



O'Reilly Emerging Telephony (eTel) 2006 Conference

By Christine Herron

<http://spacejockeys.blogs.com/christine>

O'Reilly Emerging Telephony 2006

Environment: The O'Reilly Emerging Telephony Conference (eTel) was held on January 24-26, 2006 in Burlingame, CA

Caveats: These numbers are from one of today's large solution workshops, not from the larger group that I suppose will attend the keynotes or other sessions this week.

Count: Men: 98, Women: 6

XX Footprint: 5.7%

This is pretty crappy (er, I mean low), even for tech conferences. I know there are chick developers out there. (Anecdotal example: I am managing an Omidyar-internal open source development project, and all of our groovy developers from Colorful Expressions are female ... and at least one of them is here today, since she has started working on Ruby/Asterisk projects.) Maybe this ratio is reflective of the open source community, since the general software community is not overtly macho.

Cool VoIP Hacks Using Asterisk

Tuesday, 24 January 2006, 10:45 AM

Jim Van Meggelen from Core Telecom Innovations introduced a number of VoIP hackers to today's audience at eTel in San Francisco. Many coders are out there building front-ends for Asterisk, but they aren't hacks - they're spendy, glossy, and pre-configured. Here's some of the cooler projects discussed at show-and-tell:

Cool Hack #1: Performance Tuning for Asterisk VoIP

Kristian Kielhofner, the creator of AstLinux, showed the crowd his embedded Linux distribution for running Asterisk. This hack is important if you need your VoIP tuned for performance. **First:** Heavily loaded servers interfere with sound quality - but while you don't care if a web page downloads slightly slower, you may be annoyed if it happens during a phone conversation. **Second:** High availability. Larger-scale PBXs have redundant CPU cores, so users don't notice if one goes down. This redundant coverage is very difficult to achieve in Asterisk.

There's now an AstLinux virtual machine built with VMWare Workstation 5. This allows you to experiment with Linux and Asterisk without leaving your typical environment. This application has found a niche with people doing demos, training, or user groups.

Heartbeatd is a server daemon that checks for a "heartbeat" on other servers; if another heartbeat stops, then heartbeatd instructs its host machine to take over for the failed server. This works over both Ethernet and serial interfaces, and is supported by AstLinux. Unfortunately, if Asterisk were to crash, the daemon can't report that its own self has failed.

AstLinux-HA is most useful for VoIP-only applications, since it doesn't yet support routing.

Other limitations include Asterisk's native sounds. Christian had his pal Allison re-record all of the Asterisk sounds in the highest-quality format she could. They then converted directly into every Asterisk format. This eliminated some of the problems of transcoding into loss-based compression formats. These improved files will be publicly available in a couple of weeks.

Cool Hack #2: Playing Zork with IVR

This developer (send me his name if you know it, and I'll insert it here) is using Sphinx to develop an interactive voice response (IVR) version of Zork. You remember Zork ... open the mailbox, pick up the leaflet, etc. Sphinx still has trouble with interference and clipping. (Clipping is when you lose the ends of a word, a common problem in IVR devices.)

Clipping aside ... adventuring in Zork would sure be better than Muzak for killing time while in a customer support queue.

Cool Hack #3: Integrating Speech into Browsers

Ralf Muehlen stepped up with his VoIP Click2Dial. This hack begs the question...why should this application be so hard? There's plenty of multimedia integrated into browsers. VoIP Click2Dial allows browser users to drop directly into a customer service call queue, rather than having to call in separately from a telephone. This was built using Firefox, GreaseMonkey, and a web-enabled VoIP application.

Klipper is a related hack built using KDE. This could be extended to other browsers, and could also be extended beyond VoIP. If you had an interface to other telephone routing protocols, the API could be extended to include it.

Cool Hack #4: Analyzing User Navigation

Quinn Weaver stepped up from the audience to talk about Dido, a hack for analyzing user behavior and navigation through IVR or other telephony tree systems. This was actually built using XML + Perl rather than Ruby.

There were other cool hacks discussed, but the simplicity of these was pretty appealing. Check them out, and don't forget to tip your server.

