

Collaboration Software

Clients: Email, IM, Presence, RSS & Collaborative Workspaces Should Be Integrated for Business Communication

Part 1. Strengths & Weaknesses, But Nothing's Perfect



Michael Sampson, michael.sampson@shared-spaces.com
Shared Spaces Research & Consulting
www.shared-spaces.com
August 23, 2004

Revisions

Sep 20, 2004

Page numbering, Misc. spelling and grammar issues

Aug 23, 2004

Original publication

Table of Contents

Table of Contents	3
Executive Summary	4
1. Introduction	6
2. Three Processes for Information Professionals	6
3. The Point Solutions of Today.....	8
3a. Electronic Mail.....	8
Key Services	8
Strengths of Email	8
Weaknesses of Email	9
3b. Instant Messaging	13
Key Services	13
Strengths of Instant Messaging.....	14
Weaknesses of Instant Messaging	14
3c. Presence & Availability	15
Key Services	15
Strengths of Presence.....	17
Weaknesses of Presence	17
3d. RSS Newsreader.....	18
Key Services	18
Strengths of RSS.....	19
Weaknesses of RSS	19
3e. Collaborative Workspaces.....	20
Key Services	20
Strengths of Collaborative Workspaces	21
Weaknesses of Collaborative Workspaces.....	22
4.0 Conclusion: A Collaboration "Super Client" Is Needed.....	23
Shared Capabilities for the Next Collaboration Client	23
Three Processes, One "Super Client"	25
Next Action	25
About Michael Sampson	27
About Shared Spaces Research & Consulting Ltd.....	27

Executive Summary

Of the plethora of separate collaboration clients types on the market today—email, IM, presence, RSS and collaborative workspaces—no single client is sufficient for the communication and collaboration needs of information professionals. A re-integration of the common key services of all of these offerings into a collaboration “super client” is needed.

Information professionals undertake three major processes via a range of collaboration software clients: one-to-one conversations, collaborations around projects, and notifications about changed information.

Each collaboration software client has strengths and weaknesses that means that no single client today meets the needs of today’s information professional:

- *Email*. Strengths are universal addressability, universal interoperability, and cross-organization functionality, offset by weaknesses of a disconnect between its design and current usage pattern, a deluge of spam, and a lack of immediacy, among others.
- *Instant Messaging*. Strengths are real-time attention, real-time transmission, and real-time availability, with weaknesses including no persistence of messages, screen clutter, and an interrupt style.
- *Presence*. Visually displays real-time availability, with weaknesses of poor presence management tools, and the possibility of a disconnect between asserted and actual presence.
- *RSS Newsreader*. Enables push-based notifications of changes and user-controlled subscriptions, with weaknesses of transient data, lack of user management tools, and wrong granularity of information.
- *Collaborative Workspaces*. Strengths are the provision of a set of structured tools for group work, enforced group memory, and shared content taxonomy, with weaknesses of poor cross-platform support, lack of cross-product interoperability, and poor cross-enterprise capabilities.

Many key capabilities of each client type are conceptually shared among the different clients. Users need collaboration client vendors to take the discrete components that are aggregated today into a collection of separate client products, and re-aggregate them into a collaboration “super client”.

Part 2, “Architecture and Key Capabilities of the Super Client” (to be published in late September 2004) will build on the discussion in this paper, outlining a method for aggregating the various components into an overall solution that retains key strengths and resolves the key weaknesses.

What's My Agenda?

Why have I written this paper, and under what circumstances has its been written?

Why? ... To provide a vendor-independent view of collaboration software. I work as an analyst and consultant, and have no product to sell you.

Why? ... To add a voice to the side of rationality in the collaborative software client debate, rather than the side of hype.

Why? ... To formulate and write down a structured piece that I could use in current and future consulting engagements.

How? ... No vendor or individual person asked me to or paid me to write this. It was "funded internally", which means that I wrote it instead of billing clients for current project work.

1. Introduction

Email—the collaboration software client of choice for most people—has been a wild success, with user adoption and message traffic growing rapidly since the mid 1990s. However, what we know as "email" today has some major problems, to which it must adapt or fade from relevancy. In addition, there are a plethora of new software clients that facilitate communication and collaboration, but they remain separate and non-integrated. The purpose of this paper is to return to first principles, discussing the types of software-facilitated interactions the information professional deals with on a day-to-day basis, to review the strengths and weaknesses of today's collaboration client families, and to lay the foundation for a future discussion on unifying collaboration software clients in a new way.

When I wrote the chapter on Electronic Mail for the Encyclopedia of Information Technology in 2000, I pointed out some of the challenges facing email in the coming years. Since then, my research and consulting at Ferris Research and Shared Spaces Research & Consulting have convinced me that it is time for a major change in how we think about and deliver the collaboration software clients during the next decade. Email is not dead, and will not die, but it needs to adapt to the changing requirements of business users around the world.

FUNDAMENTAL THESIS:

Of the plethora of separate collaboration clients types on the market today, no single client is sufficient for the communication and collaboration needs of information professionals. A re-integration of the common key services of all of these offerings into a collaboration "super client" is needed.

2. Three Processes for Information Professionals

Information professionals have to communicate with others, coordinate individual and team action, and leverage the expertise of others to secure the desired outcome for a project. Putting face-to-face interactions aside to focus solely on computer-mediated ones, I see three major processes that are facilitated today by a range of collaboration software clients:

