

Network Working Group  
Internet-Draft  
Expires: February 12, 2003

D. Crocker  
Brandenburg  
A. Diacakis  
F. Mazzoldi  
Net Proj  
C. Huitema  
Microsoft  
G. Klyne  
Baltimore  
J. Rosenberg  
R. Sparks  
dynamicsoft  
H. Sugano  
Fujitsu  
J. Peterson  
NeuStar  
August 14, 2002

**Common Presence and Instant Messaging (CPIM)  
draft-ietf-imp-cpim-03**

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on February 12, 2003.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.



## Abstract

Semantics and data formats for common services of Instant Messaging and online Presence, independent of underlying transfer infrastructure, are described. The CPIM profile meets the requirements specified in [RFC 2779](#) using a minimalist approach allowing interoperation of a wide range of IM and Presence systems.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">4</a>
<a href="#">1.1</a>	Terminology . . . . .	<a href="#">4</a>
<a href="#">1.2</a>	Note on the Examples . . . . .	<a href="#">4</a>
<a href="#">2.</a>	Abstract Instant Messaging Service . . . . .	<a href="#">6</a>
<a href="#">2.1</a>	Overview . . . . .	<a href="#">6</a>
<a href="#">2.2</a>	Identification of INSTANT INBOXes . . . . .	<a href="#">6</a>
<a href="#">2.2.1</a>	Address Resolution . . . . .	<a href="#">6</a>
<a href="#">2.2.2</a>	Domain Name Lookup . . . . .	<a href="#">7</a>
<a href="#">2.2.3</a>	Processing SRV RRs . . . . .	<a href="#">7</a>
<a href="#">2.2.4</a>	Processing Multiple Addresses . . . . .	<a href="#">7</a>
<a href="#">2.3</a>	Format of Instant Messages . . . . .	<a href="#">8</a>
<a href="#">2.4</a>	The Messaging Service . . . . .	<a href="#">8</a>
<a href="#">2.4.1</a>	The Message Operation . . . . .	<a href="#">8</a>
<a href="#">2.4.2</a>	Looping . . . . .	<a href="#">9</a>
<a href="#">3.</a>	Abstract Presence Service . . . . .	<a href="#">10</a>
<a href="#">3.1</a>	Overview of the Presence Service . . . . .	<a href="#">10</a>
<a href="#">3.2</a>	Identification of PRESENTITIES . . . . .	<a href="#">12</a>
<a href="#">3.3</a>	Format of Presence Information . . . . .	<a href="#">12</a>
<a href="#">3.4</a>	The Presence Service . . . . .	<a href="#">12</a>
<a href="#">3.4.1</a>	The Subscribe Operation . . . . .	<a href="#">12</a>
<a href="#">3.4.2</a>	The Notify Operation . . . . .	<a href="#">14</a>
<a href="#">3.4.3</a>	The Unsubscribe Operation . . . . .	<a href="#">14</a>
<a href="#">3.4.4</a>	The Fetch Operation . . . . .	<a href="#">14</a>
<a href="#">4.</a>	Security Considerations . . . . .	<a href="#">15</a>
<a href="#">4.1</a>	Threats . . . . .	<a href="#">15</a>
<a href="#">4.2</a>	Hop-by-hop security . . . . .	<a href="#">16</a>
<a href="#">4.3</a>	End-to-end security . . . . .	<a href="#">16</a>
<a href="#">4.3.1</a>	Instant messages . . . . .	<a href="#">17</a>
<a href="#">4.3.2</a>	Presence service . . . . .	<a href="#">17</a>
<a href="#">5.</a>	IANA Considerations . . . . .	<a href="#">18</a>
<a href="#">5.1</a>	The IM URI Scheme . . . . .	<a href="#">18</a>
<a href="#">5.2</a>	The PRES URI Scheme . . . . .	<a href="#">18</a>
<a href="#">6.</a>	Common Service DTD . . . . .	<a href="#">19</a>
<a href="#">7.</a>	Message Service DTD . . . . .	<a href="#">20</a>
<a href="#">8.</a>	Presence Service DTD . . . . .	<a href="#">21</a>
<a href="#">9.</a>	Presence DTD . . . . .	<a href="#">22</a>
	Authors' Addresses . . . . .	<a href="#">24</a>
<a href="#">A.</a>	Message/CPIM Profile for Instant Messaging . . . . .	<a href="#">27</a>