Going from Hacker to VC-Backed Entrepreneur

Tuesday, 24 January 2006, 1:45 PM

Although the curriculum doesn't change much, the VC chat was very well-attended at the O'Reilly Emerging Telephony Conference (eTel) in San Francisco. By show of hands, the room was predominantly entrepreneurs. (If there were other VCs in the room, they were shy and didn't raise their hands ... cowards.) Here's some of the themes explored by Steve Tomlin of Avalon Ventures and a panel that included a business development director from InfoSpace and Rich Levandov from Masthead Venture Partners:

What makes a hack a product? The road begins when there is a paying customer involved, along with a contractual commitment to support the functionality. This has the potential to become a business once the total price to buy the product exceeds the total cost to develop it. Your hack is really a proof case for the product that you want to develop - something more than just a powerpoint for investors. You evolve from a hack when you let the needs of your customers start to inform how you develop.

One of challenges for an entrepreneur is making sure that your interests are in synch with the goals of your VC. For example: if you haven't raised outside funds, you can more easily sell your product as an asset to a corporate acquirer. With outside investors, your required exit valuation becomes much higher ... meaning that you need to take the product much further in order to sell it, as well as need to develop a more substantial customer base.

Someone with a very early stage business needs to get inside the head of a potential strategic acquirer, in order to be successful at selling. Innovation is where startups thrive, so find acquirers where competitive innovation and spe-

cific expertise is valued. One utterly rare (yet perfect) example - Josh Schachter at del.icio.us thought it would be neat to develop a folksonomy, and thereby learn about what users thought was interesting. The venture folks put in about \$500,000 to figure out the model and finish out the UI. Then, when looking at raising more substantial venture dollars, the liquidation preferences (and other typical VC deal terms) contributed to the attractiveness of selling the business to Yahoo!.

So ... what makes a business fundable?

- Be in an area of specific interest to the individual VC, but be aware of any existing portfolio conflicts.
- Have the right timing, don't be too early/ too late/undifferentiated
- Be part of a proven team, or have a personal relationship
- Meet the hurdles on basic business metrics

Other good advice, that isn't often said: be plainspoken and down-to-earth. If you're not likable, why would the VC want to spend two years working with you? This isn't too different from a job interview. Besides, if your dialogue is full of obscurity ... what are you trying to hide?

Hot Buttons du Jour

- Viral potential - show how you can spread rapidly while holding down customer acquisition costs
- Network effects - show that you have scalability that will make a market yours
- The elevator pitch - have a simple, concise story

Most importantly, make the investor want to buy what you're selling for herself. :)

Food for Thought

- There are plenty of sources for investment, so make sure that VC is what makes the most sense for you before bringing it in. Sometimes the best alternative for a founder is to build a product

and then sell it immediately to a strategic acquirer. (If you can finish development with low capitalization, then this is a totally viable direction.)

- Once you bring in VC dollars you have to actually build up a business around your product. This is required for the VC to "put enough money to work." Also, VCs build their own business operations based upon an expected pace - how many new businesses they have to fund each year, how long the growth for those companies will be nurtured, what market their investments will be in, etc.
- Do you want a VC with operating experience, or one with a sophisticated financial background? They're usually not the same person. Decide which one will get what you're talking about.
- If folks are talking a lot about an emerging market, there's a good chance that VCs have already seen hundreds of companies in the space. Pretend it's baseball and "hit it where they ain't."

(Cool Reference: The Autodesk File)

Applying Usable Design to Emerging Telephony

Tuesday, 24 January 2006, 3:45 PM

This session was a fun next step after the morning dev seminars at eTel. Once you've developed a technology, how do you design a product around it that will tap into how consumers want to use it? These panelists brought their varied expertise to bear on the issue:

- Brian McConnell, Open Communication Systems/RadioHandi
- BJ Fogg, YackPack
- Roberto Tagliebue, Nike TechLab

Simplicity starred in BJ's talk. He explored some interesting concepts on how user response relates to relative cost and benefit. e.g., is the cost lower or higher than expected? User responses range from frustration (cost/benefit lower than expected) to satisfaction (cost/

benefit exactly what expected) to delight (cost/benefit higher than expected). There are many shades of color around this - for example, when the cost is lower than expected, but the benefit is exactly as expected, users experience a sense of growing competency. (*Disclosure: Omidyar Network is in an investor in YackPack.*)

Brian went to the other extreme and focused on complicated devices. He particularly cited the Motorola Razr, which is an extremely cool phone but very difficult to use. The device is limited both internally and externally, so users must juggle hurdles that range from limited disk space to poor user interface design.