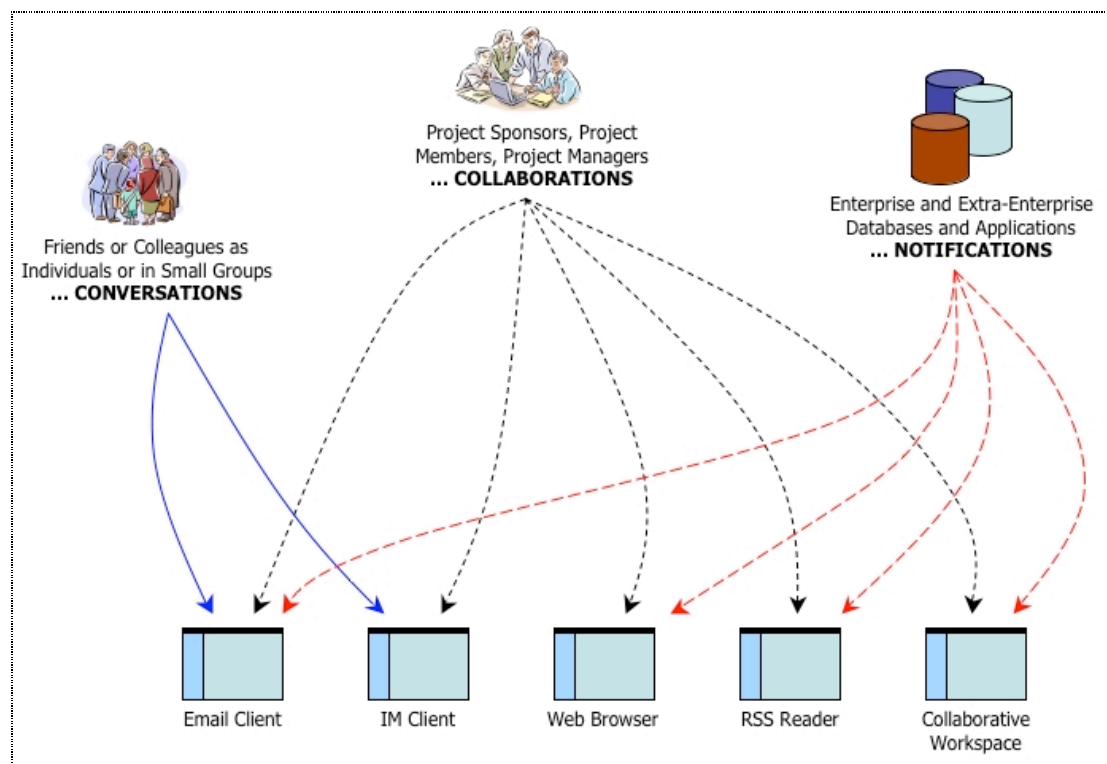
- Conversations. One-to-one conversations with others on a range of topics. These may be general 'passing-the-day' conversations, ad hoc interactions about things of common interest, or quick questions to someone else about their domain of expertise. Conversations are ongoing, and are oriented toward a specific person or group of people. Key computer-mediated tools for conversations are email, instant messaging (IM), presence, and audio/video conferencing.
- Collaborations. Regular, ongoing, and sustained interactions with one or more people about specific projects. There will be a set of conversations about the project, a set of notifications, and usually a definite end date to the

collaboration once the project is completed. Key tools for collaborations are email, IM, weblogs, web conferencing, group audio/video conferencing, file sharing, and collaborative workspaces.

- Notifications or FYIs. These messages could be from another person with a link to report or page of interest, or they could come directly from an enterprise or extra-enterprise database or application based on a profile that subscribes the person to the latest information about certain keywords or topics. Key tools for notifications are email, IM, and RSS.

The problem is that conversations, collaborations and notifications occur through multiple channels of communication, and there is no help to enable the information professional to consolidate the three, manage action across the three, and decide how to proceed.

Figure 1: Multiple Clients Needed for Today's Information Professional



Today's information workers must use multiple non-integrated collaboration software clients for three main processes in their work: conversing with friends or colleagues as individuals or in small groups, collaborating with others on projects that have specific outcomes, schedules, timeframes and deliverables, and receiving notifications about interesting happenings from internal and external data sources.

3. The Point Solutions of Today

Information professionals must use a variety of collaboration software clients to accomplish the three main communication patterns noted above. No single client, however, offers enough functionality to satisfactorily accomplish the three processes, forcing the adoption of multiple offerings. This section considers the strengths and weaknesses of each type of collaboration software client.

3a. Electronic Mail

Since its invention in the 1970s, Internet email has grown in acceptance as a valid way of exchanging messages in business enterprises. During the latter part of the 1990s, the volume of messages some people received grew beyond that which was manageable, into the 70s, 100s, or 200s of daily messages. That is a lot of information to work through each day. Coupled with the explosion of spam volumes since 2000, many in the industry are wondering whether email is dead, dying, or just on pain relief.

Key Services

Today's accepted definition of an email client is one that offers:

- A method of receiving messages into a single place, an "inbox";
- A method of sending messages via an outbox;
- Folders for storing and organizing received and sent messages; and
- An address book of people to whom messages can be sent.
- A rules engine for determining what to do with messages on certain topics, or messages from certain people. Eg, messages with the word "urgent" in the subject line could be put in a special high priority folder, while all emails from a certain named user could be automatically moved to the trash can.

Email clients targeted at the high end of the consumer market and the business market also integrate a task list, a calendar, synchronization options for mobile devices, and various ways of flagging and sorting messages.

Strengths of Email

Email offers a number of major strengths, some of which come out of the technical design of email, and others from the wide adoption of email as a standardized technology:

- *Universal Addressibility*. Internet email uses a common addressing format of [user@domain](#), with centralized-style oversight of who runs a specific domain, and decentralized delegation of the users at a specific domain. Other components of the Internet infrastructure, eg, specific records in the Domain Name System (DNS), enable the interconnection of email servers.
- *Universal Interoperability*. Any Internet email-compliant email server can route messages to any other Internet email-compliant server.
- *Cross-Organization Functionality*. Email works between organizations very easily. A message authored by one person in one organization using an email

client from Vendor A can be read by another person in another organization who is using an email client from Vendor B.

- *Store-and-Forward Delivery.* Messages are routed in a serial fashion between different servers between the originator and the recipient. If one of the servers can not establish an immediate connection with the next server, the message is queued for another attempt at delivery. If it is impossible to deliver the message, the original user is normally notified after some days or hours.
- *Foldering Responsibility with the Individual.* Individual information professionals can sort messages into whatever foldering structure works for them. There is no centralized or enforced foldering taxonomy, thus enabling the user to fit the technology to whatever working style they prefer.
- *Universal Collection Point.* Various other business IT systems can send notifications to a user's email address updating them of the latest changes of relevance to the individual. Internet content publishers use email messages to update subscribers about the publication of new content of interest. This means that the user's inbox becomes a universal collection point of new messages, notifications, and updates.
- *Text-Based Open Protocol.* Messages are sent using plain or formatted text, creating a system with a low level of complexity for interoperability.
- *Multi-Computer Access via the Web Browser.* Webmail was originally introduced as a standalone service from companies such as Hotmail, but the business email vendors added Web client support to their standard email servers. This means that users are not tied to a specific computer, but can roam across multiple computers—at work and at home—to check their email and formulate and send replies.