<a href="#">B.</a>	Message/CPIM Profile for Presence . . . . .	<a href="#">28</a>
<a href="#">C.</a>	IM URL IANA Registration Template . . . . .	<a href="#">29</a>
<a href="#">C.1</a>	URL scheme name . . . . .	<a href="#">29</a>
<a href="#">C.2</a>	URL scheme syntax . . . . .	<a href="#">29</a>
<a href="#">C.3</a>	Character encoding considerations . . . . .	<a href="#">29</a>
<a href="#">C.4</a>	Intended usage . . . . .	<a href="#">29</a>
C.5	Applications and/or protocols which use this URL scheme name	29
<a href="#">C.6</a>	Interoperability considerations . . . . .	<a href="#">29</a>
<a href="#">C.7</a>	Security considerations . . . . .	<a href="#">30</a>
<a href="#">C.8</a>	Relevant publications . . . . .	<a href="#">30</a>
C.9	Person & email address to contact for further information	30
<a href="#">C.10</a>	Author/Change controller . . . . .	<a href="#">30</a>
C.11	Applications and/or protocols which use this URL scheme name	30
<a href="#">D.</a>	PRES URL IANA Registration Template . . . . .	<a href="#">31</a>
<a href="#">D.1</a>	URL scheme name . . . . .	<a href="#">31</a>
<a href="#">D.2</a>	URL scheme syntax . . . . .	<a href="#">31</a>
<a href="#">D.3</a>	Character encoding considerations . . . . .	<a href="#">31</a>
<a href="#">D.4</a>	Intended usage . . . . .	<a href="#">31</a>
D.5	Applications and/or protocols which use this URL scheme name	31
<a href="#">D.6</a>	Interoperability considerations . . . . .	<a href="#">31</a>
<a href="#">D.7</a>	Security considerations . . . . .	<a href="#">32</a>
<a href="#">D.8</a>	Relevant publications . . . . .	<a href="#">32</a>
D.9	Person & email address to contact for further information	32
<a href="#">D.10</a>	Author/Change controller . . . . .	<a href="#">32</a>
D.11	Applications and/or protocols which use this URL scheme name	32
<a href="#">E.</a>	Issues of Interest . . . . .	<a href="#">33</a>
<a href="#">E.1</a>	Address Mapping . . . . .	<a href="#">33</a>
<a href="#">E.2</a>	Source-Route Mapping . . . . .	<a href="#">33</a>
	References . . . . .	<a href="#">23</a>
<a href="#">F.</a>	Acknowledgements . . . . .	<a href="#">34</a>
	Full Copyright Statement . . . . .	<a href="#">35</a>



## **1. Introduction**

To achieve interoperation of IM and Presence systems that are compliant with [RFC 2779](#)[10], there must be a common agreement on both Instant Messaging and Presence services. This memo defines such an agreement according to the philosophy that there must be no loss of information between IM systems that are minimally conformant to [RFC2779](#).

This memo focuses on interoperation. Accordingly only those aspects of Presence and IM that require interoperation are discussed. For example, the "open instant inbox" operation is not applicable as this operation occurs within a single IM system and not across systems.

Service behavior is described abstractly in terms of operations invoked between the consumer and provider of a service. Accordingly, each IM service must specify how this behavior is mapped onto its own protocol interactions. The choice of strategy is a local matter, providing that there is a clear relation between the abstract behaviors of the service (as specified in this memo) and how it is faithfully realized by a particular IM service.

The parameters for each operation are defined using an abstract syntax. Although the syntax specifies the range of possible data values, each Presence and IM service must specify how well-formed instances of the abstract representation are encoded as a concrete series of bits.

For example, one strategy might transmit presence information as key/value pairs, another might use a compact binary representation, and a third might use nested containers. The choice of strategy is a local matter, providing that there is a clear relation between the abstract syntax (as specified in this memo) and how it is faithfully encoded by an particular Presence or IM service.

### **1.1 Terminology**

This memos makes use of the vocabulary defined in [RFC 2778](#)[9]. Terms such as CLOSED, INSTANT INBOX, INSTANT MESSAGE, OPEN, PRESENCE SERVICE, PRESENTITY, SUBSCRIPTION, and WATCHER are used in the same meaning as defined therein

### **1.2 Note on the Examples**

In the examples that follow, this memo uses time- sequence diagrams annotated with XML fragments to illustrate operations and their parameters. The use of XML is an artifact of this memo's presentation style and does not imply any requirement for the use of



XML in an IM system.

## 2. Abstract Instant Messaging Service

### 2.1 Overview

When an application wants to send a message to an INSTANT INBOX, it invokes the message operation, e.g.,

```

+-----+               +-----+
|       |               |       |
| appl. | -- message ----> | IM   |
|       |               | svc.  |
+-----+               +-----+

```

```

<message source='im:fred@example.com'
  destination='im:barney@example.com'
  transID='1' />
...

```

Content-Type: text/plain; charset="us-ascii"

Yabba, dabba, doo!

The service immediately responds by invoking the response operation containing the same transaction- identifier, e.g.,

```

+-----+               +-----+
|       |               |       |
| appl. | <----- response -- | IM   |
|       |               | svc.  |
+-----+               +-----+

```

```

<response status='success' transID='1' />

```

### 2.2 Identification of INSTANT INBOXes

An INSTANT INBOX is specified using an instant messaging URI with the 'im:' URI scheme. The full syntax of the IM URI scheme is given in [Appendix C](#).

