

Network Working Group  
Request for Comments: 4480  
Category: Standards Track

H. Schulzrinne  
Columbia U.  
V. Gurbani  
Lucent  
P. Kyzivat  
J. Rosenberg  
Cisco  
July 2006

RPID: Rich Presence Extensions to the  
Presence Information Data Format (PIDF)

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

The Presence Information Data Format (PIDF) defines a basic format for representing presence information for a presentity. This format defines a textual note, an indication of availability (open or closed) and a Uniform Resource Identifier (URI) for communication. The Rich Presence Information Data format (RPID) described here is an extension that adds optional elements to the Presence Information Data Format (PIDF). These extensions provide additional information about the presentity and its contacts. The information is designed so that much of it can be derived automatically, e.g., from calendar files or user activity.

This extension includes information about what the person is doing, a grouping identifier for a tuple, when a service or device was last used, the type of place a person is in, what media communications might remain private, the relationship of a service tuple to another presentity, the person's mood, the time zone it is located in, the type of service it offers, an icon reflecting the presentity's status, and the overall role of the presentity.

These extensions include presence information for persons, services (tuples), and devices.

## Table of Contents

1. Introduction .....	2
2. Terminology and Conventions .....	4
3. RIPD Elements .....	4
3.1. Overview .....	4
3.2. Activities Element .....	7
3.3. Class Element .....	10
3.4. Device Identifier .....	10
3.5. Mood Element .....	10
3.6. Place-is Element .....	12
3.7. Place-type Element .....	13
3.8. Privacy Element .....	14
3.9. Relationship Element .....	15
3.10. Service Class .....	15
3.11. Sphere Element .....	16
3.12. Status-Icon Element .....	16
3.13. Time Offset .....	17
3.14. User-Input Element .....	17
4. Example .....	18
5. XML Schema Definitions .....	20
5.1. urn:ietf:params:xml:ns:pidf:rpid .....	20
6. Extending RIPD .....	30
7. IANA Considerations .....	31
7.1. URN Sub-Namespace Registration for .....	31
'urn:ietf:params:xml:ns:pidf:rpid'	
7.2. Schema Registration for Schema .....	32
'urn:ietf:params:xml:ns:pidf:status:rpid'	
8. Internationalization Considerations .....	32
9. Security Considerations .....	32
10. References .....	33
10.1. Normative References .....	33
10.2. Informative References .....	34
Appendix A. Acknowledgements .....	35

## 1. Introduction

The Presence Information Data Format (PIDF) definition [8] describes a basic presence information data format, encoded as an Extensible Markup Language (XML) [9] (SCHEMA-1 [10]) (SCHEMA-2 [11]), for exchanging presence information in systems compliant with the common model for presence and instant messaging [5]. It consists of a <presence> root element, zero or more <tuple> elements carrying presence information including a Uniform Resource Identifier (URI) for communication, zero or more <note> elements, and zero or more extension elements from other name spaces. Each tuple defines a basic status of either "open" or "closed".

However, it is frequently useful to convey additional information about a user that needs to be interpreted by an automata, and is therefore not appropriate to be placed in the <note> element of the PIDF document, which is typically intended for the human observer. Therefore, this specification defines extensions to the PIDF document format for conveying richer presence information. Generally, the extensions have been chosen to provide features common in existing presence systems at the time of writing, in addition to elements that could readily be derived automatically from existing sources of presence, such as calendaring systems or communication devices, or sources describing the user's current physical environment.

The presence data model [16] defines the concepts of service, device, and person as the data elements that are used to model the state of a presentity. (The term "presentity" is defined in RFC 2778 [5] and abbreviates presence entity. A presentity provides presence information to a presence service.) Services are encoded using the <tuple> element, defined in PIDF; devices and persons are represented by the <device> and <person> XML elements, respectively, defined in the data model [16]. However, neither PIDF nor the data model defines presence attributes beyond the <basic> status element.

This specification defines additional presence attributes to describe person, service, and device data elements, summarized as "Rich Presence Information Data format for presence" (RPID). These attributes are specified by XML elements that extend the PIDF <tuple> element and the <device> and <person> elements defined in the data model.