Weaknesses of Email

Email is under attack on multiple fronts, due to problems fundamental to the open design and the inherent nature of email.

Problem 1. The Metaphor of Usage Has Changed

The original metaphor for email was a letter that was delivered electronically at great speed. A physical letter takes at least a day, if not several, to be transported from the place of its writing to the intended place of its reading. Email overcame this limitation in the physical world, by enabling near-instantaneous delivery of the words, thoughts and feelings of the author to the addressee, wherever they happened to be in physical space. It was expected that people would receive a relatively small number of emails per day, and since the pace of life and business was slower than today, people were willing (or at worst, had no option but to) to wait a couple of days before needing a reply to their letter/memo/email. The technology of email was designed for this original metaphor.

Eric Hahn, founder of Proofpoint
"In 1978, it was expected that each user would
receive a maximum of five messages a day"
INBOX Conference, June 2004

An apt metaphor for today's actual usage of email is that of a text fragment from a voice conversation. Instead of calling to speak with the other party—who is likely to be busy, unavailable, or otherwise engaged in other matters—people send an email outlining one or two sentences out of the overall communication. The recipient replies with the next couple of sentences, and so on. This creates in essence a series of conversation fragments, each sent in response to a conversation fragment from the other, and each spread over the course of multiple minutes, hours, days or weeks. In a one-to-one communication situation, the two parties can maintain some sense of communication coherence, but once the exchange expands to three or more people, the fragmentation of thought-response-counter-response becomes unmanageable. Finally, individuals have multiple ongoing conversations with multiple people covering multiple projects, leading to a no-win situation via the use of email. The technology of email was not designed for this usage metaphor. Today's users want to engage in real-time communications with others via email ... a use case that the technology was not designed to support.

Problem 2. Deluge of Spam

Depending on which research firm or anti-spam vendor you listen to, the percentage of spam as a proportion of overall email traffic is in the 50-95% range. Those that send spam have leveraged the openness, universal addressability, and zero cost of delivery design characteristics of email in an abusive way that was never intended by the original technical architects. Assuming that all messages—valid emails and spam—get through to a user's inbox, that means that 50 to 95 messages out of every 100 are not helpful to the user in getting their job done. The user has to plow through the overwhelming volume of spam just to find the 5 to 50 messages that are valid and helpful.

Anti-spam vendors alleviate the problem, whilst simultaneously creating new problems. On the positive side, anti-spam software blocks inappropriate messages from getting through, thus reducing inbox clutter. On the negative side, if the anti-spam filter is too forceful, it can wrongly classify messages as spam, thus reducing the value of email since valid and important messages don't get through. Either the information professional must periodically review the list of messages marked as spam to retrieve those that were inappropriately tagged, or the sender must attempt to establish communication again, along with feeling frustrated at the lack of response from the desired recipient. It is definitely an area for fostering misunderstanding and inter-personal frustration.

It is my belief that if there was no spam or that the problem of spam was completely resolved, email would still be insufficient for the communication needs of information professionals. Spam definitely highlights and exacerbates the problems of email, but it isn't enough to kill email by itself.

Problem 3. Lack of Immediacy

Email messages are delivered through an interconnected network of relay and mailbox servers, where each server on the path between the sender and final

recipient captures the message in full before attempting to pass it on to the next destination in the message route. As such, the "store-and-forward" nature of email does not lend itself to those occasions where an instantaneous reply is required. If the intended recipient only has email, the sender must revert to alternative means of ensuring a quick response is achieved, most likely through a follow-on telephone call to the effect that "I've just sent you an email, and would like your reply immediately." For those cases when an immediate reply is needed, the technical design of email fails the user.

Problem 4. Torrent of Email

Individuals involved in multiple projects, those with a public persona, and those with managerial responsibility across people and projects can become overwhelmed with the torrent of incoming messages. There are a myriad of reasons for this: the conversation-fragment usage style of email, unclear lines of responsibility and levels of accountability for specific project outcomes, and trying to stay on top of new content at too many publications or Web sites.

There are two levels at which a resolution to this must be focused: firstly, the development of good email hygiene habits (eg, responding quickly to "conversation fragments", keeping the inbox clear of messages that have been dealt with, and unsubscribing from low-value notifications re Internet content), and secondly, the examination of what messages can be eliminated at their source through cultural changes, eg, clear definition of what types of messages someone needs to be sent or copied on, a team-based decision on how to use email effectively, regular face-to-face meetings or teleconferences to have the conversations in real-time, and alternative technologies that are more appropriate for certain types of messages, eg, shifting conversations out of email into collaborative workspaces.

Problem 5. Lack of Content Security

Email messages are usually sent in plain text between email servers, meaning that messages can be easily intercepted and read via easy-to-obtain software programs. Whilst the impact of such interception ranges from low to irrelevant for mass-customized update notifications from Web sites, the cost is potentially huge for confidential message traffic related to business plans, key strategies, and new acquisition targets. Individuals are unaware of the risks they face in sending email messages, inherent in the technology of email due to the lack of message encryption. There are a variety of encryption options on the market, but none have achieved widespread popular adoption.

Problem 6. Lack of Shared Taxonomy

Individuals can organize their collection of messages in any way they want, eg, via a nested hierarchy of folders for specific clients or projects, via categories assigned to individual messages, or via reliance on full-text search across all messages in the mailbox. While this extreme flexibility makes email easy to embrace in the first instance, in the context of group work facilitated by email, no one will store their own copy of the messages in the same way. This is problematic in a number of

scenarios:

1. A project member rings to discuss a recent email with another member of the project team. The second member can not remember where they filed the email message (or if they merely deleted it). If there was shared taxonomy enforced across all members of the project, the first could say "it is the third message from the top in folder Project A1".
2. Once an individual starts down a specific organizing track, it is time consuming to embrace an alternative method. Eg, if messages are sorted into folders by client, the user has to go through each folder and each message in turn to implement a new "by project" classification.
3. Email doesn't provide a coherent group memory. It is difficult to see the complete history of a group interaction by looking at one person's email inbox ... some messages will be in the Sent file, and others will be spread across different email folders. In addition, the user may have deleted some of the messages related to the project at hand.