#### 2.2.1 Address Resolution

A client determines the address of an appropriate system running a server by resolving the destination domain name that is part of the identifier to either an intermediate relay system or a final target system.

Only resolvable, fully-qualified, domain names (FQDNs) are permitted



when domain names are used in an IM URI (i.e., domain names that can be resolved to SRV[11] or A RRs).

### **2.2.2 Domain Name Lookup**

A client lexically identifies a domain to which instant messages will be delivered for processing, a DNS lookup **MUST** be performed to resolve the DOMAIN[3]. The names **MUST** be fully-qualified domain names (FQDNs) -- mechanisms for inferring FQDNs from partial names or local aliases are a local matter.

The lookup first attempts to locate SRV RRs associated with the domain. If a CNAME RR is found instead, the resulting domain is processed as if it were the initial domain.

If one or more SRV RRs are found for a given domain, a sender **MUST NOT** utilize any A RRs associated with that domain unless they are located using the SRV RRs. If no SRV RRs are found, but an A RR is found, then the A RR is treated as if it was associated with an implicit SRV RR, with a preference of 0, pointing to that domain.

### **2.2.3 Processing SRV RRs**

To process an IM URI, a lookup is performed for SRVs for the target domain and a desired IM transfer protocol.

For example, if the destination INSTANT INBOX is "im:fred@example.com", and the sender wishes to use an IM transfer protocol called "SIP", then a SRV lookup is performed for:

`_im._sip.example.com.`

The returned RRs, if any, specify the next-hop server.

The choice of IM transfer protocol is a local configuration option for each system.

Using this mechanism, seamless routing of IM traffic is possible, regardless of whether a gateway is necessary for interoperation. To achieve this transparency, a separate RR for a gateway must be present for each transfer protocol and domain pair that it serves.

### **2.2.4 Processing Multiple Addresses**

When the lookup succeeds, the mapping can result in a list of alternative delivery addresses rather than a single address, because of multiple SRV records, multihoming, or both. For reliable operations, the client **MUST** be able to try each of the relevant



addresses in this list in order, until a delivery attempt succeeds. However, there MAY also be a configurable limit on the number of alternate addresses that can be tried. In any case, the client SHOULD try at least two addresses. Two types of information are used to rank the domain addresses: multiple SRV records, and multihomed domains.

Multiple SRV records contain a preference indication that MUST be used in sorting. Lower numbers are preferable to higher ones. If there are multiple destinations with the same preference, and there is no clear reason to favor one (e.g., by recognition of an easily-reached address), then the sender MUST randomize them to spread the load across multiple servers for a specific destination.

The destination domain (perhaps taken from the preferred SRV record) may be multihomed, in which case the resolver will return a list of alternative IP addresses. It is the responsibility of the resolver to have ordered this list by decreasing preference if necessary, and the sender MUST try them in the order presented.

### **2.3 Format of Instant Messages**

An INSTANT MESSAGE comprises a "message/cpim" MIME object, as defined in CPIM MSGFMT and MESSAGE/CPIM PROFILE FOR INSTANT MESSAGING.

### **2.4 The Messaging Service**

THE COMMON SERVICE DTD and THE MESSAGING SERVICE DTD define the abstract syntax of the operations invoked with the service.

Note that the transaction-identifier parameters used with the service are potentially long-lived. Accordingly, the values of transaction-identifiers should appear to be unpredictable.

#### **2.4.1 The Message Operation**

When an application wants to send an INSTANT MESSAGE, it invokes the message operation.

The message operation has these parameters:

- o The source parameter specifies the INSTANT INBOX on whose behalf this message is sent (using an IM URI);
- o The destination parameter specifies the INSTANT INBOX that the message should be delivered to (using an IM URI);
- o The transID parameter specifies the transaction-identifier



associated with this operation; and,

- o The message to be sent.

When the service is informed of the message operation, it performs these steps:

1. If the source or destination does not refer to a valid INSTANT INBOX, a response operation having status "failure" is invoked.
2. If access control does not permit the application to request this operation, a response operation having status "failure" is invoked.
3. Otherwise:

If the service is able to successfully deliver the message, a response operation having status "success" is invoked.

If the service is unable to successfully deliver the message, a response operation having status "failure" is invoked.

If the service must delegate responsibility for delivery, and if the delegation will not result in a future authoritative indication to the service, a response operation having status "indeterminant" is invoked.

If the service must delegate responsibility for delivery, and if the delegation will result in a future authoritative indication to the service, then a response operation is invoked immediately after the indication is received.

When the service invokes the response operation, the transID parameter is identical to the value found in the message operation invoked by the application.

#### **2.4.2 Looping**

The dynamic routing of instant messages can result in looping of a message through a relay. Detection of loops is not always obvious, since aliasing and group list expansions can legitimately cause a message to pass through a relay more than one time.

Instant messaging uses a hop count mechanism, for detecting looping.



