Presence is beautiful: integrating legacy and IP enabled voice and video devices into a presence platform

Bradley Whittington and Alfredo Terzoli
Department of Computer Science
Rhodes University, Grahamstown, South Africa
Tel: +27 46 603 8291
studies@radbrad.rucus.net a.terzoli@ru.ac.za

Abstract— **This paper presents an analysis of current developments in the Jabber Messaging protocol and Asterisk multi-protocol software PBX. The developments could integrate both legacy and IP enabled telephony device status into the presence framework, to enrich the concept of user identity in a networked environment. The paper discusses experimental work on the Jingle framework extension to the Jabber protocol which provides a mechanism to initiate peer to peer connections within Jabber. Using Asterisk, the Jabber extensions could provide a platform to easily integrate legacy and IP based voice and video device status into the presence framework. The paper will examine how Asterisk and Jabber could be leveraged to easily represent devices in a presence framework, while minimising cost of legacy integration.**

Index Terms— **Asterisk, VoIP, Jabber, Identity, Presence, Instant Messaging, XMPP, pbx, legacy devices**

I. Introduction[1]

USER identity and presence management in the virtual environment is not a new topic. There has been much discussion on the topic, since it is seen as a step towards creating a more immersive and potentially productive user environment. For example, we can be more efficient if we don't have to pick up a phone and dial a person, only to discover an engaged tone. The experimental Jingle framework [6] aims to provide a standard mechanism to initiate peer to peer exchanges (such as voice and video) within the Jabber protocol. The inclusion of a Jabber/Jingle channel within the Asterisk multi-protocol gateway provides interesting potential; the channel could allow status information from legacy devices to become available to other presence aware devices in a straightforward, standardised manner.

The Asterisk software PBX already provides a multi-protocol gateway to interface both legacy and IP devices. In the existing iLanga [9], [8], [5] development work, we saw that it was possible to impose the concept of a single user identity across multiple devices (including legacy PBX phones, SIP, H323 etc.), and provide an aggregated status indicator in the web-based based iLanga interface. Bearing this in mind, the experimental development of an Asterisk Jabber channel potentially provides a mechanism to push the identity state information into an already mature messaging and presence platform.

This paper will first examine the eXtensible Messaging and Presence Protocol (XMPP [11], [10]) in relation to Jabber, discuss general presence issues and terminology, move on to the recent extensions to the Jabber protocol, and examine how these developments could be leveraged with Asterisk to push legacy device status to the desktop in the context of a presence framework.

II. XMPP and Jabber

The eXtensible Messaging and Presence Protocol is a set of IETF ratified standards maintained and implemented by the Jabber Foundation. Jabber is the de-facto implementation of the eXtensible Messaging and Presence Protocol (XMPP). The XMPP open standard builds on the initial publication (Feb 2000) of RFC 2778 [3]: "A model for presence and instant messaging". There are currently two protocols published by the IETF which are approaching the problem domain from two different directions, Session Initiation Protocol (SIP [4]) for Instant Messaging and Presence Leveraging Extensions (SIMPLE), and eXtensible Messaging and Presence Protocol (XMPP). This paper will focus on XMPP, and specifically Jabber.

III. Presence

Wikipedia states "In computer and telecommunications networks, presence information conveys a presentity's availability and willingness to communicate" [14]. As a user of a presence notification service, you can subscribe to other users published states on particular communication channels.

Presence information is becoming increasingly important, as the concept of a buddy-list and the address book become central to business operations. Knowing

[1]This work was undertaken in the Distributed Multimedia Centre of Excellence at Rhodes University, with financial support from Telkom SA, Business Connexion, Comverse, Verso Technologies, Tellabs, StorTech, THRIP, and the National Research Foundation.

the state of a user or their associated devices can allow you to be more selective in your communications, or place the virtual equivalent of a "Do not disturb" sign on all your communication mechanisms. In an ideal world all relevant devices would be able to notify and interact on a unified presence platform.

