artifact of how IM has developed.

Even without a common IM standard, interoperability is not much of a technical challenge, however. The open source community has demonstrated that since 1999. To interoperate or not to interoperate is actually a business decision. It comes down to giving corporate customers

Nine IM Accounts and Counting

what they want. In some cases that means interoperability and in some cases it means creating a walled or gated community.

Commercial IM providers serve individuals, enterprises, and service providers. Enterprise customers' wants are derived from the ability to secure and control corporate communications, embed presence information in other applications and services, and satisfy the needs of their clients—the company employees.

Service providers' wants are a bit different. They are searching for new revenue opportunities and ways to maintain and build their consumer "communities." They view realtime messaging as being central to their goals of increasing revenues with new services and maintaining their distinct communities (building gaming communities, adding location-based services to mobile users, providing presence-enabling mobile address books, building mobile subscription services). Given that consumer end users haven't had to pay for IM services, it is unlikely that they'd be willing to start unless the IM service had a perceived "added value."

In terms of the consumer IM services, we already know that systems can interoperate with one another, provided you are a registered user in multiple places. If Jabber's open XML protocol for IM, Extensible Messaging and Presence Protocol (XMPP), were adopted as the universal server-to-server gateway between consumer systems, for instance, each system would appear as another XMPP server on the existing, proven network. In this scenario, and under the XMPP naming convention, the AOL Instant Messenger (AIM) handle of example123 could become example123@aol.com and the Yahoo handle of example123 could become example123@yahoo.com.

XMPP can serve as a universal standard that connects each service while being seamless to the end user. Again, the above underscores that this is not a technical challenge but is a business decision.

Some IM companies see risks to their proprietary franchise from open architecture (i.e., their collaboration suites in the case of enterprise vendors, or control of the

customer experience in the case of consumer services). Obviously, some companies believe there is more of a financial reward in keeping their systems closed. Specifically, some companies monetize their consumer service with ads, and some see their community as having intrinsic value. Opening up their systems would require them to change their worldviews.

HISTORY THROUGH TODAY

Interoperability probably became a real issue the day AOL acquired ICQ ("I seek you")—not just because AIM and ICQ were separate and noninteroperable systems, but because it made MSN and Yahoo take notice of IM's value to consumers. By building large user bases of their own, each IM service has divided the IM world into large, noninteroperating domains.

A unified standard didn't emerge immediately because the major consumer services grew very rapidly well before any standards were proposed, and the market was able to run too far ahead of the standards process.

This is partially the result of the standards bodies not having been responsive enough (largely because the main IM vendors had not been involved at all) and partly because common ideas of what is required in an IM protocol took time to mature.

Most of the work on potential IM standards has occurred within the Internet Engineering Task Force (IETF). In response to the growing popularity of IM, the IETF formed the Instant Messaging and Presence Protocol (IMPP) Working Group in 1998. This group was charged with developing a protocol that could serve as a common standard for IM. The IMPP Working Group defined a set of requirements for IM—essentially a menu of items the ultimate protocol must address. These requirements are encapsulated in IETF RFCs 2778 and 2779, which define a minimal feature set for IM but don't address common features such as contact lists, groupchat, or file transfer. (At this point, the IMPP Working Group is no longer the focus for any further protocol development in the IM community.)

Unfortunately, the members of the IMPP group could not reach consensus on a common protocol, leaving the field open to multiple approaches within the IETF. Several proposals were put forward, including some that have fallen out of contention. PRIM (Presence and Instant Messaging) was an early proposal that had a number of proponents but is now obsolete (no work has been done on it since 2001). APEX (Application Exchange) was an XML-based protocol that has been withdrawn from IETF consideration. (Most APEX supporters have since gone over to XMPP, since it too is an XML protocol.)

THE CONTENDERS

Multiple standards are still vying for prominence today. The main contenders are XMPP and SIMPLE, both of which are still under discussion within the IETF.

XMPP is an IETF adaptation of the open Jabber protocol for IM and presence. SIMPLE—SIP for Instant Messaging and Presence Leveraging Extensions—is based on the IETF signaling protocol known as the Session Initiation Protocol, or SIP. SIMPLE is a set of extensions built on top of SIP that will provide for an IM and presence system. Microsoft has thrown its considerable weight behind SIMPLE.