### 3. Abstract Presence Service

#### 3.1 Overview of the Presence Service

When an application wants to (periodically) receive the presence information associated with a PRESENTITY, it invokes the subscribe operation, e.g.,

```

+-----+               +-----+
|       |               |       |
| appl. | -- subscribe ----> | pres. |
|       |               | svc.  |
+-----+               +-----+

<subscribe watcher='pres:wilma@example.com'
target='pres:fred@example.com'
duration='86400' transID='2' />

```

The service immediately responds by invoking the response operation containing the same transaction- identifier, e.g.,

```

+-----+               +-----+
|       |               |       |
| appl. | <----- response -- | pres. |
|       |               | svc.  |
+-----+               +-----+

<response status='success' transID='2' duration='3600' />

```

A WATCHER may have at most one subscription for a PRESENTITY.

If the response operation indicates success, the service immediate invokes the notify operation to communicate the presence information to the WATCHER, e.g.,



```

+-----+               +-----+
|       |               |       |
| appl. | <----- notify -- | pres. |
|       |               | svc.  |
+-----+               +-----+

<notify watcher='pres:wilma@example.com'
target='pres:fred@example.com'
transID='1234'>
<presence entityInfo='http://www.example.com/fred/'>
<tuple destination='im:fred@example.com' status='open'
/>
</presence>
</notify>

```

If the duration parameter is non-zero, then for up to the specified duration, the service invokes the notify operation whenever there are any changes to the PRESENTITY's presence information. Otherwise, exactly one notify operation is invoked, achieving a one-time poll of the presence information. Regardless, there is no application response to the notify operation (i.e., the application does not invoke a response operation when a notify operation occurs).

The application may prematurely cancel a subscription by invoking the unsubscribe operation, e.g.,

```

+-----+               +-----+
|       |               |       |
| appl. | -- unsubscribe --> | pres. |
|       |               | svc.  |
+-----+               +-----+

<unsubscribe watcher='pres:wilma@example.com'
target='pres:fred@example.com'
transID='3' />

```

The service immediately responds by invoking the response operation containing the same transaction- identifier, e.g.,

```

+-----+               +-----+
|       |               |       |
| appl. | <----- response -- | pres. |
|       |               | svc.  |
+-----+               +-----+

<response status='success' transID='3' />

```



### **3.2 Identification of PRESENTITIES**

A PRESENTITY is specified using the PRES URI scheme, which is further described in [Appendix D](#).

To resolve identifiers associated with the Presence service, the mechanism defined in [Section 2.2.1](#) is used, except that the processing of a PRES URI is performed by looking up SRV RRs for a desired presence transfer protocol.

For example, if the destination PRESENTITY is "pres:fred@example.com", and the sender wishes to use a presence transfer protocol called "PEPP", then a SRV lookup is performed for:

\_pres.\_pepp.example.com.

### **3.3 Format of Presence Information**

The format of a Presence message is a MIME "Message/cpim" object, as defined in MESSAGE/CPIM PROFILE FOR PRESENCE and XML/MIME[6].

### **3.4 The Presence Service**

THE COMMON SERVICE DTD and THE PRESENCE SERVICE DTD define the abstract syntax of the operations invoked with the service.

An implementation of the service must maintain information about both presence information and in- progress operations in persistent storage.

Note that the transaction-identifier parameter used with the service is potentially long-lived. Accordingly, the values generated for this parameter should appear to be unpredictable.

#### **3.4.1 The Subscribe Operation**

When an application wants to (periodically) receive the presence information associated with an PRESENTITY, it invokes the subscribe operation.

The subscribe operation has these parameters:

- o The watcher parameter specifies the WATCHER associated with the subscription;
- o The target parameter specifies the PRESENTITY associated with the presence information;



- o The duration parameter specifies the maximum number of seconds that the SUBSCRIPTION should be active; and,
- o The transID parameter specifies the transaction-identifier associated with this operation.

When the service is informed of the subscribe operation, it performs these steps:

1. If the watcher or target parameter does not refer to a valid PRESENTITY, a response operation having status "failure" is invoked.
2. If access control does not permit the application to request this operation, a response operation having status "failure" is invoked.
3. If the duration parameter is non-zero, and if the watcher and target parameters refer to an in-progress subscribe operation for the application, a response operation having status "failure" is invoked.
4. Otherwise:

If the service is able to successfully deliver the message, a response operation having status "success" is invoked.

A response operation having status "success" is immediately invoked. (If the service chooses a different duration for the subscription then it conveys this information in the response operation.)

A notify operation, corresponding to the target's presence information, is immediately invoked for the watcher.

For up to the amount of time indicated by the duration parameter, if the target's presence information changes, and if access control allows, a notify operation is invoked for the watcher.

Note that if the duration parameter is zero-valued, then the subscribe operation is making a one-time poll of the presence information. Accordingly, Step 4.3 above does not occur.

When the service invokes a response operation as a result of this processing, the transID parameter is identical to the value found in the subscribe operation invoked by the application.



### **3.4.2 The Notify Operation**

The service invokes the notify operation whenever the presence information associated with a PRESENTITY changes and there are subscribers to that information.

