

COVINGTON & BURLING

ORIGINAL

1201 PENNSYLVANIA AVENUE NW WASHINGTON, DC  
WASHINGTON, DC 20004-2401 NEW YORK  
TEL 202.662.6000 LONDON  
FAX 202.662.6291 BRUSSELS  
WWW.COV.COM SAN FRANCISCO

ERIN M. EGAN  
TEL 202.662.5145  
FAX 202.778.5145  
EEGAN@COV.COM

October 3, 2000

BY HAND

RECEIVED

NOV 14 2000

00:30

FEDERAL COMMUNICATIONS COMMISSION  
OFFICE OF THE SECRETARY

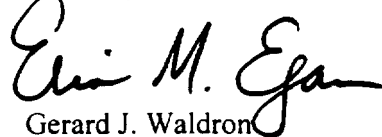
Ms. Magalie Roman Salas  
Secretary  
Federal Communications Commission  
445 – 12<sup>th</sup> Street, S.W. – The Portals  
TW-B204  
Washington, D.C. 20554

Dear Ms. Salas:

Attached is a paper that addresses some technical questions on IM interoperability that arose at a meeting with the Cable Services Bureau and representatives from iCast, Tribal Voice, Excite@Home, AT&T and Microsoft.

Should you have any questions concerning the attached, please contact the undersigned.

Sincerely,



Gerard J. Waldron  
Erin M. Egan

Attachment

- cc: Chairman William E. Kennard
- Commissioner Susan Ness
- Commissioner Harold W. Furchtgott-Roth
- Commissioner Michael K. Powell
- Commissioner Gloria Tristani
- Ms. Deborah Lathen
- Mr. Bill Johnson
- Ms. Royce Dickens
- Mr. Darryl Cooper
- Ms. Linda Senecal
- Mr. Andy Wise
- Ms. Nancy Stevenson
- Mr. John Berresford
- Mr. Doug Sicker
- Mr. Michael Kende
- Dr. Robert Pepper
- Mr. Jim Bird

No. of Copies rec'd 2  
List A B C D E

# The Basics of Instant Messaging Interoperability

*This paper presents the basics of Instant Messaging interoperability and discusses the differences between server-to-server and client-to-client interoperability. The first part sets forth the logistics of logging on to an IM system and how that system would work with interoperability. The paper next explains how new services, such as instant news or entertainment services, will be provided in the "buddy" format, subject as always to the control of the customer. The next part contrasts an interoperable system with the operation of a system with no interoperability and analyzes the difficulty suffered by consumers where there is a lack of client interoperability. The paper then reviews client-side interoperability and demonstrates that, by definition, client-side interoperability is just as secure or private (no more and no less) than no-interoperability. Moreover, these basic types of interoperability do not require the sharing of intellectual property – only an understanding of what protocol is being used by each IMSP. The paper finally concludes that server-to-server interoperability may also avoid the need to run multiple clients for multiple services, but it is exponentially more complex without yielding concomitant benefits on privacy and security.*

## ***Service Logon and Presence – Single System***

### Establishing an Account

Each IM service provider ("IMSP") requires users to open an account to use its service. In order to establish an account, a user must choose a login name (or username) and a password, both of which are required for authentication purposes. The account may be subscription based, or it may be provided without charge. To use a typical free IM service, the user must go to the IMSP's Web site, download any necessary client-software (hereafter "IMSP client"), and then open an account (choosing a username and password).

### Logging In

When users want to use an IM service, they need to log on and provide their chosen username and password. This username and password information allows the IMSP to authenticate a particular user logging in as the owner of the account. For security reasons, the username and password themselves actually are not sent to the IMSP. Rather, the IMSP client exchanges with the IMSP information derived from the username and password. This is a standard technique for securely validating a user's credentials or authenticity. If the user cannot provide the proper combination of username and password, the IMSP will deny the user access to the IM service.

### Presence

Once the user is logged in, the user establishes a "presence" relative to the IMSP. This presence feature allows a user to appear to other users of the same IMSP. IMSPs allow the user to create different presence types, such as "online," "busy" or "away." A user also can decide to watch the presence information of "buddies," if permitted by the buddies' privacy settings. (Note that AOL is one of the few IM providers which permits its users to monitor the presence of "buddies" without their knowledge.) If a user can detect the presence of his or her buddies, the user will receive notifications whenever a buddy logs in, logs out or changes presence information.

## Messaging

Once a presence is established, a user may begin to send instant messages to other users. The destination user is typically a pre-selected buddy – that is, someone with whom the sender has agreed to exchange instant messages. When a message is delivered, the recipient will see the message instantly appear on his or her device’s screen, or at least the recipient will get instant notification that a message has arrived (e.g., a pager might beep when it receives an instant message). Depending on system capabilities, the message can be a simple text message, a picture, a video or a sound file.