Multimodal interface design such as is required by the Palm Treo is particularly challenging. Mobile UI types include text/HTML, WAP/WML, SMS, MMS, voice, speech recognition, applet viewers, and native applications, such as those based directly upon the Windows API. Each of these UIs are good at enabling a different type of application. For example: SMS is great for sending alerts or checking 411. MMS is ideal for sending photos. Neither is good at both. A wish list of multimodal interfaces would include features like click to dial (text/web + voice) and voice email (native application + voice).

Roberto chimed in on future product design. Improvements to two-way technologies must be more than just making technology go faster, last longer, weigh less, or do more. Next-stage design needs to tap into the emotional aspects of connection. Designers must understand how, when, and why we use the things we do - and then apply those lessons to their products. Sitting here with a software background, this resonated with themes I've considered in developing use cases.

During final Q&A, there was a bunch of grumbling from session attendees. What's interesting about this is that it came from both extremes - developers are both frustrated by the low usability of very expensive phones, and defensive because of the increasing demands from usability design. Making a product simple is five times as difficult as making it complex.

There is even a perceived threat emerging from the consumer products companies, who have spent years working with psychographics, etc. and are now entering technology markets.

Some final tips on designing for usability:

- Always bring people to focus groups in pairs. Not only will they be more comfortable with a friend there, but they'll be more likely to talk to each other about what they think. This is much more insightful than what they'll tell you.
- Give people more than one object to respond to. You'll get more comparison and context.
- Connect design to real behavioral data. Use information that's already available from devices, or from service history, in order to drive what you think is important.

Emerging Telephony in a Yahoo!-Centric World

Wednesday, 25 January 2006, 9:45 AM

Versioning has finally been introduced to the world of voice communications. Apparently, we've all been mired in Voice 1.0 for years ... and at the O'Reilly Emerging Telephony Conference, the community has envisioned not only Voice 2.0, but also Voice 3.0:

Jeff Bonforte (who runs Messenger and Voice at Yahoo!), outlined the basics of Yahoo!'s vision for emerging telephony. This begins with the move from Voice 1.0 to Voice 2.0: here, voice moves to software and becomes a platform in which prices are based upon flat fees. This evolution has been happening of late. What's next? Voice 3.0, which takes that platform and transforms it into a medium for network integration, in which pricing is based on per-application or per-transaction use.

Yahoo!'s emerging telephony strategy describes a utopia of rich user experience and content. (And interestingly, one of the benefits listed is increased access to Y! itself. Huh! That being said, even Yahoo! acknowledges that consum-

ers aren't ready for Voice 3.0. Bonforte cited a stat (backable?) that there are more rotary phones than Vonage phones, which says a lot about consumers' death grip on Voice 1.0.

So, where is Yahoo! in this market shift? Yahoo! believes that it will make the leap to 3.0 more comfortably than its competitors. Bonaforte touted these advantages to the eTel audience:

- Yahoo! has existing relationships with consumers
- Yahoo! has a deep inventory of content, developed both by and for consumers
- Yahoo! can apply its deep learnings around its core asset — search — to voice recognition. e.g., Yahoo! search can handle both ten ways to spell a name, and Yahoo! voice will handle the ten ways to pronounce it. This was earned by years spent processing massive amounts of user input.

Given this confidence, it's not surprising to also hear that Yahoo! is "extra" confident that Google won't be able to bring as much to the telephony market. e.g., Google has great search expertise, but smaller communities for mail and messaging relative to Yahoo!. I'm not sure that I buy into this argument, since it's been hard to overestimate Google's rampant entrepreneurship, but the consumer dynamics are worth considering.