Problem 7. Transmission of Viruses and Worms

Microsoft Outlook is one of the two most widely used email clients in the business market, and its inadequate security model and the widespread embrace of the Microsoft desktop monoculture has resulted in the wide transmission of viruses and worms. The Melissa virus of 1999 and the ILOVEYOU virus of 2000 were estimated to have caused billions of dollars of damage, mainly through lost productivity of information professionals. More recent versions of Outlook—XP and 2003—have resolved many of the virus vulnerability issues, but a wide swath of the market continues to use the older versions.

Problem 8. Lack of Real-Time Tools

Email is only store-and-forward, with no option to resort to real-time sharing of information. While messages are often delivered very quickly—in the order of a couple of seconds—between email servers, the potential for bottlenecks, failed links, and the false classification of messages as spam by intermediate servers can push the delivery times out to a couple of minutes or worse. If two people want to share a specific piece of text-based information by email while they are talking on the phone, they have to hope that the intermediate delivery paths are well-kept and functioning optimally. There is no guarantee of fast delivery, and once the email has left the user's email client, they essentially have no method of seeing where a specific message is and why it has not yet been delivered.

Problem 9. Storage is Siloed in the Email Client

Messages stored in the email client are stored separately from spreadsheets, documents and other files on the user's computer. In the Windows world, for example, the user has an email client with an associated data store (a .PST file for Outlook or a .NSF file for Notes) as well as their "My Documents" folder on the hard drive. There is no easy or transparent way for mixing-and-matching related messages in the mail store with documents in the document store.

Problem 10. Attachment Distribution is Unreliable: Blocking, Stripping, Quotas

Sending documents by email to another person or a group is fraught with challenges. The document may be blocked from leaving an organization, may be stripped from the email before entering the recipient's organization, or may be rejected by the user's mailbox due to overreaching the user's size quota. Some organizations inform users that an attachment has been blocked or stripped, but in the situations where they don't, the two users are left to their own devices to find out about the problem, and resolve it using alternative means.

Problem 11. Expiration of Email Addresses

Research companies have estimated that 50-80% of email addresses change each year. Without a centrally coordinated mechanism of informing and enforcing address changes, users can "fall off the face of the email earth", nullifying ongoing communication options.

3b. Instant Messaging

Instant messaging establishes a bi-directional real-time communications session between two or more people, enabling the simultaneous transmission of text from one user's computer to the screen of another, and vice versa.

Instant messaging was introduced to the world at large via the ICQ ("I Seek You") system from Mirabilis, Inc. in 1996. AOL acquired the company, and introduced a sister service (AOL Instant Messenger). Microsoft entered the consumer market in April 1997 (with MSN Messenger), followed by Yahoo some months later with Yahoo Messenger. On the business side, Lotus was an early entrant, introducing Lotus Sametime (a re-development of the technology it acquired through the purchase of Ubuque, Inc.) giving it an early lead that it has held to this day. Microsoft introduced an IM product alongside Exchange 2000 Server (based on technology acquired from Flash Communications), and came out with a re-architected standalone business IM offering in October 2003.

Key Services

Instant messaging offers the following key services:

1. The sending and receiving of text messages in real-time between two or more people;
2. A list of contacts that the individual can communicate with via instant messaging. The contact list is enhanced with presence and availability indicators: presence is a binary choice of present or not, and availability gives added flavor to the presence setting, eg, if the person is present, they may have an availability setting of "Busy, on the Phone", "Ready to Talk", or "Writing a Report ... Urgent Enquiries Only";
3. A way of displaying new messages, most usually through a pop-up window on the computer display. Some IM clients assign a new tab in the IM client to

- each new message; and
4. A way of setting who can and can not send instant messages to the user. IM clients are usually 'open to all' by default, but give the user the ability to establish their own closed network of permitted contacts.

Strengths of Instant Messaging

IM manifests a number of strengths as a collaboration client:

1. *Real-Time Attention*. In a manner akin to a phone call, both parties to an IM can give their real-time undivided attention to the text messages of the other person. This means that issues can be solved immediately, rather than waiting for a reply to an email. In reality, once users become used to IM, it is not difficult to have two, three or more conversations going in real time.
2. *Real-Time Transmission*. Messages appear immediately on the user's screen, or not at all. Users know if they have the attention of the remote party
3. *Real-Time Availability*. The IM client displays the status of the remote party, meaning that it is immediately clear whether the person is available for interaction or not. While the possibility remains that the remote person is away from their computer, the person wanting to initiate a communication can have a reasonable degree of confidence that the other person is there and available to talk.

Weaknesses of Instant Messaging

As with email, IM has a number of weaknesses:

1. *No Persistence of Messages*. Many IM clients do not save the transcript of an IM conversation once it is completed. That means the user loses any history of what was discussed.
2. *Excludes Off-Line People*. IM clients enable a user to interact with people who are currently online and available, but many have no support for interacting with offline people. Users must switch to their email client to send a message to an offline person, although some clients—such as the MSN Messenger client—enable users to send an email from directly within the IM client. Perhaps the concept of a "queued IM" would be beneficial, so that a sent message is delivered as soon as the other person comes online.
3. *Screen Clutter*. If users have multiple IM sessions underway, they have to keep switching between the various conversation windows. On smaller screens, that's a pain. In addition, if the user is trying to work on a word processing document or spreadsheet at the same time, the IM conversation window or windows overlap with the document or spreadsheet.
4. *Interrupt Style*. IM gives other people the tools to inflict interruptions on others. While a user is thinking about writing a document, for example, a request to join an IM conversation could pop-up on the screen, breaking the writer's train of thought. The user could obviously turn off the IM client when working on such things, but may want to remain accessible to certain key people. IM clients need the sense of "selective availability", whereby users can nominate who can interrupt them via IM at any given point in time. Eg,

an “open to all” setting would allow anyone to send in an IM, whereas an “open to team” setting would only permit IMs from other team members, and queue the rest for subsequent delivery.