XMPP. Of the two, XMPP is the more complete standard in that its work with the IETF is nearly finished, and it has been field proven over the course of almost five years with nearly 10 million users (actually, many more than that if you consider that the open source community uses XMPP to communicate with MSN, ICQ, AOL, and Yahoo users).

Some of the standards bodies have made the interoperability problem too complex by trying to solve a rigorous, academically complete set of functionality instead of just defining an extensible core that can be added to over time. This approach leads to difficulties in incremental implementation and deployment. Jabber took the extensible approach with XMPP, so we have been able to bite off smaller chunks and solve them independently.

XMPP emerged as a de-facto interoperability standard before it was ever introduced as a proposed industry standard through the IETF. Before IM became a serious business tool, there was not significant pressure on the industry leaders—Microsoft, Yahoo, and AOL—to interoperate. XMPP was created out of developer frustration, well

before businesses began to recognize IM and long before the question of interoperability became a corporate decision. As this situation emerged over the past 18 months, XMPP was the only relatively complete standard.

SIP/SIMPLE. SIMPLE is a protocol that remains largely undefined. It is currently a set of proposed guidelines for how to build software extensions on top of the SIP platform. Until final ratification of these guidelines, some of the proposals have general industry acceptance and some are still open to debate.

SIP was originally designed to negotiate connections between nodes on a network. It excels at enabling one device to tell another device to take an action—that is, ring a phone—and having the receiving device(s) respond with an announcement that it has received the other device's connection request, as well as its availability to handle that request—that is, sending a ringing or busy signal back to the originating phone. As a signaling protocol, SIP is a known quantity and well proven in the ultra-high availability realm of telecom carriers. At its core, SIP

Jabber took the extensible approach with XMPP, so we have been able to bite off smaller chunks and solve them independently.

is meant simply to negotiate a connection and then pass that connection off to a subsequent communications protocol.

SIP however, should not be confused with SIMPLE. SIMPLE is an effort to layer presence and availability information on top of SIP, including IM.

Currently I know of one installation of SIMPLE, at Reuters, which is a beta customer for Microsoft's forthcoming SIMPLE-based IM server (also known as Live Communications Server, or LCS). IBM uses Sametime (renamed Lotus Instant Messaging and Web Conferencing) for its IM infrastructure, said to support SIP and SIMPLE. It is not a SIMPLE implementation, but a proprietary infrastructure that is not natively interoperable with SIMPLE.

Because SIMPLE has not cleared the IETF, the claims of SIMPLE adherence most likely refer to an adherence to the generally accepted proposals, with proprietary extensions to cover areas of the protocol still open to debate.

Nine IM Accounts and Counting

(Peter Ford, Microsoft's chief architect for MSN Messenger, discusses the advantages of SIP/SIMPLE in "A Discussion with Peter Ford," on pp. 18 of this issue.)

Wireless Village. The only exception to the IETF focus of these efforts to settle on a standard has been Wireless Village, an IM protocol for mobile telephony applications that was originally developed by handset manufacturers Ericsson, Motorola, and Nokia. The Wireless Village protocol is fairly complete, but work on it has been somewhat stalled within the Open Mobile Alliance (OMA).

SECURITY IN THE IM WORLD

Traditionally, the major consumer IM services have not paid much attention to security issues. They have not used strong authentication, channel encryption such as Secure Sockets Layer (SSL), or end-to-end encryption between users. Because their protocols are proprietary and often in binary format, and because they have full control over both servers and clients, security breaches in their systems are difficult to detect, let alone guard against or fix. Open protocols, on the other hand, are open to public scrutiny and therefore tend to be much more secure.

IM applications have several potential security holes. One is weak authentication, such as sending passwords in the clear. The consumer IM services all send cleartext passwords over the wire, which is highly insecure since anyone can learn your password by sniffing traffic from your computer. The use of SSL between the user's client and the service makes this practice marginally less insecure, and several of the consumer services are working to add SSL support to their systems.