OpenZoep: Open Source VoIP

Wednesday, 25 January 2006, 11:00 AM

This morning's eTel presentation on open source VoIP was compelling. Erik van Eykelen from Voipster BV stood up to talk about OpenZoep. OpenZoep (pronounced "open soup") is a neat-looking solution that offers plenty, and you can have this under either a GNU/GPL or a commercial license:

- in/outbound P2P calls
- outbound PSTN calls
- IM
- presence
- phonebook

On the client side, OpenZoep has a VoIP engine built using C, and an XPCOM API that makes the engine accessible from either Firefox or other Mozilla-based applications. This client creates a VoIP app on your own desktop, which then does the work of translating simple commands (hang up, pick up call, etc) into complex commands that are sent to the server. Voipster chose Jabber XMPP to handle this server communication.

Server-side on OpenZoep is also built using Jabber code. This handles SIP/PSTN well, but accounting and billing are still difficult, since there isn't a good open standard for accounting.

Eykelen must constantly be asked why they are bothering to develop this software, when Skype is still free. His answer? OpenZoep offers both a GPL and a commercial license, is interoperable with both IM and SIP, and (so he claims) offers an easy, extensible API. The company's goal is to drive complete interoperability (and hence competition) between VoIP systems, so that there is just one VoIP phone book.

More info is at extensions.mozilla.org - the company just uploaded OpenZoep yesterday and they've 1,000 downloads so far. Try it out and let them know what you think.

Why Consumers Aren't Adopting VoIP

Wednesday, 25 January 2006, 11:15 AM

A number of speakers delved into the challenges of VoIP adoption at the O'Reilly Emerging Telephony Conference. David Beckemeyer from TelEvolution posited that VoIP 1.0 is offering plenty of provider choices ... unfortunately, pretty similar choices. The trouble in this market is that it's difficult to switch vendors. Termination fees, loss of address, etc. are all very similar to how difficult it used to be to switch cell phone providers:

Beckemeyer's view of online competition breaks down into four segments: applications, access, network, and physical connections. There was some beating on the old Web 2.0

(sorry to use this phrase) drum on how anyone can offer services now, since it's so cheap to tie together existing application components and so on. This wasn't tied too well to why consumers aren't adopting VoIP, though of course we can imagine that the complexity of this segmentation implies a hurdle to adoption.

PhoneGnome is intended to create the same impact that Earthlink's TotalAccess created. (Beckemeyer is founder of Earthlink.) e.g., the reason people aren't adopting VoIP is not only because the technology is still too difficult for the everyday Joe, but also because there hasn't been any effort to educate the consumer, or to provide them with tools that will actually increase their level of competency. This idea has some merit - it's not just about making technology simple, but also about understanding user behavior and making an effort to inform and educate your customers on the possibilities.

Wednesday, 25 January 2006, 11:30 AM

Phil Wolff from Skype Journal was up to address VoIP adoption at eTel as well, and kicked off with a thick stack of slides containing Skype statistics. 240 million downloads, 6 million registered users, etc. While this is all well and good, he expressed some frustration with how the Skype API is getting farther and farther behind the product. If the API could instead lead the product, it would be a good step towards leveraging open development efforts. Ideally, a more open development effort would provide the usability and accessibility enhancements necessary for adoption

All the slide decks aside, consumer VoIP must be easier to use, and the benefits must be better articulated. Until then, why give up a perfectly functioning land line?

Community VoIP and Wireless Efforts Face Goliaths

Wednesday, 25 January 2006, 1:45 PM

Whether you're miles into the country or thirty stories up in a high-rise, there are advantages to shared access. At eTel, speakers

touched upon how communities can share these resources to improve their own access while reducing cost:

Brian Capouch, a computer science professor from St. Joseph's College, is addressing the challenges of the "rankly rural" - and he doesn't mean residents from a small town, he means those who live literally miles into the country. The hurdles these individuals must face in obtaining online services include:

- There aren't very many rural residents, so there aren't many advocates and there isn't any political power
- 3000 users spread over 1500 square miles is extremely low density - and that represents 100% market saturation
- Many users have accepted that they simply will never get Internet access, and therefore don't know anything about using it

Capouch is using a cheap, open source approach to bring VoIP (and Internet) to these rural residents. He's created the Indiana Farm Net, a wireless voice network built from scratch. While he's happy to handle small challenges (the ants that get into his access points), his clear-voiced frustration with large challengers (the telcos that try to squash his community efforts) earned him the longest, loudest round of applause of the conference so far.