The notify operation has these parameters:

- o The watcher parameter specifies the WATCHER associated with the subscription;
- o The target parameter specifies the PRESENTITY associated with the presence information;
- o The transID parameter specifies the transaction-identifier associated with this operation; and,
- o The presence information for the PRESENTITY.

There is no application response to the notify operation.

### **3.4.3 The Unsubscribe Operation**

When an application wants to terminate a subscription, it issues a SUBSCRIBE 0 with the ID of an existing subscription.

There is no explicit UNSUBSCRIBE command.

### **3.4.4 The Fetch Operation**

When an application wants to directly request presence information to be supplied immediately, it issues a SUBSCRIBE 0 with a new subscription ID.

There is no explicit FETCH command.



## **4. Security Considerations**

This memo makes no specific requirements on security procedures for interoperation between IM systems. Accordingly, trust between interconnected IM systems is determined in a bilateral matter.

However this memo does require that each IM system control access to its Instant Messaging and Presence services. Consult both [RFC 2778](#) and [RFC2779](#) for a discussion of security considerations for IM systems.

### **4.1 Threats**

Attacks, of concern for instant messaging, include access, deletion, insertion, reordering and modification of messages by unauthorized principals. Replay is a combination of a subset of these attacks.

These attacks can take place in the communication links between sending client and its server, between two servers, between the receiving client and its server, or by attacking any of the hosts involved. This document, not being concerned with client-server interchanges, only addresses threats aimed at server-server communication.

Countermeasures against unauthorized access are encrypted communication and encrypted messages.

Countermeasures against insertion of false messages are authentication and authorization of sending servers and strongly signed messages.

Countermeasures against reordered messages are date- stamped or serial-numbered messages, coupled with digital signatures that include the date or serial number, if modification is not otherwise guarded against.

Countermeasures against replayed messages are date stamps and unique message IDs, coupled with digital signatures that include the date or serial number, if modification is not otherwise guarded against.

Countermeasures against deletion of messages are integrity-protected connections between servers where the server's identity is verified. Serial-numbered messages can also be useful in detecting deleted messages.

Attacks that target the server hosts rather than the communication channels can successfully defeat all countermeasures that depend on host security. Digital signatures and encrypted messages do not



depend on host security, for intermediate systems, but cannot by themselves guard against deletion or reordering of messages.

For presence, the attacks include giving presence information to unauthorized watchers, not reporting watcher information back to a presentity, and insertion, modification, deletion and replay of presence update messages. The same set of countermeasures is relevant.

Instant messaging and presence systems can provide security at two levels: hop-by-hop and/or end-to-end.

#### **4.2 Hop-by-hop security**

A useful but imperfect level of security can be provided on a hop-by-hop basis, with all aspects of the communication including message content and originator verification, using transfer level security between servers. The main drawback of this approach is that it requires that each server that handles message or presence information must be trusted. But it is relatively easy to deploy, because it depends only on bilateral arrangements between directly communicating servers.

The underlying principles for using hop-by-hop security are:

Each server and/or domain must keep their own house in order, ensuring that operations and information accesses are allowed only to appropriately authorized parties, and

Each server and/or domain must make its own choices about the levels of trust to be established to any other server and/or domain with which they directly communicate.

When passing IM and presence information between services using different protocols, a gateway system **MUST** be capable of using security mechanisms appropriate to each of the protocols concerned, and must have access to keys needed to authenticate any other system with which it needs to directly communicate in a secure fashion.

#### **4.3 End-to-end security**

End-to-end security is widely regarded as being more satisfactory than hop-by-hop security, as the need to trust intermediate parties is reduced. However, some aspects of end-to-end security are difficult to achieve because they need bilateral arrangement between any pair of communicating parties about acceptable security standards to use, and key exchange. Reliance on bilateral agreements does not scale well. A moderating alternative is a third-party certification



service and this approach, so far, has not found large-scale use.

The two IETF standards for end-to-end MIME object security are OpenPGP[7] and S/MIME[8]. They require a public key operation for each message. For repeated, short transactions, this overhead can be onerous. A version of these specifications, which permitted re-use of the public key across multiple messages, would greatly reduce instant messaging overhead.

#### **[4.3.1](#) Instant messages**

End to end security for instant messages can be provided using any of the MIME-based security mechanisms (S/MIME [\[8\]](#), OpenPGP [\[7\]](#)), as instant message payload content is not interpreted or reformatted in transit.

This specification allows any pair of communicating parties to use any MIME-based security framework for instant messages (c.f. [section 2.3](#)), but mechanisms for establishing the required bilateral arrangements and key exchange are not specified here.

#### **[4.3.2](#) Presence service**

End-to-end security for presence notifications and subscriptions could be provided by any MIME-based security mechanism.



## **5. IANA Considerations**

The IANA assigns the "im" and "pres" URL schemes.

### **5.1 The IM URI Scheme**

The Instant Messaging (IM) URI scheme designates an Internet resource, namely an INSTANT INBOX.