Another potential source of security holes is the presence of scripts, viruses, worms, or other malicious content in messages received over the network. It is this type of vulnerability that causes so much turmoil in the e-mail world. The problem mainly stems from the inclusion of binary content, which looks like a file but is actually a malicious computer program. Open protocols developed by the IETF for IM generally are not binary, which makes it harder to hide malicious content. XMPP is a single-purpose, well-defined XML protocol that cannot contain random binary content. This helps protect against the inclusion of viruses and other malicious programs. Any of the extended content in XMPP must be well defined by schemas or open protocol definitions, and if a client or server does not understand a given bit of content, it must ignore it rather than try to parse it or run it (which is how e-mail viruses and worms propagate).

In the XMPP/Jabber community, file transfer is a well-controlled process and is subject to the same kind of whitelisting and blacklisting that applies to messages. While nothing can prevent a single user from incorrectly applying the safety mechanisms that are present in XMPP, the presence of such mechanisms makes it more difficult to propagate malicious programs on the network. SIP-based SIMPLE, on the other hand, allows the inclusion of any file type in a message, which opens a potential security breach such as that found in e-mail.

Spam is also a serious problem on the e-mail network, but there are good reasons to believe that it can be contained on a well-designed IM network. There are several causes of spam in the e-mail world. One is that the "from" address of an

Given that immediacy and speed are highly valued, however, expect IM to make serious inroads in places where e-mail, the Web, and voice now reign supreme.

e-mail address is easily spoofed, which is why you may receive messages that appear to be from addresses such as `someone-you-trust@some-company-you-trust.com`, even though the message is not actually from that address.

Another reason e-mail spam is such a problem is that no straightforward mechanisms exist for blocking messages from unwanted addresses. Modern IM protocols such as XMPP have built-in mechanisms that make it very hard to spoof addresses. They also contain sophisticated blocking systems so that you will not receive communications from users who are on your personal “blacklist.” Furthermore, some IM protocols, such as XMPP contains the concept of subscriptions: No one can see your presence on the network unless you specifically approve their request. One criterion for message blocking can be subscription state, so that you will receive messages only from people whose subscription requests have been approved. Mechanisms such as these are built into advanced IM protocols, which were designed after the harsh lessons of e-mail spam were learned.

THE FUTURE OF INTEROPERABILITY

Interoperability will happen. It happened with e-mail once Prodigy, CompuServe, MCIMail, and other proprietary services came under pressure to interoperate, and it will happen with IM as well. The main impetus will come from the companies that are beginning to deploy IM in large numbers. These companies will simply demand interoperability with their partners, suppliers, and customers.

The question is whether some of the major consumer IM services will learn the lesson of e-mail and agree to interoperate as well. Naturally this decision will become easier once there are common protocols that can be used for inter-domain communication. In the next few years some of the business and political opposition to interoperability will melt away. Consumer services may need to find other ways to monetize their investment in IM, perhaps based on quality of service, feature differentiation, and possibly charging for gateways to other messaging services.

Proprietary protocols are unsustainable in the long term. We will see convergence to one or two standards, which are likely to be SIMPLE and XMPP. XMPP is in a strong position simply because it has provided a complete feature set since 1999 and is widely deployed. SIMPLE is still incomplete and does not yet provide a full IM feature set (e.g., it does not do contact lists or groupchat). However, with the backing of such heavyweights as IBM and Microsoft, it is bound to progress and is likely to be

widely accepted. SIMPLE and XMPP will coexist, with gateways between the two protocols, especially on networks with mixed deployments. (We’re seeing this already within several telecommunications service providers.)

Longer term, XMPP will probably become entrenched as a new layer in the network stack, particularly as more and more enterprises and service providers understand the value of “presence” beyond IM and how it can be leveraged to facilitate communications between applications, services, and devices. Already XMPP is being used for content syndication, network management, workflow applications, groupware, gaming, pure machine communications, and more—with no traditional IM users involved. The fact that XMPP is a highly extensible XML streaming technology has led to its use in many innovative applications. This is much more difficult with SIMPLE, since it is not easily extensible. While right now IM is often perceived as mere chat, forward-looking organizations increasingly see the strategic value of an extensible platform for near-realtime communications. Companies that leverage the strategic value of the technology will be more successful, which will in turn lead to even stronger adoption of presence and messaging technologies.