Wednesday, 25 January 2006, 2:00 PM

Ejovi Nuwere from FON described a similar approach, but in this case, the model is being used to create shared network access in more urban and suburban areas. In FON's utopia, all individuals would share their WiFi hotspot as a member of a community in which all members shared their hotspots. For example, you share (with assurances about security) your home hotspot with the FON community. Then, when you are downtown or out of state, you use someone else's hotspot.

Nuwere's approach is a fascinating (and very ambitious) approach to this problem, and hats off to his effort to organize and mobilize an

international community of hotspot owners. If you check out FON's web site for collaborators, you see many of the usual suspects for bottom-up communities, from Dan Gillmor to Ethan Zuckerman.

Both Nuwere and Capouch cited mobile operators and telcos as the Goliath's threatening their projects. In both situations, incumbents have invested in lobbying and other activities intended to limit what these grassroots efforts are trying to accomplish.

Putting Voice (VoIP) onto Muni WiFi Systems

Wednesday, 25 January 2006, 2:45 PM

Public safety support, government agency networking, and residential Internet access are all familiar reasons to roll out a municipal wireless system. But what about voice? The mesh network that enables ubiquitous urban coverage could also support a whole host of voice applications:

Steve Rayment from BelAir Networks walked through the variety of mesh network types that they support: single-radio shared, multi-radio switched, and dual-radio shared. Shared mesh time-slices the spectrum between each node, so you can process ~ 1400 packets/second. Switched mesh fills the spectrum between each node, so you get ~ 5000 packets/second. With switched mesh, BelAir estimates that you can handle 4000 users per node.

This all sounds great, but it means that quality of service becomes mission-critical. Modest disruptions or changes in latency can create ripples of adverse effect. Deploying carrier-grade, voice-capable networks will be much more challenging than supporting Skype calls.

To this point: Matthew Gast from Trapeze Networks dug into the issues of QoS. VoIP over 802.11 is currently dominated by proprietary standards for security, radio management, and CAC extensions. For VoIP to work better over 802.11, 802.11 needs to work more like the cell phone network does. Applications for capacity management (in particular, a directed

handoff to the next access point as you walk down the street) are wanting. To get past the myriad issues here, 802.11 hotspot managers need to stop thinking like data network managers, and start thinking like telephone network managers.

Here's to the O'Reilly Emerging Telephony Conference (eTel) for bringing these concepts together.

Do We Really Need an Open Phone?

Wednesday, 25 January 2006, 3:15 PM

Why are people deploying Linux onto cell phones? The number of vendors is starting to add up. There are technical and financial advantages, but Bill Weinberg from the OSDL posited at the O'Reilly eTel Conference that it's really because it is becoming strategic via platform consolidation. In addition, the holy grail of "open phone" would allow hackers to modify its user interface, improve performance, upgrade the kernel, or invent new applications:

There's a new divide emerging in the open source community - is open source a means to an end, or an end unto itself? This leads to some other existential questions that Weinberg posited:

- Are phones mono-function, or general-purpose computing devices?
- Should all devices be hackable? Routers ... phones ... and pacemakers?
- Are consumers better served by open or closed devices?
- Why would you want an open phone?

Weinberg actually wants to hear answers from the community on these questions - send him mail with your answers.

Is Open Source the Panacea for Mobile Application Deployment?

Wednesday, 25 January 2006, 5:15 PM

As mobile device variety is increasing, vendors such as Funambol believe that open standards are the best way forward for device manufacturers and carriers. And (un?)fortunately, RIM and Microsoft can't address the bulk of the market. At the O'Reilly Emerging Telephony Conference (eTel), Fabrizio Capobianco of Funambol states that the market disconnect is thus:

“Proprietary solutions have been developed and sold without the developer or consumer in mind. Developers want a transparent community of peers contributing to the code, making it better for developers to turn into differentiated products and services for their organizations and companies. Consumers want access to mobile email and applications, regardless of network type or device manufacture.”