The syntax of an IM URL is given in [Appendix C](#).

### **5.2 The PRES URI Scheme**

The Presence (PRES) URI scheme designates an Internet resource, namely a PRESENTITY or WATCHER.

The syntax of a PRES URL is given in [Appendix D](#).

## 6. Common Service DTD

Note that the DTDs given in the following sections are used to describe abstract information services, and do not alone provide a complete description of an instant messaging and presence system.

```
<!-- DTD for the IM common profile, as of 2000-08-16
Refer to this DTD as:
```

```
<!ENTITY % IMCOMMON PUBLIC "-//Blocks//DTD IM
COMMON//EN"
"http://xml.resource.org/syntaxes/IM/im-common.dtd">
%IMCOMMON; -->
```

```
<!-- DTD data types: entity syntax/reference example
=====
```

```
a language tag LANG c.f., [RFC-1766] "en", "en-US",
etc. seconds SECONDS 0..2147483647 600
```

```
unique-identifier UNIQID 1..2147483647 42 authoritative
identity URI c.f., [RFC-2396] http://invisible.net/ -->
```

```
<!ENTITY % LANG "NMTOKEN">
<!ENTITY % SECONDS "CDATA">
<!ENTITY % UNIQID "CDATA"> <!ENTITY % URI "CDATA">
<!-- Abstract syntax for the response operation -->
<!ELEMENT response (#PCDATA)>
<!ATTLIST response status (success | failure |
indeterminant)
#REQUIRED transID %UNIQID;
#REQUIRED duration %SECONDS;
#IMPLIED xml:lang %LANG;
#IMPLIED >
```



## 7. Message Service DTD

```
<!-- DTD for the abstract IM messaging service, as of
2000-08-16 Refer to this DTD as:
!ENTITY % IMMESSAGING PUBLIC "-//Blocks//DTD IM
MESSAGING//EN"
"http://xml.resource.org/syntaxes/IM/im-messaging.dtd">
%IMMESSAGING; -->
```

```
<!ENTITY % IMCOMMON PUBLIC "-//Blocks//DTD IM
COMMON//EN"
"http://xml.resource.org/syntaxes/IM/im-common.dtd">
%IMCOMMON; <!-- DTD data types: entity syntax/reference
example
```

```
=====
INBOX c.f., Section 5.1 im:fred@example.com -->
<!ENTITY % INBOX "CDATA">
<!-- Abstract syntax for the message operation -->
<!ELEMENT message (#PCDATA)>
```

```
<!ATTLIST message source %INBOX;
#REQUIRED destination %INBOX;
#REQUIRED transID %UNIQID;
#REQUIRED >
```



## 8. Presence Service DTD

```
<!-- DTD for the abstract IM presence service, as
of 2000-08-16 Refer to this DTD as:
<!ENTITY % IMPRESENCE PUBLIC "-//Blocks//DTD IM
PRESENCE//EN"
"http://xml.resource.org/syntaxes/IM/im-presence.dtd">
%IMPRESENCE; -->
<!ENTITY % IMCOMMON PUBLIC "-//Blocks//DTD IM
COMMON//EN"
"http://xml.resource.org/syntaxes/IM/im-common.dtd">
%IMCOMMON;
<!-- DTD data types: entity syntax/reference example
=====
PRESENTITY c.f., Section 5.2 pres:fred@example.com -->
<!ENTITY % PRESENTITY "CDATA">
<!-- Abstract syntax for presence information -->
<!ELEMENT presence (tuple+)>
<!ATTLIST presence entityInfo %URI; "" >
<!ELEMENT tuple (#PCDATA)>
<!ATTLIST tuple destination %URI;
#REQUIRED status (open | closed)
#REQUIRED >
<!-- Abstract syntax for the subscribe operation -->
<!ELEMENT subscribe EMPTY>
<!ATTLIST subscribe watcher %PRESENTITY; #REQUIRED
target %PRESENTITY;
#REQUIRED duration %SECONDS;
#REQUIRED transID %UNIQID;
#REQUIRED >
<!-- Abstract syntax for the notify operation -->
<!ELEMENT notify (presence)>
<!ATTLIST notify watcher %PRESENTITY;
#REQUIRED target %PRESENTITY;
#REQUIRED transID %UNIQID;
#REQUIRED>
```



## 9. Presence DTD

```
<!-- DTD the IM presence information of 2000-11-6 Refer
to this DTD as:
<!ENTITY % IMPRESENCEINFO PUBLIC "-//Blocks//DTD IM
PRESENCE//EN"
"http://xml.resource.org/syntaxes/IM/im-presence-
info.dtd"> %IMPRESENCEINFO; -->

<!ENTITY % IMCOMMON PUBLIC "-//Blocks//DTD IM
COMMON//EN"
"http://xml.resource.org/syntaxes/IM/im-common.dtd">
%IMCOMMON;
<!-- DTD data types: entity syntax/reference example
=====
PRESENTITY c.f., Section 5.2 pres:Fred@example.com -->
<!ENTITY % PRESENTITY "CDATA">
<!-- Abstract syntax for presence information -->
<!ELEMENT presence (tuple+)>
<!ATTLIST presence entityInfo %URI; "" >
<!ELEMENT tuple (#PCDATA)>
<!ATTLIST tuple destination %URI;
#REQUIRED status (open | closed)
#REQUIRED >
```