As people begin to apply IM-based technologies in innovative ways, we will see support for IM protocols move farther down the network stack to become a key part of the infrastructure. A technology like XMPP could become an ever-more useful layer of abstraction on top of the physical network, as has happened with protocols like HTTP and TCP.

IM is not going to replace e-mail or the Web. Most new technologies supplement rather than replace older infrastructure (just as TV did not wipe out radio). Given that immediacy and speed are highly valued, however, expect IM to make serious inroads in places where e-mail, the Web, and voice now reign supreme. The relatively spam-free and more secure nature of IM will pull messaging away from e-mail; faster forms processing and notification will lead to many workflow applications over IM infrastructures; and the first full-scale deployments of voice over IP (VoIP) and even video using both SIMPLE and XMPP should emerge in the next few years.

When it comes to IM, chat is just the beginning. Innovative organizations already have this vision. Technology leaders realize what an infrastructure for presence-enabled realtime messaging can make possible, and they are working actively to make that vision become a reality. It’s happening today in fields as diverse as financial services and gaming.

Two features make IM unique: rapid-fire asynchronous

Nine IM Accounts and Counting

messaging, and realtime presence information. We've only just begun exploring what it means to mix these and add them to a wide range of applications and devices. For example, one extension to presence is geographical location information. Once your car is a node on the network, its presence information could be provided (subject to permissions you control) to other nodes on the network, such as your garage door. Why push a button to open your garage door when it can open automatically

Why push a button to open your garage door when it can open automatically whenever your car comes within 20 feet?

whenever your car comes within 20 feet? Sure, that seems like a frivolous use of the technology, but don't think it won't happen just because it's frivolous. Adding presence information (from basic on/off status to extended presence about more sophisticated states) to applications and devices will open up a wealth of uses that we've only just begun to think about. The same is true of asynchronous messaging.

While some industry pundits have bought into Microsoft's contention that the IM game is over and that the direction of IM technology will be based on SIMPLE, millions are actually building innovative applications and deploying large messaging and presence services using XMPP. Why? Because they can deploy today, knowing that XMPP is natively interoperable, extensible, and being chosen by some of the world's largest companies.

With growth will come new challenges. We don't know what all the challenges will be, but when entire companies and supply chains are actively using IM-based technology not just for chat but also for interacting with

applications, things will certainly get interesting. Federated networks are one area that will receive a lot of attention in the coming years, as companies seek to integrate realtime technology into their interactions with partners and suppliers.

Another challenge will come from the presence of all sorts of applications on the network. Indeed, in the future much of the IM traffic will be the chatter of applications talking to one another, with hardly a human involved. We will also see people interacting with applications over this realtime infrastructure—for example, in workflow processing and all manner of notification systems. A final challenge will be integrating presence more deeply into devices, applications, and ways of working. There are obvious privacy and security questions here, so the issues are not just technical but also social and perhaps even political.

Despite the challenges, IM will undoubtedly prevail. Ten years ago very few people were using e-mail or the Web as critical productivity tools, whereas now they are ubiquitous. The same will likely be true of IM. In another 10 years we will wonder how we could have thought that IM was just about teen chat, and how we could not have seen that interoperability using common standards was inevitable? Q

LOVE IT, HATE IT? LET US KNOW:

feedback@acmqueue.com or www.acmqueue.com/forums

JOE HILDEBRAND, as Jabber's chief architect, is responsible for the technical vision of its product line. He works on several Jabber open source projects. He is a member of the Jabber Software Foundation and an elected member of the Jabber Council. He previously served as chief architect for a regional IT consulting company where he built frameworks. He has also designed and implemented systems for sending and parsing Department of Defense battlefield messages. He holds a bachelor's degree in mechanical engineering from Virginia Tech.

© 2003 ACM 1542-7730/03/1100 \$5.00.