Funambol is the latest vendor to attack the problematic environment of mobile application development. Frankly, as a former maker of developer tools I believe that their environment is very well-done. They provide plugins that can be downloaded to mobile devices, as well as a server that processes SyncML and sends the message to the application developer's own back-end server.

BUT, is open source the panacea for mobile development? There's some truth to Capobianco's statement, but open source is not the only solution to this problem. (Has anyone noticed how pervasive email is, regardless of which OS you are using?) And the lovely development solution of Sync4J would be just as lovely if issued under a commercial license. Though undoubtedly, it would be a lot more expensive.

Yahoo! Mobility Breaks Open the Space-Time Continuum

Thursday, 26 January 2006, 9:45 AM

Yahoo! Research Berkeley is a new research partnership between Yahoo! Inc. and UC Berkeley, looking into the intersection of social media and mobile media technology. Marc Davis came to the eTel conference to talk about some of their projects:

Davis is applying Moore's Law to devices such as digital cameras. Since cameras have become so much lighter and cheaper, many folks carry one with them every day in the form of their cell phone. Having a camera with you every day changes how you think about photography. Have camera phones already become input devices? Undoubtedly, the folks at Flickr think so. But what about what's next? What will it take for cell phones to become sensor devices?

Mobility can fuse context, content, and community together; mobility embodies what someone is doing, where they are doing it, and their ability to share it. Here's some examples of how it flows through, in functional terms:

- Content: Created using free text, tags, structured metadata, image analysis, or weather services
- Context: Understood in terms of 'where' via cellular IDs, GPS, bluetooth, and image analysis; understood in terms of 'when' via network time servers or calendar events
- Community: Participate (and know who else is) through tagging, searching, sharing, and remixing

Given this perspective, how does context tie to community? How can I share more easily? The current trend is for the introduction of context awareness to the device, from face or place recognition to bluetooth network awareness.

Davis sees the Yahoo! network as a platform for telephonic applications. Yahoo! knows where people are and what they are interested in, so they are uniquely positioned to do something with the information. This is just swell

corporate hype (and from a researcher, too!) - I was much more intrigued by his groovy exploration of space, time, and social space. In this world, the phone supplies your sense of place in the continuum.

Driving VoIP Towards Open Standards

Thursday, 26 January 2006, 11:15 AM

I can barely keep up with all of the great stuff here at O'Reilly eTel. (Carpal tunnel, here I come.) Michael Robertson of Linspire started SIPphone with the vision of an open dial tone that anyone in the world could connect to. What do you need for this?

- Open directory (don't discriminate against users from different systems)
- Open standards (let us all talk to each other)
- Common codecs

Robertson notes that when users have a hardware client, they are much more likely to use VoIP. While only 10% of the SIPphone user base is hardware-based, 35% of SIPphone calls are from hard clients like routers. (Yes, the other 65% of SIPphone calls are from soft clients.) Given this, he's paying a lot of attention to VoIP hardware. SIPphone's VoIP uses SIP, but their IM and presence is based upon Jabber, and they peer using GoogleTalk. SIPphone also hooked up with the router manufacturers to develop a standard called Plug-n-Dial in order to support auto-provisioning for consumer devices.

Gizmo is now available for download. This VoIP client competes directly with Skype, but is all standard-based. Robertson believes that Skype's closed system fails the test for using open standards. (Skype does have a SIP gateway, but they don't offer it directly to customers.) MSN/Yahoo!/AOL are also closed, with some talk but no action - there's still little delivery on the announcements that these companies have been making. On the other hand, Robertson loves Google's openness despite his hate of the Google bandwagon. GoogleTalk and Jingle are what the industry needs to drive

open standards and interoperability. Besides, while Google's VoIP services don't have a large user base, they "are Google" and they do have a strategy.