## References

- [1] Crocker, D., "Standard for the format of ARPA Internet text Messages", [RFC 822](#), STD 11, August 1982.
- [2] Resnick, P., "Internet Message Format", [RFC 2822](#), STD 11, April 2001.
- [3] Mockapetris, P., "Domain Names - Concepts and Facilities", [RFC 1034](#), STD 13, November 1987.
- [4] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", [RFC 2045](#), November 1996.
- [5] Callas, J., Donnerhake, L., Finney, H. and R. Thayer, "OpenPGP Message Format", [RFC 2440](#), November 1998.
- [6] Klyne, G., "XML Coding of [RFC822](#) Messages", [draft-klyne-message-rfc822-xml-00](#) (work in progress), November 2001.
- [7] Atkins, D. and G. Klyne, "Common Presence and Instant Messaging: Message Format", [draft-ietf-imp-cpim-msgfmt-05](#) (work in progress), December 2001.
- [8] Sugano, H., "CPIM Presence Information Data Format", [draft-ietf-imp-cpim-pidf-00](#) (work in progress), August 2001.
- [9] Ramsdell, B., "S/MIME Version 3 Certificate Handling", [RFC 2632](#), June 1999.
- [10] Day, M., Rosenberg, J. and H. Sugano, "A Model for Presence and Instant Messaging", [RFC 2778](#), February 2000.
- [11] Day, M., Aggarwal, S. and J. Vincent, "Instant Messaging / Presence Protocol Requirements", [RFC 2779](#), February 2000.
- [12] Gulbrandsen, A., Vixie, P. and L. Esibov, "A DNS RR for Specifying the Location of Services (SRV)", [RFC 2782](#), February 2000.
- [13] Allocchio, C., "GSTN Address Element Extensions in Email Services", [RFC 2846](#), June 2000.



Authors' Addresses

Dave Crocker  
Brandenburg InternetWorking  
675 Spruce Drive  
Sunnyvale, CA 94086  
US

Phone: +1 408/246-8253  
EMail: dcrocker@brandenburg.com

Athanassios Diacakis  
Network Projects Inc.  
4516 Henry Street  
Suite 113  
Pittsburgh, PA 15213  
US

Phone: +1 412/681-6950 x202  
EMail: thanos@networkprojects.com

Florencio Mazzoldi  
Network Projects Inc.  
4516 Henry Street  
Suite 113  
Pittsburgh, PA 15213  
US

Phone: +1 412/681-6950  
EMail: flo@networkprojects.com

Christian Huitema  
Microsoft Corporation  
One Microsoft Way  
Redmund, WA 98052-6399  
US

EMail: huitema@microsoft.com



Graham Klyne  
Baltimore Technologies  
1310 Waterside  
Arlington Business Park  
Theale, Reading RG7 4SA  
UK

Phone: +44 118 903 8000  
EMail: gk@acm.org

Jonathan Rosenberg  
dynamicsoft  
200 Executive Drive  
Suite 120  
West Orange, NJ 07052  
US

EMail: jdrosen@dynamicsoft.com

Robert Sparks  
dynamicsoft  
200 Executive Drive  
Suite 120  
West Orange, NJ 07052  
US

EMail: rsparks@dynamicsoft.com

Hiroyasu Sugano  
Fujitsu Laboratories Ltd.  
200 Executive Drive  
64 Nishiwaki, Ohkubo-cho  
Akashi 674-8555  
JP

EMail: suga@flab.fujitsu.co.jp



Jon Peterson  
NeuStar, Inc.  
1800 Sutter St  
Suite 570  
Concord, CA 94520  
US

Phone: +1 925/363-8720  
EMail: jon.peterson@neustar.biz

## [Appendix A](#). Message/CPIM Profile for Instant Messaging

Implicit default namespace URI:  
urn:ietf:params:cpim-headers:

Message/CPIM headers that MUST be recognized and understood by an instant messaging client:

From

To

cc

DateTime

Subject

Require

(Other headers, if present, may be ignored unless they are named in a "Require" header.)

Message/CPIM headers that MUST be present in an instant message:

From

To

DateTime   [[[?]]]

Subject    [[[?]]]



**[Appendix B](#). Message/CPIM Profile for Presence**

[Ed. - This section contains detail that creates a profile of Content-Type=Message/CPIM, to cover use for Presence transactions. Text to be partly extracted from draft-ietf-impp-cpim-pidf-00.txt.]

## **[Appendix C](#). IM URL IANA Registration Template**

This section provides the information to register the im: instant messaging URL.