A. Building a unified presence platform

There are two possible approaches to a unified presence management platform: a closed/proprietary system, and an open/inter-operable system. Closed systems can benefit from quick implementation and extensions on desired platforms, but suffer from increasing research and development costs as more platforms are catered for, while potentially marginalising user groups if their platforms of choice are not currently supported or will not be supported for political reasons. Open systems suffer from committee deliberations on protocol standardisation and extension, with a potentially longer time to implementation from initiation. Already established open systems/standards allow shorter research and development times for new products, since they can leverage existing discussions and platforms to further enrich the design landscape. Implementations of open standards benefit by pushing implementation costs into the open market, and onto the original equipment manufacturer. There have been a number of attempts to create closed systems of identity and presence management, which have seen significant user take up, but are ultimately tending towards standardising mechanisms of inter-network operations. Ideally there should be some level of inter-framework/network method of communication, so that users and devices are free to choose the implementation that suits their needs the best.

B. Passive vs. active state queries

Passive and active state queries can be aligned closely to the concept of push and pull. A passive query can be likened to an information push since state information is pushed (not necessarily obtrusively) to a another user already subscribed to state notifications. In the case of instant messengers this amounts to a state icon associated with a user, or pop-up messages denoting user state changes (e.g. "Bob is no longer Away"). An active state query can be likened to an information pull, i.e. the user actively seeks out state information. In the example of telephony this would be an engaged signal when you have dialled a user's extension, indicating they are not able to receive calls.

C. Sensed/Automatic and Active state changes

An example of sensed/automatic state changes would be a client application that monitors keyboard and mouse activity to decide if a person is at their desktop or not. Locking of a workstation can provide an automatic Away state, etc. Sensed states could be useful when used in conjunction with inter application communication - e.g. setting "do not disturb" automatically if you have particular applications open.

Active state changes involve active user intervention, for example setting your away message to "Away for the weekend". Active state changes can be useful for setting "do not disturb" messages, so that other users will know to leave you alone.

D. Basic presence information

Contemporary instant messaging services allow users to set their state to some fixed states (Available, Busy, Away, etc.), as well as arbitrary sub-states (e.g. Away for the weekend). The value of presence information is borne of the described state of a user on a particular channel of communication. In a simple telephony system user states are usually "Available", "On the Phone", and "Unavailable". In iLanga [5], [8] user interface allowed users to see other user's telephone states without lifting their handsets, allowing them to make decisions based on the provided information. In these two examples we see that while there is value in the communicated states, we are limited to making presence related decisions based on the state information we have available, we cannot necessarily see if a user is unavailable on IM if they are on the phone, since there are two separate systems in operation.

E. Multiple Points of Presence (MPOP)

The concept of multiple points of presence describes how one user may have many states on different communication channels. A user may be marked as available to chat via instant message, but may be marked as on the phone. Presence frameworks should publish channel related state information, so that relevant states can be monitored by interested parties, and state aggregators can be employed to provide an aggregated view of a user's state.

F. State aggregation

Effective aggregation of MPOP is important within a presence and notification framework. For example, within the iLanga system, a user could have multiple phone devices (SIP, IAX, H323), and the user interface presented an aggregated picture of a user's device states, since for most cases it is only valuable to know if a user available to speak on one of the devices (unless somehow talented, people can usually only speak on one device at a time).

G. XMPP Resources and Resource priorities

XMPP/Jabber caters for MPOP by associating clients with a "resource" (e.g. bob@jabber.org/Home, where Home is a specified resource). Resources are associated

with priorities, so that they can be mapped relative to each other. As a default, messages will be sent to the resource with the highest priority, but a specific resource can be singled out. Resources have a number of applications, by allowing a user to sign in from multiple locations (/Home, /Work, /Mobile, etc.), or having specific devices available as resources (e.g. subscribing to Bob's /CoffeeMachine, would yield status information for his USB coffee maker). State aggregation also occurs using resources, i.e. if Bob is inactive on all his resources, then his identity will be marked as away.

H. XMPP and SIMPLE

XMPP and SIMPLE differ in two fundamental aspects: XMPP is a standard ratified from the already implemented open source Jabber protocol, while SIMPLE builds on the IETF specified SIP standard. The core aim of XMPP/Jabber is to push heavy work onto servers where possible [12], to keep client implementations simple and straightforward. SIP pushes complexity to the client [4], with the intention of creating rich clients and efficient, scalable servers. These design decisions have led to widescale adoption of XMPP within organisations for messaging and presence information, primarily since the protocol is mature, with many stable reference implementations. SIMPLE has only seen use on existing SIP stacks, such as heavy VoIP phones, Microsoft's Messenger protocols, etc.