It is important to note that as IMSPs begin to foster the delivery of additional services, *a provider of an additional service would be considered a buddy*. For instance, a provider of instant news-update services would be a buddy.

Some IMSPs also allow the user to send instant messages to people who may not be on the user’s “buddy list,” including, even, a random user. In these cases, based on the IMSP’s *features and the privacy setting of both the receiver and the sender*, the message may or may not be delivered. Other IMSPs may ask a destination user if he or she wants to become a buddy of the sender before allowing the sender to engage in instant messaging with the receiver.

## Privacy Settings

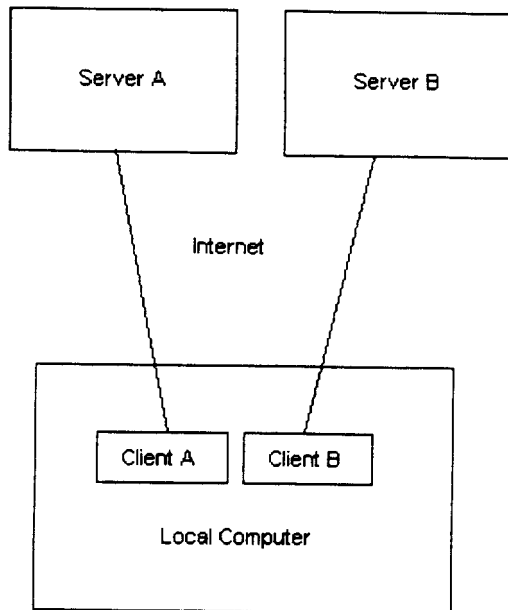
Though all IM providers share certain basic privacy features, the nature of privacy protections offered by IMSPs differs. IMSPs offer their users different mechanisms for protecting their privacy, such as preventing third parties from seeing a user’s presence if the user does not want them to and blocking third parties from sending that user instant messages. The default settings also can differ from IMSP to IMSP. At a minimum, users can *configure* their privacy settings to control which other users are allowed to monitor their presence information and which users can send them messages. At the other end of the scale, users can configure their settings so that all users, except those put on a “banned” list, can detect their presence and send them messages.

### ***Without Interoperability***

In the current environment, if users want to talk to their buddies on different services, users need separate IMSP software running on their desktop device for each service provider. This means that users typically have to download and set up multiple IMSP client software programs. Users also need to open accounts and to establish usernames and passwords with each IMSP. It works as follows: (1) each instant messaging client connects to its respective service provider; (2) users log in individually to each IM service, having to remember their different login and password combinations; (3) users set up their buddies for each IMSP client/service; (4) users establish presence on the respective IMSPs; and (5) users send and receive instant messages.

If the two IMSP clients cannot speak to each other (i.e., they are not interoperable), a user cannot simultaneously send instant messages to buddies who are on different systems. The user will be able to send instant messages within the separate communities, but not across them. In addition, in the no-interoperability case, the user must remember which of his or her buddies are on which IMSP and must use the appropriate IMSP client to communicate with those buddies. Also, as the user tries to add buddies, the user and the prospective buddies must agree on which IMSP to use.

### No Interoperability Case



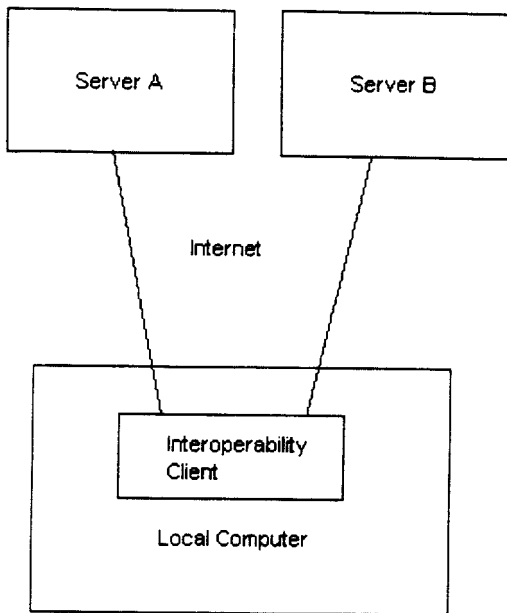
### ***Client Side Interoperability***

Where IMSP client software is interoperable (i.e., in the above diagram, Client A can interoperate with Client B), many of these obstacles can be eliminated. The first four steps of the process would work in the same way described above: each IMSP client would connect to its respective service provider and transmit the users' username and password to log in and each IMSP client would establish the user's presence on its respective IMSP. However, where the clients are interoperable, the user would be able to select which IMSP client to use to send and receive messages and to view presence information.