### **[C.1](#) URL scheme name**

im

### **[C.2](#) URL scheme syntax**

The syntax follows the existing mailto: URL syntax specified in [RFC2368](#). The ABNF is:

```
IM-URL      = "im:" [ to ] [ headers ]
to           = #mailbox
headers      = "?" header *( "&" header )
header       = hname "=" hvalue
hname        = *urlc
hvalue       = *urlc
```

### **[C.3](#) Character encoding considerations**

Representation of non-ASCII character sets in local-part strings is limited to the standard methods provided as extensions to [RFC 2822](#)<sup>[1]</sup>

### **[C.4](#) Intended usage**

Use of the im: URL follows closely usage of the mailto: URL. That is, invocation of an IM URL will cause the user's instant messaging application to start, with destination address and message headers fill-in according to the information supplied in the URL.

### **[C.5](#) Applications and/or protocols which use this URL scheme name**

It is anticipated that protocols compliant with [RFC2779](#), and meeting the interoperability requirements specified here, will make use of this URL scheme name.

### **[C.6](#) Interoperability considerations**

The underlying exchange protocol used to send an instant message may vary from service to service. Therefore complete, Internet-scale interoperability cannot be guaranteed. However, a service conforming to this specification permits gateways to achieve interoperability sufficient to the requirements of [RFC2779](#).



### **C.7 Security considerations**

When IM URLs are placed in instant messaging protocols, they convey the identity of the sender and/or the recipient. In some cases, anonymous messaging may be desired. Such a capability is beyond the scope of this specification.

### **C.8 Relevant publications**

[RFC2779](#), [RFC2778](#)

### **C.9 Person & email address to contact for further information**

Jon Peterson [mailto:jon.peterson@neustar.biz]

### **C.10 Author/Change controller**

This scheme is registered under the IETF tree. As such, IETF maintains change control.

### **C.11 Applications and/or protocols which use this URL scheme name**

Instant messaging service; presence service



## **[Appendix D](#). PRES URL IANA Registration Template**

This section provides the information to register the pres: presence URL .

### **[D.1](#) URL scheme name**

pres

### **[D.2](#) URL scheme syntax**

The syntax follows the existing mailto: URL syntax specified in [RFC2368](#). The ABNF is:

```
PRES-URL      = "pres:" [ to ] [ headers ]
to             = #mailbox
headers       = "?" header *( "&" header )
header        = hname "=" hvalue
hname         = *urlc
hvalue        = *urlc
```

### **[D.3](#) Character encoding considerations**

Representation of non-ASCII character sets in local-part strings is limited to the standard methods provided as extensions to [RFC 2822](#)[1]

### **[D.4](#) Intended usage**

Use of the pres: URL follows closely usage of the mailto: URL. That is, invocation of an PRES URL will cause the user's instant messaging application to start, with destination address and message headers fill-in according to the information supplied in the URL.

### **[D.5](#) Applications and/or protocols which use this URL scheme name**

It is anticipated that protocols compliant with [RFC2779](#), and meeting the interoperability requirements specified here, will make use of this URL scheme name.

### **[D.6](#) Interoperability considerations**

The underlying exchange protocol used to send an instant message may vary from service to service. Therefore complete, Internet-scale interoperability cannot be guaranteed. However, a service conforming to this specification permits gateways to achieve interoperability sufficient to the requirements of [RFC2779](#).



### **D.7 Security considerations**

When PRES URLs are placed in presence protocols, they convey the identity of the sender and/or the recipient. In some cases, anonymous messaging may be desired. Such a capability is beyond the scope of this specification.

### **D.8 Relevant publications**

[RFC2779](#), [RFC2778](#)

### **D.9 Person & email address to contact for further information**

Jon Peterson [mailto:jon.peterson@neustar.biz]

### **D.10 Author/Change controller**

This scheme is registered under the IETF tree. As such, IETF maintains change control.

### **D.11 Applications and/or protocols which use this URL scheme name**

Instant messaging service; presence service



## **Appendix E. Issues of Interest**

This appendix briefly discusses issues that may be of interest when designing an interoperation gateway.

### **E.1 Address Mapping**

When mapping the service described in this memo, mappings that place special information into the im: address local-part MUST use the meta-syntax defined in [RFC 2486](#)[12].

### **E.2 Source-Route Mapping**

The easiest mapping technique is a form of source- routing and usually is the least friendly to humans having to type the string. Source-routing also has a history of operational problems.

Use of source-routing for exchanges between different services is by a transformation that places the entire, original address string into the im: address local part and names the gateway in the domain part.

For example, if the destination INSTANT INBOX is "pepp://example.com/fred", then, after performing the necessary character conversions, the resulting mapping is:

```
im:pepp=example.com/fred@relay-domain
```

where "relay-domain" is derived from local configuration information.

Experience shows that it is vastly preferable to hide this mapping from end-users - if possible, the underlying software should perform the mapping automatically.



[Appendix F](#). Acknowledgments

## Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