Here are the parts of an open VoIP ecosystem that still need work:

- Standards-based NAT and firewall traversal that can compete with Skype. SIP-aware devices actually end up mangling your packets, so maybe it's better to avoid them.
- Codecs. Everyone is using Global IP Sound (GIPs), but it's proprietary and costly. We need a streamlined way to talk to hardware devices.
- MIME types. There's half a dozen ways to initiate a call. There needs to be one that everyone can get behind.
- Address book data standards. Cell phone companies lock you down the most when you have a large address book, since there's no easy way to easily take addresses with you when you switch providers.

Today, SIPphone and Google are the only companies that Robertson believes have a commitment to open standards. Need more to develop competitive pressure that drives the larger companies to compete based on the actual service being provided. Not because they've locked you in, or because that's the service that your friend uses.

Owning Your Identity in the New World

Thursday, 26 January 2006, 11:30 AM

The issues around identity extend far beyond the borders of emerging telephony, but it was a recurring theme at the O'Reilly Emerging Telephony Conference. Both the hackers and the Microsofties had things to say:

Ten years ago, there tended to be just one large provider for any given service. Today, there are many more — and users switch providers much more easily. This brings up the concept of who owns your identity, and how complex your

identity is. Do you bring it with you when you leave? Do you abandon it behind you? Do you add it to a stack of baggage that you must carry with you forever?

The rows of identity stovepipes that we use can add up - phone, email, IM, web sites, blogs, VoIP, etc. How many ways are there to connect? Well, let me count mine...at least, those that I still check or use daily:

- 12 e-mail aliases: "me" at gsb.stanford.edu, stanfordalumni.org, columbia.edu, christine.net, spacejockeys.com, cyclepartners.com, omidyar.net1, omidyar.net2, earthlink.net, alphachiomega.net, artsfestsf.org, missionresearch.com
- 7 phone numbers: Home, Home Office, Mobile, Omidyar Main, Omidyar Direct, Space Jockeys Fax, Omidyar Fax
- 2 IM addresses: "me" at Yahoo!, .Mac
- 5 web sites: Christine.net (still haven't integrated with this blog), Omidyar.net, Mission Research, LINES Ballet, Space Jockeys
- 3 blogs: Christine.net, Mystery House (still in development), Advice Line (still in development)
- 1 VoIP address: "me" at Skype

To be eloquent: Yuck.

So, what's in the identity landscape? Johannes Ernst from NetMesh shared the landscape that he sees. There's a group forming called the Liberty Alliance, that is trying to build a "trusted digital ecosystem." The Web Services Interoperability Organization is trying to push Web services interoperability, which ends up making requirements upon identity. Finally, Ernst proposes basing identity on URLs with YADIS Capability Discovery. This identity URL sits at the bottom of a stack with OpenID authentication and LID authentication, along with any authentications that need to be layered on by service providers such as Six Apart. Using a URL as a simple form of identity is a valid idea, but getting everyone to agree is going to be a huge challenge.

Things that Ernst wants vendors to think about:

- Do customers only use my service?
- Would social functionality help my product?
- Do I lose visitors on the sign up for an account page?
- What would identity data theft do to my reputation?

Even Microsoft's talk wound its way to a discussion about identity today. In their vision, you take these silos of identity and federate them. Once you solidify your identity, you can wrap it in context, presence, relationships, etc. This all makes sense, but then Amritash Raghav went on to build a crazy bullseye of Unified Communications Vision with arrows bursting forth. Which perhaps also makes sense, but it was a bit dense to process.

This is all good, interesting stuff - but the user needs to be in control and that wasn't clear from any of these talks. e.g., this needs to be different from an open-standards form of Passport. My business is my business, and there's no need for it to reside on someone else's server.

Using Open Source Telephony to Drive Advocacy

Thursday, 26 January 2006, 1:45 PM

Today's speakers at O'Reilly eTel had a solid do-gooder streak. Tad Hirsch from MIT's Media Lab is wholly invested into the intersection of art, activism, and technology. He's currently using open source telephony to promote community development:

Cell phones are more popular devices than PCs are in immigrant communities. There are many barriers to technology access that must be addressed: awareness, language, trust, and culture. It's not enough in this world to know that a technology exists; you must also believe that you are allowed to use it. If you can get past these cultural hurdles to a technology, then you still need to learn how to use it.