5. *Reduced Efficiency Due to Slow Typing.* It is faster to have a phone conversation with someone that do the entire communication via IM. Many people can't type very fast, and so users wait to see what the other person is going to say in their latest missive. IM is okay for initiating a given conversation, but alternative communication channels better serve many weightier or in-depth issues. IM can actually decrease productivity if the user is constantly waiting for the other person to type their thoughts.
6. *Possibility for Confused Message Stream.* Given that some users type slowly, the point-counterpoint stream in an IM transcript can become confused, whereby one person is responding to a point that the other has moved away from. It becomes confusing, therefore, to marry up what one person has said in response to any given point.
7. *Conversations are Siloed in the IM Client.* If the IM client does actually store the transcript of the conversation, the transcripts are saved in the IM message store, which is separate from everything else that the user is working on.
8. *Poor Multi-Computer Support.* IM is difficult to use from multiple computers. Firstly, many IM services only permit the user to log in to the service from a single computer, with AOL being a recent exception to this. That means that if the user is logged in at work, and then logs in at home, the work connection is logged out. Secondly, if the user is logged in at work and then the user logs in at home, any invitations to a conversation or subsequent responses to current conversations that display on the work computer are not transferred to the home computer.

A couple of messaging thought leaders are so fed-up with the problems of email, that they have proposed the dropping of email in favor of “instant messaging-like closed networks”. In other words, rather than sending email messages to others, users would engage in real-time IM exchanges. The idea has some attractiveness, given the strengths noted above, but is unworkable as a total solution due to the weaknesses I have outlined.

3c. Presence & Availability

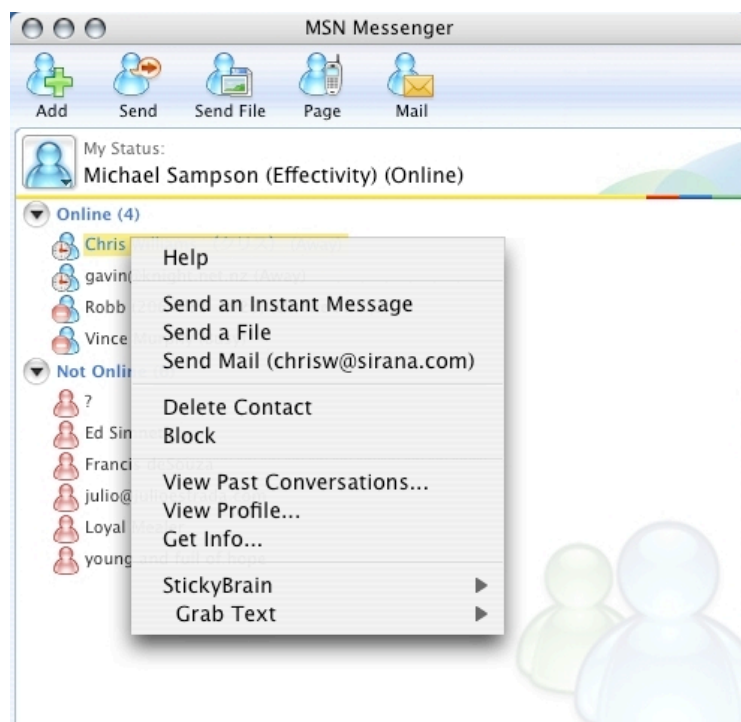
Presence and availability was introduced to the world at large via IM clients, but has much wider applicability for use in other collaboration software. Presence and availability indicators elicit a yawn unless they are embedded into collaboration software clients.

Key Services

Presence offers the following key services:

- Presence Indicator and Control Panel. The ability for a user to set whether they are “present” or “not present”, and to have this displayed to everyone else on the IM service that wants to communicate with them.
- Availability Indicator and Control Panel. The ability for a user to set a finer level of granularity on their presence—such as “available”, “on the phone”, or “ready to talk”—or their non-presence, eg, “Out to lunch”, “Back at 2pm”, or “At the doctor”.
- List of Possible Communication Mechanisms. A given person can be contacted through a variety of communication channels, such as via email, via IM, via a phone call, or via text message. If the user knows about all of those addresses for a given person, they should be shown. See Figure 2.

Figure 2: Possible Communication Mechanisms



Michael can interact with Chris through two communication channels: via instant messaging, and via email. If Michael also knew Chris’s mobile phone number, he could send a text message.

- Privacy Rules. Privacy rules allow for the user to set who is able to view their presence information. Settings range from “Allow everyone” to “Deny everyone”, with a number of levels in between. See Figure 3.

Figure 3: Privacy Rules for Presence in iChat



The Privacy Level in the iChat IM client is currently set to "Allow Anyone", which means that any other person logged into AOL Messenger can see this user's presence information.

Strengths of Presence

Presence and availability offer one key strength:

1. *Visual Notification of Availability*. When people work in different offices, the presence and availability indicator gives some degree of insight into whether that person is in or not. That's often useful for a person when deciding how best to initiate a conversation.

Weaknesses of Presence

The weaknesses of presence and availability are:

1. *Not Actionable By Itself*. Presence and availability is a technology that offers little value unless it is integrated into collaboration software. By itself, presence and availability only indicates the state of a person.
2. *Poor Ability to Manage Presence*. User's are not given capabilities to see who is tracking their presence. Tools are given to block specific individuals from seeing and tracking presence, but tools are not given to see who actually is tracking presence.
3. *Only a Proxy for True Presence*. Presence is set by a user indicating their presence and availability via a status box. However, users can say they are present but actually not be, eg, user's that leave their computers on overnight with a presence setting of "available". That provides a disconnect between asserted and actual presence, which can confuse users ("she says she's there, but why isn't she responding to me?"), and will become more of a problem once presence-based routing of queries and requests is built into

business applications.

