A Model for Presence and Instant Messaging

Status of this Memo

Copyright Notice

Abstract

This document defines an abstract model for a presence and instant
messaging system. It defines the various entities involved, defines
terminology, and outlines the services provided by the system. The
goal is to provide a common vocabulary for further work on
requirements for protocols and markup for presence and instant
messaging.

1. Introduction

A presence and instant messaging system allows users to subscribe to
each other and be notified of changes in state, and for users to send
each other short instant messages. To facilitate development of a
suite of protocols to provide this service, we believe that it is
valuable to first develop a model for the system. The model consists
of the various entities involved, descriptions of the basic functions
they provide, and most importantly, definition of a vocabulary which
can be used to facilitate discussion.

We note that the purpose of this model is to be descriptive and
universal: we want the model to map reasonably onto all of the
systems that are informally described as presence or instant
messaging systems. The model is not intended to be prescriptive or
achieve interoperability: an element that appears in the model will
not necessarily be an element of an interoperable protocol, and may
not even be a good idea.

In this document, each element of the model appears in upper case
(e.g., PRESENCE SERVICE). No term in lower case or mixed case is
intended to be a term of the model.

The first part of this document is intended as an overview of the
model. The overview includes diagrams, and terms are presented in an
order that is intended to help the reader understand the relationship
between elements. The second part of the document is the actual
definition of the model, with terms presented in alphabetical order
for ease of reference.

The overview is intended to be helpful but is not definitive; it may
contain inadvertent differences from the definitions in the model.
For any such difference, the definition(s) in the model are taken to
be correct, rather than the explanation(s) in the overview.

2. Overview

The model is intended to provide a means for understanding,
comparing, and describing systems that support the services typically
referred to as presence and instant messaging. It consists of a
number of named entities that appear, in some form, in existing
systems. No actual implementation is likely to have every entity of
the model as a distinct part. Instead, there will almost always be
parts of the implementation that embody two or more entities of the
model. However, different implementations may combine entities in
different ways.

The model defines two services: a PRESENCE SERVICE and an INSTANT
MESSAGE SERVICE. The PRESENCE SERVICE serves to accept information,
store it, and distribute it. The information stored is
(unsurprisingly) PRESENCE INFORMATION. The INSTANT MESSAGE SERVICE
serves to accept and deliver INSTANT MESSAGES to INSTANT INBOXES.

2.1 PRESENCE SERVICE

The PRESENCE SERVICE has two distinct sets of "clients" (remember,
these may be combined in an implementation, but treated separately in
the model). One set of clients, called PRESENTITIES, provides
PRESENCE INFORMATION to be stored and distributed. The other set of
clients, called WATCHERS, receives PRESENCE INFORMATION from the
service.

```
                    +---------------------------+
                    |      PRESENCE SERVICE     |
                    |                           |
                    +---------------------------+
                          ^                 |
                          |                 |
                          |                 v
                    +------------+     +------------+
                    | PRESENTITY |     |  WATCHER   |
                    +------------+     +------------+
```
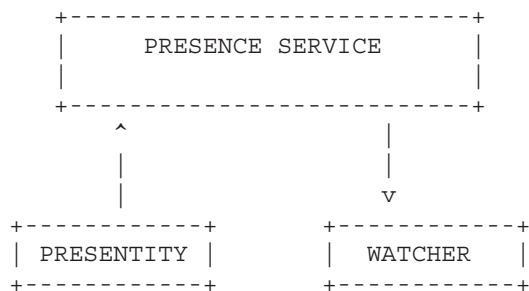
                   Fig. 1: Overview of Presence Service

   There are two kinds of WATCHERS, called FETCHERS and SUBSCRIBERS. A
   FETCHER simply requests the current value of some PRESENTITY's
   PRESENCE INFORMATION from the PRESENCE SERVICE. In contrast, a
   SUBSCRIBER requests notification from the PRESENCE SERVICE of
   (future) changes in some PRESENTITY's PRESENCE INFORMATION. A
   special kind of FETCHER is one that fetches information on a regular
   basis. This is called a POLLER.

```
         +----------------WATCHER--------------+
         |                                     |
         |   +----FETCHER---+   +--SUBSCRIBER--+   |
         |   |              |   |              |   |
         |   | +--POLLER--+ |   |              |   |
         |   | |          | |   |              |   |
         |   | +----------+ |   |              |   |
         |   +--------------+   +--------------+   |
         +-------------------------------------+
```
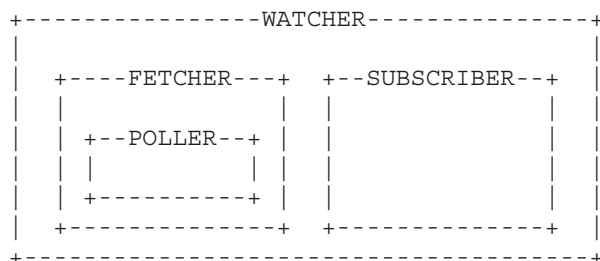
                     Fig. 2: Varieties of WATCHER

   The PRESENCE SERVICE also has WATCHER INFORMATION about WATCHERS and
   their activities in terms of fetching or subscribing to PRESENCE
   INFORMATION. The PRESENCE SERVICE may also distribute WATCHER
   INFORMATION to some WATCHERS, using the same mechanisms that are
   available for distributing PRESENCE INFORMATION.