IV. Jabber, Jingle and Asterisk

With the implementation of the GoogleTalk application, Google Inc. decided to follow a standards based approach to producing an IM and voice client. As mentioned, there are two particularly well documented, standardised presence and IM protocols: SIMPLE, and XMPP. SIMPLE is relatively immature, growing out of the SIP standard, while XMPP is a mature presence platform (started in 1999), with a great number of ratified extensions and implementations. XMPP did not provide a standard mechanism for managing and initiating peer to peer sessions (such as voice and video). Google chose to follow the XMPP/Jabber standard, since Google services aim to enrich user identies within the Google set of services, and GoogleTalk extends Google identities further into the desktop.

A. The Jingle Framework

Google initiated work on extending the Jabber/XMPP protocol to provide a standard extension, in collaboration with the Jabber Foundation, for initiating peer to peer sessions, negotiating a number of network issues, such as NAT traversal and proxied connections. The current experimental standard is codenamed Jingle [6]. Jingle is essentially a framework within Jabber to provide a mechanism to initiate various peer to peer sessions. Specific sessions are specified in other Jabber extensions, which extend the Jingle framework. Example Jingle extensions are: A Jingle transport method that results in using the Inter-Asterisk eXchange protocol (IAX) for the final communication [2], A Jingle transport method that results in sending data using the Real-time Transport Protocol (RTP) over a raw User Datagram Protocol (UDP) connection [7].

B. Asterisk and IM/Presence

Previous work to integrate Jabber instant messaging into Asterisk (e.g. Asterisk-IM plugin for the Wildfire Jabber server [1]) has suffered because of no ideal implementation path. Until the Jingle framework, working with Asterisk servers involved loosely coupled integration, resulting in the need for explicit references in the dialplan, rather than implicit notifications. Jingle provides a better framework for building an effective Asterisk channel which allows for tighter integration into both the Jabber and Asterisk servers. Work on the Jingle standard has also pushed the Jabber protocol to deal more effectively with non-IM resources [13], resulting in more efficient information transfer, and scope for better client user interfaces.

C. Messaging and legacy devices

With the addition of a text-to-speech gateway within Asterisk, a number of interesting use-case scenarios are presented, empowering legacy phones with a rich toolset, and extending the reach of existing XMPP clients. Users could send text messages to voice devices, or voicemail facilities using standard desktop XMPP clients. Messages could even integrate prompts, allowing users to ask questions, and receive a response based on a provided response set (e.g. the telephone user could be prompted "Push 1 to reply Yes, Push 2 to reply No"). Legacy phone devices could be used to access logged messages, send messages (typed using the keypad) to available friends, or change a users status.

V. Bringing legacy device status to the desktop

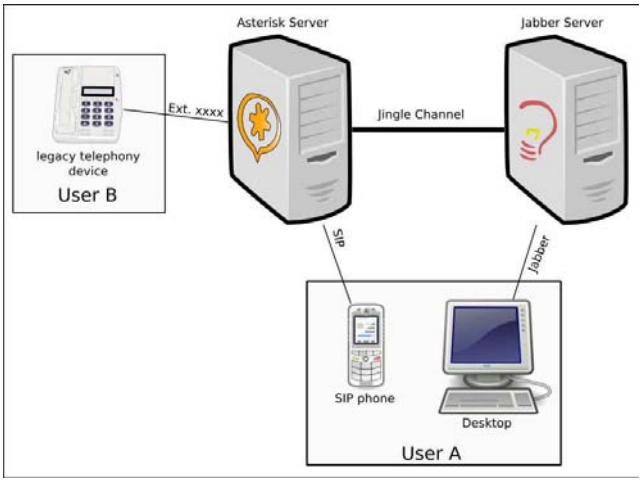

Fig. 1. A possible system configuration

Pushing legacy device information to presence aware devices and software is a challenging, valuable goal. The ability to integrate legacy systems into a modern IP/Presence aware network means less capital expenditure with less system changes, providing a migration path to future technologies, while allowing system administrators the freedom to choose appropriate technologies.

Using Asterisk attached to either an existing PBX system, or as the primary PBX would allow an administrator to associate legacy extensions with particular Jabber identities. Device availability and status could be piped from Asterisk to the user's identity in the Jabber server.