3d. RSS Newsreader

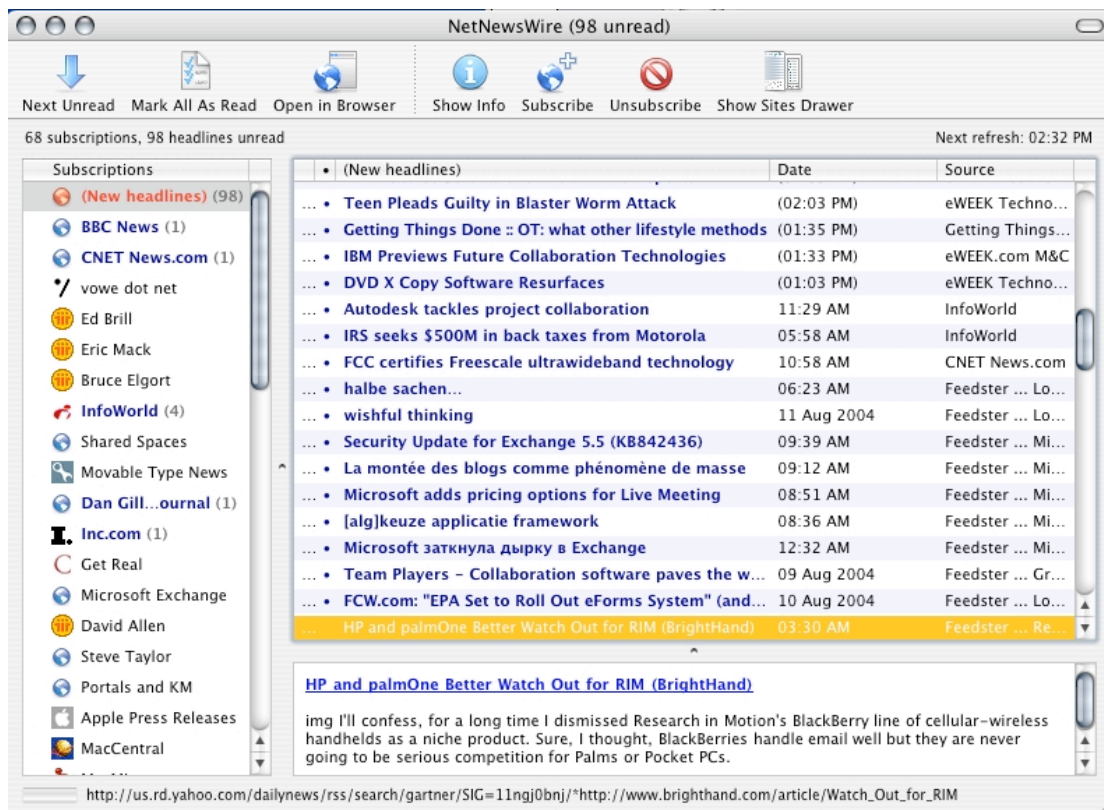
RSS (Really Simple Syndication or a couple of other terms) is a pull-based technology for providing content from Web sites to readers. Instead of the user having to visit a Web site to view changed information, the Web site administrator configures the site to keep a special changed-items page up-to-date. This page is structured using the XML markup language, and to view the changes the user tells their newsreader to periodically poll the remote Web server for the latest version of the page. If there are changes noted in the page, the user's newsreader displays the latest version of the page, highlighting those components of the page that have changed since last time.

Key Services

The key services of an RSS newsreader are:

- A subscriptions manager for maintaining a list of active subscriptions. This means that the user enters the address of the RSS page into their newsreader as a valid and current subscription.
- An inbox for displaying the content of the pages that the user has subscribed to. A given page may contain 10 or 20 different items, which are then rendered as separate headlines, along with an extract (or the full article) in a preview pane. See Figure 4.

Figure 4: The NetNewsWire RSS Client



NetNewsWire is an RSS client. Users subscribe to RSS pages of interest—show down the left hand column—which then display in the top pane on the right hand side. A small preview pane shows the content of the item.

Strengths of RSS

RSS is a very useful technology for use in computer-mediated communication, as follows:

1. Automatic Pull-Based Notification of Changes. Users do not need to visit the Web site in question to learn if something new has been posted. Rather, the RSS page for the Web site will be updated automatically, and the user's RSS newsreader will highlight those new and changed items.
2. User Ownership of the Subscription. The user has total control over whether they pull-down the latest version of the RSS page or not. If the page no longer offers information of value, the user can delete the subscription from the subscriptions manager.

Weaknesses of RSS

There are some major weaknesses with RSS:

1. Transient Data. RSS newsreaders display whatever the current page says. There is no concept of saving the separate fragments of the syndicated page,

- because the page remains under the control of a foreign party.
2. *Lack of User Management Tools*. RSS newsreaders lack the concept of "forwarding" a separate fragment of the page to a friend or colleague.
 3. *Granularity of Information is Wrong*. The user subscribes to the entire RSS page as a whole, not specific information contained therein. The user is therefore subject to whatever the feed owner wants to include in the page. One way around this is to subscribe to specific keywords or phrases through Feedster—or a similar service—which indexes hundreds of thousands of RSS feeds from the Web, and which returns those keywords in an integrated feed for the user.
 4. *Adds to Information Overload*. There's a lot going on around the Web, with lots of people writing material on topics of interest. While users will meet people that they wouldn't otherwise have met, it requires a lot of self-control to hold back the flood of new information that will present itself on a daily basis due to subscriptions to RSS-enabled sites.
 5. *Lack of a Rules Engine*. While users have the ability to subscribe to an RSS feed in its entirety, they lack the ability to granularize their subscription. For example, users should be able to subscribe to an RSS page and enter certain keywords that reflect those entries in the page that they actually want to see. There are no constructs in a newsreader that would enable a user to block certain content from displaying.
 6. *Duplicated Entries*. Newsreaders lack the ability to eliminate duplicated entries in the material displayed. For example, if the user has multiple subscriptions to keywords at Feedster, and the same item is returned under multiple keyword options, the newsreader can't interpret those and present a duplicate-free listing. This is not to say it won't be done, but it hasn't been implemented into newsreader clients yet.
 7. *No Personalization Options*. RSS is a one-to-many push-based thing at the moment, with many people able to subscribe to the same page of information.