   Changes to PRESENCE INFORMATION are distributed to SUBSCRIBERS via
   NOTIFICATIONS. Figures 3a through 3c show the flow of information as
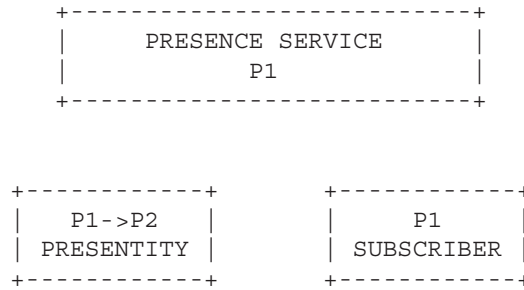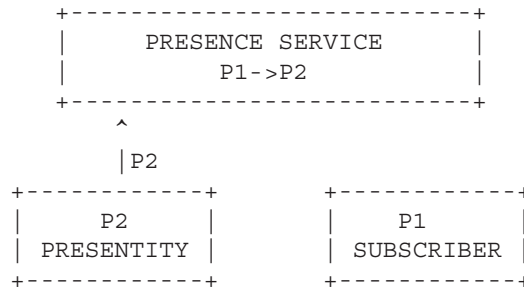   a piece of PRESENCE INFORMATION is changed from P1 to P2.

```
                     +---------------------------+
                     |      PRESENCE SERVICE     |
                     |            P1             |
                     +---------------------------+


            +------------+          +------------+
            |   P1->P2   |          |     P1     |
            | PRESENTITY |          | SUBSCRIBER |
            +------------+          +------------+

                 Fig. 3a: NOTIFICATION (Step 1)



                     +---------------------------+
                     |      PRESENCE SERVICE     |
                     |          P1->P2           |
                     +---------------------------+
                        ^
                        |P2
            +------------+          +------------+
            |     P2     |          |     P1     |
            | PRESENTITY |          | SUBSCRIBER |
            +------------+          +------------+

                 Fig. 3b: NOTIFICATION (Step 2)



                     +---------------------------+
                     |      PRESENCE SERVICE     |
                     |            P2             |
                     +---------------------------+
                                       |P2
                                       v
            +------------+          +------------+
            |     P2     |          |   P1->P2   |
            | PRESENTITY |          | SUBSCRIBER |
            +------------+          +------------+

                 Fig. 3c: NOTIFICATION (Step 3)
```
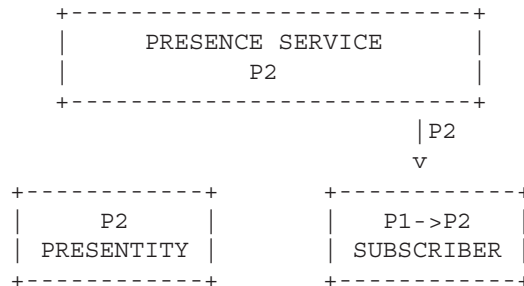
2.2 INSTANT MESSAGE SERVICE

   The INSTANT MESSAGE SERVICE also has two distinct sets of "clients":
   SENDERS and INSTANT INBOXES. A SENDER provides INSTANT MESSAGES to
   the INSTANT MESSAGE SERVICE for delivery. Each INSTANT MESSAGE is

addressed to a particular INSTANT INBOX ADDRESS, and the INSTANT
MESSAGE SERVICE attempts to deliver the message to a corresponding
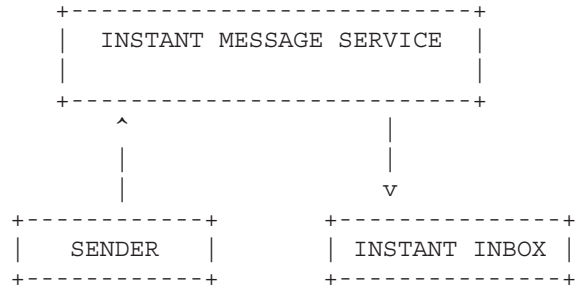INSTANT INBOX.

```
          +----------------------------+
          |   INSTANT MESSAGE SERVICE  |
          |                            |
          +----------------------------+
               ^                |
               |                |
               |                v
       +------------+      +---------------+
       |   SENDER   |      | INSTANT INBOX |
       +------------+      +---------------+
```

              Fig. 4: Overview of Instant Message Service

2.3 Protocols

   A PRESENCE PROTOCOL defines the interaction between PRESENCE SERVICE,
   PRESENTITIES, and WATCHERS. PRESENCE INFORMATION is carried by the
   PRESENCE PROTOCOL.

   An INSTANT MESSAGE PROTOCOL defines the interaction between INSTANT
   MESSAGE SERVICE, SENDERS, and INSTANT INBOXES. INSTANT MESSAGES are
   carried by the INSTANT MESSAGE PROTOCOL.

   In terms of this model, we believe that the IMPP working group is
   planning to develop detailed requirements and specifications for the
   structure and formats of the PRESENCE PROTOCOL, PRESENCE INFORMATION,
   INSTANT MESSAGE PROTOCOL, and INSTANT MESSAGES.

2.4 Formats

   The model defines the PRESENCE INFORMATION to consist of an arbitrary
   number of elements, called PRESENCE TUPLES. Each such element
   consists of a STATUS marker (which might convey information such as
   online/offline/busy/away/do not disturb), an optional COMMUNICATION
   ADDRESS, and optional OTHER PRESENCE MARKUP. A COMMUNICATION ADDRESS
   includes a COMMUNICATION MEANS and a CONTACT ADDRESS. One type of
   COMMUNICATION MEANS, and the only one defined by this model, is
   INSTANT MESSAGE SERVICE. One type of CONTACT ADDRESS, and the only
   one defined by this model, is INSTANT INBOX ADDRESS. However, other
   possibilities exist: a COMMUNICATION MEANS might indicate some form
   of telephony, for example, with the corresponding CONTACT ADDRESS
   containing a telephone number.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.