Although this certainly improves upon the process that currently exists in the non-interoperability environment, the user experience with this type of functional inter-exchange of instant messages remains limited. Users still must have IMSP client software running on their computer or other device for each service on which they wish to have, and to receive, the presence of their buddies. This means that a user typically has to download and set up multiple programs.

This process can be simplified by ensuring that multiple IMSPs use compatible communications protocols at their respective servers. In such a scenario, each IMSP will be responsible for exactly the same transactions (logon and presence). The difference will be that a single IMSP client may perform these transactions for multiple IMSPs, rather than requiring an IMSP client for each IMSP. The only change in this scenario occurs on the user's local computer. Instead of many IMSP clients connecting to one server each, a single client establishes connections to multiple IM servers.

### Interoperable Client



In order to establish these connections, the interoperability client needs to know the username and password for each IMSP on which the user has an account, just as each client needs to know the username and password for its respective IMSP today. But as far as each IMSP is concerned, the exact same logon scenario is taking place. The only difference is that one client is executing this scenario multiple times.

The process would work as follows: (1) When a user seeks to use an IM service, the interoperability client would log on to multiple IMSPs at once, obtaining presence information from each IMSP and aggregating this information to present it to the user. (This aggregation occurs entirely on the local user's machine, inside the client, as can be seen in the picture. No information is collated or aggregated outside the user's computer.) (2) The IMSP client then would present to the user all the presence information, enabling the user to see all of his or her buddies on all IM services in a single IMSP client.

### Privacy and Security

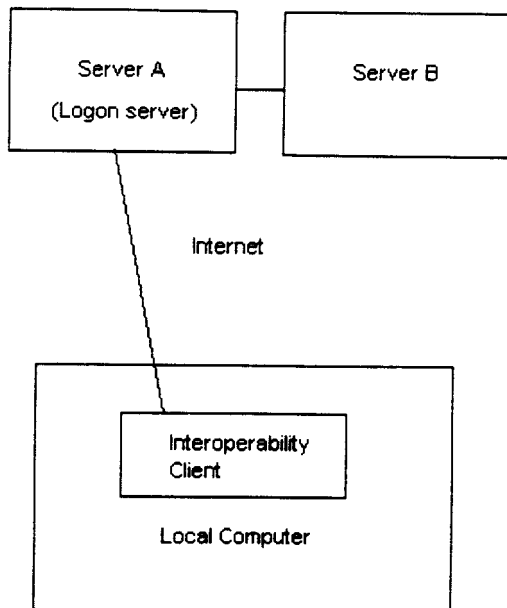
In both these interoperability examples, *no more information is shared between servers than in the non-interoperability case*. The user enters his or her username and password for each service into client-software, but the information is transmitted in exactly the same way as in the non-interoperability case. With client-side interoperability, each IMSP controls presence management and privacy individually. Since client-side interoperability only affects the user's local machine, the security and privacy of the respective IMSPs are not weakened. Each server enforces privacy and security just as it would in the non-interoperability case. *By definition, client-side interoperability is no less secure or private than no interoperability.*

Moreover, these basic types of interoperability do not require the sharing of intellectual property – only an understanding of what protocol is being used by each IMSP.

## *Server-Server Interoperability*

Server-server interoperability also avoids the need to run multiple clients for multiple services, but it is a more complex undertaking. The solution is different from client-side interoperability in that it only requires a single account, and a single logon sequence. The user connects to one arbitrary server (the “logon server”) with a single account’s username and password.

### Server-Server Interoperability Case



The logon server is responsible for contacting other services and retrieving presence information for a user’s buddies on all other services. This information is aggregated by the logon server and sent down to the client on a single connection. Likewise, an instant message to a buddy on another service must first be sent through the logon server, then to the service that the other buddy is logged on to.

To make server-server interoperability work, the servers of different IMSPs must “trust” one another. Trust among servers is important because presence information and instant messages go through multiple IMSPs (unlike in client-side interoperability). Also, there are many links in the chain for exchanging this information, and each link is a different IMSP. Developing mechanisms for establishing trust among different IMSPs’ servers and schema for exchanging information are complex tasks. The IETF’s working group on Instant Messaging and Presence Protocols is focusing intently on these issues. However, IM interoperability does not need to be delayed to address this most complex form of interoperability because, as demonstrated above, because client-to-client interoperability can be designed and implemented to protect consumer privacy and security.