3e. Collaborative Workspaces

A collaborative workspace provides a set of structured tools that enable multiple people to work together. Just as email clients are composed of standardized components (inbox, folders, contacts, and rules) meant to enhance individual productivity and communication, a collaborative workspace focuses squarely on team productivity and communication.

Key Services

Key services delivered in a collaborative workspace are:

- A method of contributing and viewing pages of text or pictures that is shared among multiple people. Different collaborative workspace products implement this shared contribution model at different levels; for some, the level of sharing is a single discrete contribution that no one else can alter (eg, a comment on a Weblog or a post in a discussion database), or alternatively a totally open shared editing model whereby anyone can edit what previous

- contributors have written (eg, in a wiki, or in an open Notes database).
- **Workspace-based Presence.** People that are visiting the collaborative workspace can see via an onscreen list who else is currently visiting the collaborative workspace, and can engage in on-screen text messaging with that person, or a group of people.
- **Workspace-based Shared Files.** People can deposit a file (a Word document, an Excel spreadsheet, a photograph, a PowerPoint presentation, etc) into a file folder that other people can also access to read or perhaps to change.
- **Screen Sharing.** People can share their desktop screen with other people in the collaborative workspace. This means that someone else can see what applications the first person is running, and can view what the other person is typing or doing. When combined with a VoIP or telephone call, this provides multiple simultaneous ways for people to work jointly ("What do you think of this paragraph?" "It looks fine to me, except for line 2", etc).
- **An access rights manager for controlling which people have which rights in the collaborative workspace.** Some people will have top-level rights to edit any element of content, whereas others will have lower rights to edit only certain pieces of content, and others still will only be able to read content. For example, in a Weblog, the owner has the top-level right to edit/write new blog entries and comments, whereas (if permitted by the owner) everyone else can post a comment if they want, but cannot edit what the owner has written.
- **It may offer a variety of collaboration-enablement/tracking/reminding tools.** For example, people can use a shared calendar for booking meetings, a shared to-do list for tracking who has agreed to complete specific items of next action, alerts for when new content has been posted to the collaborative workspace, and workflow tools for routing items to another person when a specific trigger point is reached, eg, one member of the collaborative workspace finishes a task.
- **Methods to integrate with existing enterprise databases and applications.** Eg, the shared calendar tool may integrate with the calendaring system on Notes/Domino or Microsoft Exchange.
- **A rules engine for requesting notification when certain changes are made in the collaborative workspace,** eg, a new person is added, or a document is updated.

Strengths of Collaborative Workspaces

The strengths of collaborative workspaces are:

1. **Structured Tools for Group Work.** Collaborative workspaces deliver structured tools for shared activities. These tools introduce a degree of structure into the way the group or team works together, and calls different things by their proper names. For example, rather than email message being used to share the contact details for a group member, these are written into a contact card.
2. **Enforced Group Memory.** All of the work of the team is stored in a centralized place, which gives users access from multiple computers, and takes the responsibility for maintaining a history of the group work away from any one individual.

3. *Shared Content Taxonomy*. Since users are accessing the same collaborative workspace, its taxonomy is shared among the users. This means that everyone sees the historical messages or stored files sorted and presented in the same or similar ways.
4. *Invite-Only Access to Minimize Spam*. Collaborative workspaces are usually on an invite-only basis, meaning that they are immune from unauthorized access and uninvited messages. A user who abuses their access rights can be banned from future access.

Weaknesses of Collaborative Workspaces

The weaknesses of collaborative workspaces are:

1. *Forced Trade-Off Between Cross-Platform Support and Offline Access*. Today's products aren't sufficiently mature to offer cross-platform support and offline usage. Organizations today have to choose between these two essential elements: either embrace a collaborative workspace product that works on a single platform and offers offline capabilities (eg, Groove on Microsoft Windows), or gain Web browser access from multiple operating systems but without offline support.
2. *Lack of Cross-Product Interoperability*. Users are tied to the specific collaborative workspace product that has been embraced for a project. In other words, if it is decided that the project will be done with IBM Lotus QuickPlace, everyone has to use QuickPlace. Users do not have the choice of being involved in the project but using their collaborative workspace client of choice, eg, Groove, eRoom, or Windows SharePoint Services. This means that users involved in multiple projects will have to learn how to use multiple collaborative workspace offerings. This is a huge barrier to adoption.
3. *Poor Cross-Enterprise Story*. Due to the lack of cross-product interoperability, it is difficult to make collaborative workspaces function appropriately across enterprises. Everyone has to use the same product, which means licensing costs and procedures, installation costs and procedures, and authorization procedures.

4.0 Conclusion: A Collaboration "Super Client" Is Needed

Shared Capabilities for the Next Collaboration Client

Users need collaboration client vendors to take the discrete components that are aggregated today into a collection of separate and different client products, and re-aggregate them into a collaboration "super client" for tomorrow. Figure 5 depicts the capabilities that are shared across the various clients of today, and thus points of similarity for the collaboration "super client".

Stowe Boyd, Corante Research

*"The next-generation of converged social tools
should offer presence-based real-time
communications plus blog-based community
focused solutions"*

Get Real, June 2004

Notes on the Shared Capabilities matrix:

1. Items in the RSS page are sorted into folders if you call the feed a folder. For Feedster pages, where the pages are on a specific topic, the feed folder is essentially akin to a "by topic" foldering structure in email.