Hirsch spoke of a variety of creative approaches to community development. For example, he's connected volunteers to service clients with an asterisk-enabled, distributed phone book. Volunteers can easily receive calls at home, making it easier to drive participation. The combination of Asterisk + Apache + MySQL can also enable ad-hoc conference calls, which not only extends the network, but also allows non-profits to provide better services. e.g., improve communication by pulling in an interpreter in real-time when required for client services.

Thursday, 26 January 2006, 2:00 PM

Blaine Cook and Rabble Henshaw-Plath from Odeo are helping to drive activism by enabling voice via emerging telephony. During the Republican National Convention, they set up an Asterisk server that provided a solid platform for advocacy:

- Text-to-speech processing for mobile RSS updates from Indymedia
- Event calendars using TTS and RSS
- Live radio call-in
- Streaming MP3 radio stations

Total cost? \$7.00. (Undoubtedly, not including the value of their own time and expertise.)

As the 2004 election came around, Cook and Henshaw-Plath made an ambitious attack on mass messaging:

- Goal: get SMS messaging to 20,000 volunteers
- Hack: set up Java applets inside browsers to support P2P SMS
- Accomplished: messaging for 2,000 volunteers, with volunteers sending out info on polling locations in order to get out the vote on election day

Total cost on this one? \$1500 for VoIP message delivery, including 15 servers. (While this doesn't include time costs, at least the SMS delivery was actually free.)

In both of these anecdotes, these two hackers proved that open source (in this case, Asterisk)

is empowering - providing that an organization has the technical sophistication to take advantage of the reduced cost of market entry. We all still need to work on a solution for that need.

Death to the Carriers from a Thousand Widgets

Thursday, 26 January 2006, 2:15 PM

Was last year the year for mobile content? Will it be this year? Or what about next year? At eTel, Benjamin Keighran from BluePulse offered up an explanation for why mobile content hasn't taken off:

It's overly complex for producers to publish mobile content, and there's no one-stop-shop for services. On the other side, consumers want everything on their PC to be accessible on their phone. (Well, do they really? I'm not too sure.)

BluePulse claims that what's needed is a clean consumer interface to this content from their cell phones, and an easy, one-stop shop for producer services. They direct consumers to set up an account at the BluePulse web site, and then select the widgets that they want to use. *(It wasn't clear from the talk, but I would expect that widgets could download to both your PC and your cell phone ... or at least I hope they do, since the idea is to share content between these devices.)*

In the last month, the BluePulse community has grown by 15% per day. The company offers free widget downloads, and consumers are checking them out. The company's SDK is coming soon, which will add appeal for producers - in theory, you could write your own widgets and people can download them, all without any carrier intervention.

Keighran is clearly earnest and excited about the impact BluePulse could create. I'm looking forward to the SDK to see how much they really could break carrier control on mobile content distribution.

Jabber - Is It Really this Good?

Thursday, 26 January 2006, 4:30 PM

Peter Saint-Andre from Jabber gave a very speedy (and I thought I talked fast) talk on Jabber and Jingle. Here's the basics, in case you aren't at O'Reilly eTel and haven't been riding the Jabber bandwagon:

Jabber is a "highly programmable presence and messaging framework, uniquely capable of bridging applications, networks, devices, multi-media, and protocols." More to the point, it's a BSD-licensed open source server that supports server-to-server communications within a decentralized network. The company went to the IETF and formalized its protocols as XMPP.

Saint-Andre doesn't know how many servers are running Jabber. The company's best guess is over 50,000. (Unfortunately, supporting a decentralized network is terrible for marketing.) The company's web site has a who's-who laundry list of large companies and agencies that have adopted Jabber.

Jabber vs. SIP (according to Jabber)

- Authentication is optional in SIP (where did this message really come from?)
- No spam or viruses on Jabber
- Jabber packets are smaller than SIP packets

Jabber seems to work pretty closely with Google. Jingle is Google's XMPP extension that enables Jabber clients to share media content. In addition, Jabber is working with Google on JEPs - once the protocols are set up for the standards process, they will be released. Hopefully, Jabber will maintain its connection to the community as it cultivates these big company relationships in an effort to drive standards.