This extension has two main goals:

1. Provide rich presence information that is at least as powerful as common commercial presence systems. Such feature-parity simplifies transition to systems complying with the Common Profile for Instant Messaging (CPIM) [14], both in terms of user acceptance and protocol conversion.
2. Maintain backward-compatibility with PIDF, so that PIDF-only watchers and gateways can continue to function properly, naturally without access to the functionality described here.

We make no assumptions as to how the information in the RPID elements is generated. Experience has shown that users are not always diligent about updating their presence status. Thus, we want to make it as easy as possible to derive RPID information from other information sources, such as personal calendars, the status of communication devices such as telephones, typing activity, and

physical presence detectors as commonly found in energy-management systems.

Many of the elements correspond to data commonly found in personal calendars. Thus, we attempted to align some of the extensions with the usage found in calendar formats such as iCal [13].

The information in a presence document can be generated by a single entity or can be composed from information published by multiple entities.

Note that PIDF documents and this extension can be used in two different contexts, namely, by the presentity to publish its presence status and by the presence server to notify some set of watchers. The presence server MAY compose, translate, or filter the published presence state before delivering customized presence information to the watcher. For example, it may merge presence information from multiple presence user agents, remove whole elements, translate values in elements, or remove information from elements. Mechanisms that filter calls and other communications to the presentity can subscribe to this presence information just like a regular watcher and in turn generate automated rules, such as scripts [15], that govern the actual communications behavior of the presentity. Details are described in the data model document.

Since RPID is a PIDF XML document, it also uses the content type `application/pidf+xml`.

## 2. Terminology and Conventions

This memo makes use of the vocabulary defined in the IMPP model document [5]. Terms such as CLOSED, INSTANT MESSAGE, OPEN, PRESENCE SERVICE, PRESENTITY, WATCHER, and WATCHER USER AGENT in the memo are used in the same meaning as defined therein.

The key words MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL in this document are to be interpreted as described in BCP 14, RFC 2119 [1].

## 3. RPID Elements

### 3.1. Overview

Some of the RPID elements describe services, some devices, and some the person. As such, they either extend `<tuple>`, `<device>`, or `<person>`, respectively. Below, we summarize the RPID elements. The next sections will then provide more detailed descriptions.

activities: The <activities> status element enumerates what the person is doing.

class: An identifier that groups similar person elements, devices, or services.

deviceID: A device identifier in a tuple references a <device> element, indicating that this device contributes to the service described by the tuple.

mood: The <mood> status element indicates the mood of the person.

place-is: The <place-is> status element reports on the properties of the place the presentity is currently at, such as the levels of light and noise.

place-type: The <place-type> status elements reports the type of place the person is located in, such as 'classroom' or 'home'.

privacy: The <privacy> element distinguishes whether the communication service is likely to be observable by other parties.

relationship: When a service is likely to reach a user besides the person associated with the presentity, the relationship indicates how that user relates to the person.

service-class: The <service-class> element describes whether the service is delivered electronically, is a postal or delivery service, or describes in-person communications.

sphere: The <sphere> element characterizes the overall current role of the presentity.

status-icon: The <status-icon> element depicts the current status of the person or service.

time-offset: The <time-offset> status element quantifies the time zone the person is in, expressed as the number of minutes away from UTC.

user-input: The <user-input> element records the user-input or usage state of the service or device, based on human user input.

The 'From/until?' column in Table 1 indicates by an 'x' that the element can take 'from' and 'until' attributes. An 'x' in the 'Note?' column marks elements that can include a <note> element. The usage of these elements within the <person>, <tuple>, and <device> elements is shown in columns 4 through 6. An 'x' in the respective

column indicates that the RPID element MAY appear as a child of that element.

Element	From/until ?	Note ?	<person>	<tuple>	<device>
<activities>	x	x	x		
<class>			x	x	x
<deviceID>				x	
<mood>	x	x	x		
<place-is>	x	x	x		
<place-type>	x	x	x		
<privacy>	x	x	x	x	
<relationship>		x		x	
<service-class>		x		x	
<sphere>	x		x		
<status-icon>	x		x	x	
<time-offset>	x		x		
<user-input>			x	x	x

Table 1

In general, it is unlikely that a presentity will publish or announce all of these elements at the same time. Rather, these elements were chosen to give the presentity maximum flexibility in deriving this information from existing sources, such as calendaring tools, device activity sensors, or location trackers, as well as to manually configure this information. In either case, there is no guarantee that the information is accurate, as users forget to update calendars or may not always adjust the presence information manually.