Figure 5: Shared Capabilities

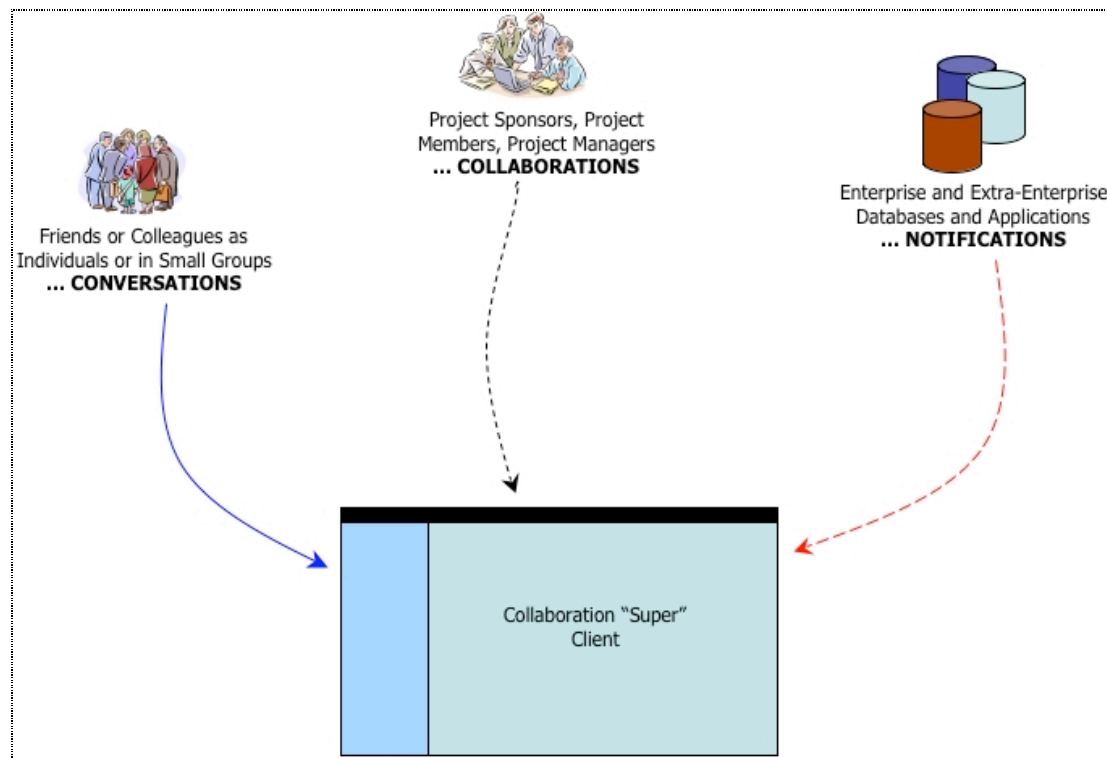
Capability	Email	Instant Messaging	Presence	RSS News Reader	Collaborative Workspace
New Message Display	Inbox	Pop-up Windows		Inbox	Inbox of all latest changes
Nature of Messages	Stored until Deleted	Deleted when Finished		Stored until underlying feed page changes	Stored until deleted by authorized person
Message Distribution	Yes, via the Outbox	Yes, sent immediately			Yes, via a discussion forum or workspace-based email message
Folders for Sorting	Yes			Yes (1)	
Rules Engine	Yes			No, but needs one	Yes, for alerts
Access Manager	Yes, via sender blocking rules	Yes, via sender blocking rules		Yes, via active subscriptions	Yes, for workspace access
Address Book	Of contacts	Of contacts that are present		Of RSS page subscriptions	Of contacts
Calendar	Individual, with access to group calendars				Workspace-based shared calendar
Task List	Individual, and delegated				Workspace-based shared task list
Communication profile options			Yes, based on directory or address book entries		Yes, in the person's group profile document
Presence	Depends on the client	Yes			Workspace-based presence
Shared Files	No, but the ability to send	No, but the ability to send			Yes, in a repository with shared access
Shared Screen Capabilities					Depends on the product

Many of the capabilities in today's collection of collaboration clients are similar in intent, albeit different in maturity. Common capabilities should be merged into a collaboration "super client".

Three Processes, One "Super Client"

Early in this paper I outlined the three interaction processes that information professionals are involved in—conversations with friends or colleagues as individuals or in small groups, collaborations with people on specific projects, and notifications from enterprise and extra-enterprise databases and applications. With a collaboration "super client" that integrates the capabilities outlined above, users has a single client that facilitates and enables them to manage the processes they are involved with on a day-to-day basis, in a coherent and cohesive manner. See Figure 6.

Figure 6: An Integrated Collaboration "Super Client" With Shared Capabilities



Tomorrow's information workers should have a single collaboration "super client" that integrates all the functionality needed for the three main processes in their work: conversing with friends or colleagues as individuals or in small groups, collaborating with others on projects that have specific outcomes, schedules, timeframes and deliverables, and receiving notifications about interesting happenings from internal and external data sources.

Next Action

Part 2 of this argument, "*Architecture and Key Capabilities of the Super Client*" will be published in late October 2004. It builds on the analysis presented herein regarding the strengths and weaknesses of today's collection of separate software clients, and outlines an architecture for developing a re-integrated super client for the next-generation of collaboration.

Between now and then, if you want to discuss any of the points that I've raised above, or to discuss your ideas and perspectives on this topic, please get in contact by email (michael.sampson@shared-spaces.com), or leave a comment at www.shared-spaces.com.

About Michael Sampson

Michael Sampson has been an active researcher, analyst and consultant in the messaging and collaboration market since 1994, working with an international client base in the United States, the United Kingdom, Europe, Australia, and New Zealand, his country of residence. Michael is passionate about helping businesses and government agencies leverage the power of collaborative technologies in the context of their day-to-day work processes for business success and outstanding government results.

Michael established Shared Spaces Research & Consulting to work closely with interesting businesses and government agencies working to develop an all-encompassing collaboration infrastructure by leveraging the power of shared spaces to overcome the limitations of today's inadequate individual communication technologies.

Contact Michael at michael.sampson@shared-spaces.com, or phone +64 3 317 9484.

About Shared Spaces Research & Consulting Ltd

Shared Spaces Research & Consulting fulfils two complementary objectives in the market: to publish thoughtful self-funded vendor-neutral independent research on how businesses and government agencies can leverage the power of shared spaces for enterprise collaboration, and to provide consulting services on shared spaces. Key technologies explored include:

- Collaborative Team Workspaces
- Real-Time Interaction Technologies
- Collaborative Business Portals
- Presence & Availability in Business Applications
- Wireless Collaboration & Messaging
- Collaboration Auto-Discovery.

Visit www.shared-spaces.com for daily coverage of interesting happenings in the messaging and collaboration industry.