A. Monitoring and accessing legacy devices from the desktop

If legacy devices could be effectively associated with a users' organisational identity, along with status information, presence information for the user could become more effective. Knowing a user is currently occupied on the phone without picking up your phone could save you time and frustration. Having the ability to set event specific actions to occur based on presence could be useful: e.g. setting your client to automatically initiate a call with the user when their phone status changes to available (provided you are available).

Indirect call initiation is a powerful tool, for example being able to initiate a call to another user from your hard phone to their hard phone using your contact list.

B. Presence and Context sensitive information

An additional function of presence and device information is associating context sensitive information. Some example scenarios: You recieve an email from Alice, and when you read the email you could have access to the availability of Alice on various channels, in addition to "Reply to email", her IM and telephone status could be displayed, allowing you to initiate an instant message conversation, or a voice call at the click of a button.

VI. Conclusions

The experimental extensions to the Jabber protocol, in conjunction with the construction of a suitable Asterisk channel will enable desktop access to status information associated with a foreign user's telephony devices. The XMPP presence platform could cater for the simple case of a legacy TDM device's states, to the possible richer states that SIP capable devices could provide.

The ability to integrate legacy systems into a modern IP/Presence aware network means less capital expenditure with less system changes, providing a migration path to future technologies, while allowing system administrators the freedom to choose appropriate technologies.

It is clear that the experimental extensions to the XMPP protocol are a significant step towards a unified presence platform. Coupled with effective, standardised, protocol/network gateways more focus can be placed on creating an effective presence management platform, and less on the implementation of the devices.

Bradley Whittington is currently reading towards an MSc in Computer Science at Rhodes University, Grahamstown (email: msc@radbrad.rucus.net)

Alfredo Terzoli is currently Project Director at the Centre of Excellence at Rhodes University, Grahamstown (email: A.Terzoli@ru.ac.za)

References

[1] Asterisk IM. http://www.jivesoftware.org/asterisk-im/, 2006.

[2] Cano, A. F., and Saint-Andre, P. JEP-0179: Jingle IAX Transport Method. JEP-0179 (Experimental,Standards Track), 2006.

[3] Day, M., Rosenberg, J., and Sugano, H. A Model for Presence and Instant Messaging. RFC 2778 (Informational), Feb. 2000.

[4] Handley, M., Schulzrinne, H., Schooler, E., and Rosenberg, J. SIP: Session Initiation Protocol. RFC 2543 (Proposed Standard), Mar. 1999. Obsoleted by RFCs 3261, 3262, 3263, 3264, 3265.

[5] Hitchcock, J., Penton, J., and Terzoli, A. A multilanguage, open source approach to connecting user interfaces to a next-generation PBX. In *South African Telecommunications Networks and Applications Conference* (2004).

[6] Ludwig, S., Beda, J., Saint-Andre, P., and Hildebrand, J. JEP-0166: Jingle. JEP-0166 (Experimental,Standards Track), 2006.

[7] Ludwig, S., Beda, J., Saint-Andre, P., and Hildebrand, J. JEP-0177: Jingle Raw UDP Transport. JEP-0177 (Experimental,Standards Track), 2006.

[8] Penton, J., and Terzoli, A. Asterix: A converged TDM and packet-based communications system. In *South African Telecommunications Networks and Applications Conference* (2003).

[9] Penton, J., and Terzoli, A. ilanga: A next generation VoIP-based, TDM-enabled PBX. In *South African Telecommunications Networks and Applications Conference* (2004).

[10] Saint-Andre, P. Extensible Messaging and Presence Protocol (XMPP): Core. RFC 3920 (Proposed Standard), Oct. 2004.

[11] Saint-Andre, P. Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence. RFC 3921 (Proposed Standard), Oct. 2004.

[12] Saint-Andre, P. JEP-0137: Protocol Design Guidelines. JEP-0137 (Informational), 2004.

[13] Saint-Andre, P., and Hildebrand, J. JEP-0168: Resource Application Priority. JEP-0168 (Experimental,Standards Track), 2005.

[14] Wikipedia. Presence information — wikipedia, the free encyclopedia, 2006. [Online; accessed 26-April-2006].