The namespace URIs for these elements defined by this specification are URNs [2], using the namespace identifier 'ietf' defined by [4] and extended by [6]:

urn:ietf:params:xml:ns:pidf:rpidd

The elements marked with the value 'x' in column 2 of Table 1 MAY be qualified with the 'from' and 'until' attributes to describe the absolute time when the element assumed this value and the absolute time until which this element is expected to be valid. Note that there can be multiple elements of the same type, whose time ranges SHOULD NOT overlap.

Elements MAY contain an 'id' attribute that allows to uniquely reference the element.

Enumerations can be extended by elements from other namespaces, as described in Section 6. The <activities>, <mood>, and <place-type> elements can also take <other> elements containing text, for custom free-text values specific to an application.

All elements described in this document are optional within PIDF documents.

### 3.2. Activities Element

The <activities> element describes what the person is currently doing, expressed as an enumeration of activity-describing elements. A person can be engaged in multiple activities at the same time, e.g., traveling and having a meal. The <activities> element can be quite helpful to the watcher in judging how appropriate a communication attempt is and which means of communications is most likely to succeed and not annoy the person. The activity indications correspond roughly to the category field in calendar entries, such as Section 4.8.1.2 of RFC 2445 [13].

An activities enumeration consists of one or more elements using elements drawn from the list below, a string enclosed in the <other> element, or IANA-registered values from other namespaces (Section 7).

If a person publishes an activity of "permanent-absence", it is likely that all services will report a status of CLOSED. In general, services MAY advertise either service status for any activity value.

Activities such as <appointment>, <breakfast>, <dinner>, <holiday>, <lunch>, <meal>, <meeting>, <performance>, <travel>, or <vacation> can often be derived from calendar information.

**appointment:** The person has a calendar appointment, without specifying exactly of what type. This activity is indicated if more detailed information is not available or the person chooses not to reveal more information.

**away:** The person is physically away from all interactive communication devices. This activity element was included since it can often be derived automatically from security systems, energy management systems, or entry badge systems. Although this activity would typically be associated with a status of CLOSED across all services, a person may declare himself or herself away

to discourage communication, but indicate that he or she still can be reached if needed. However, communication attempts might reach an answering service, for example.

**breakfast:** The person is eating the first meal of the day, usually eaten in the morning.

**busy:** The person is busy, without further details. Although this activity would typically be associated with a status of CLOSED across all services, a person may declare himself or herself busy to discourage communication, but indicate that he or she still can be reached if needed.

**dinner:** The person is having his or her main meal of the day, eaten in the evening or at midday.

**holiday:** This is a scheduled national or local holiday.

**in-transit:** The person is riding in a vehicle, such as a car, but not steering. The <place-type> element provides more specific information about the type of conveyance the person is using.

**looking-for-work:** The presentity is looking for (paid) work.

**lunch:** The person is eating his or her midday meal.

**meal:** The person is scheduled for a meal, without specifying whether it is breakfast, lunch, or dinner, or some other meal.

**meeting:** The person is in an assembly or gathering of people, as for a business, social, or religious purpose. A meeting is a sub-class of an appointment.

**on-the-phone:** The person is talking on the telephone. This activity is included since it can often be derived automatically.

**other:** The person is engaged in an activity with no defined representation as an <activities> element. The enclosed string describes the activity in plain text.

**performance:** A performance is a sub-class of an appointment and includes musical, theatrical, and cinematic performances as well as lectures. It is distinguished from a meeting by the fact that the person may either be lecturing or be in the audience, with a potentially large number of other people, making interruptions particularly noticeable.

permanent-absence: The person will not return for the foreseeable future, e.g., because it is no longer working for the company. This activity is associated with a status of CLOSED across all services.

playing: The person is occupying himself or herself in amusement, sport, or other recreation.

presentation: The person is giving a presentation, lecture, or participating in a formal round-table discussion.

shopping: The person is visiting stores in search of goods or services.

sleeping: This activity category can often be generated automatically from a calendar, local time information, or biometric data.

spectator: The person is observing an event, such as a sports event.

steering: The person is controlling a vehicle, watercraft, or plane.

travel: The person is on a business or personal trip, but not necessarily in-transit.

tv: The person is watching television.

unknown: The activity of the person is unknown. This element is generally not used together with other activities.

vacation: A period of time devoted to pleasure, rest, or relaxation.

working: The presentity is engaged in, typically paid, labor, as part of a profession or job.

worship: The presentity is participating in religious rites.

The <activities> element MAY be qualified with the 'from' and 'until' attributes as described in Section 3.1.

Example:

```
<activities>
  <note>Enjoying the morning paper</note>
  <vacation/>
  <breakfast/>
  <other>reading</other>
</activities>
```

### 3.3. Class Element

The <class> element describes the class of the service, device, or person. Multiple elements can have the same class name within a presence document, but each person, service, or device can only have one class label. The naming of classes is left to the presentity. The presentity can use this information to group similar services, devices, or person elements or to convey information that the presence agent can use for filtering or authorization. This information is not generally presented to the watcher user interface.

The <class> element MUST NOT be qualified with the 'from' and 'until' attributes as described in Section 3.1.

### 3.4. Device Identifier

The <deviceID> element in the <tuple> element references the device that provides a particular service. The element is defined syntactically in the data model [16] schema. One service can be provided by multiple devices, so that each service tuple may contain zero or more <deviceID> elements. There is no significance in the order of these elements.

The <deviceID> element MUST NOT be qualified with the 'from' and 'until' attributes as described in Section 3.1.

### 3.5. Mood Element

The <mood> element describes the mood of the presentity. The mood values are enumerated chosen by the presentity. The mood itself is provided as the element name of a defined child element of the <mood> element (e.g., <happy/>); one such child element is REQUIRED. The user MAY also specify a natural-language description of, or reason for, the mood in the <note> child of the <mood> element, which is OPTIONAL. (This definition follows the Jabber Extension JEP-107.) It is RECOMMENDED that an implementation support the mood values proposed in Jabber Extension JEP-0107, which in turn are a superset of the Wireless Village [18] mood values and the values enumerated in the Affective Knowledge Representation that has been defined by Lisetti [17]:

A mood enumeration consists of one or more elements using elements drawn from the list below, a string enclosed in the <other> element, or IANA-registered values from other namespaces (Section 7).

The <mood> element MAY be qualified with the 'from' and 'until' attributes as described in Section 3.1.

- o afraid
- o amazed
- o angry
- o annoyed
- o anxious
- o ashamed
- o bored
- o brave
- o calm
- o cold
- o confused
- o contented
- o cranky
- o curious
- o depressed
- o disappointed
- o disgusted
- o distracted
- o embarrassed
- o excited
- o flirtatious
- o frustrated
- o grumpy
- o guilty
- o happy
- o hot
- o humbled
- o humiliated
- o hungry
- o hurt
- o impressed
- o in\_awe
- o in\_love
- o indignant
- o interested
- o invincible
- o jealous
- o lonely
- o mean
- o moody
- o nervous
- o neutral
- o offended
- o other
- o playful
- o proud
- o relieved
- o remorseful

- o restless
- o sad
- o sarcastic
- o serious
- o shocked
- o shy
- o sick
- o sleepy
- o stressed
- o surprised
- o thirsty
- o unknown
- o worried

Example:

```
<mood>
  <note>I'm ready for the bar BOF!</note>
  <sleepy/>
  <thirsty/>
</mood>
```

### 3.6. Place-is Element

The <place-is> element describes properties of the place the person is currently at. This offers the watcher an indication of what kind of communication is likely to be successful. Each major media type has its own set of attributes. Omitting the element indicates that the property is unknown.

For audio, we define the following attributes:

noisy: The person is in a place with a level of background noise that makes audio communications difficult.

ok: The environmental conditions are suitable for audio communications.

quiet: The person is in a place such as a library, restaurant, place of worship, or theater that discourages noise, conversation, and other distractions.

unknown: The place attributes for audio are unknown.

For video, we define the following attributes:

toobright: The person is in a bright place, sufficient for good rendering on video.

ok: The environmental conditions are suitable for video.

dark: The person is in a dark place, and thus the camera may not be able to capture a good image.

unknown: The place attributes for video are unknown.

For text (real-time text and instant messaging), we define

uncomfortable: Typing or other text entry is uncomfortable.

inappropriate: Typing or other text entry is inappropriate, e.g., since the user is in a vehicle or house of worship.

ok: The environmental conditions are suitable for text-based communications.

unknown: The place attributes for text are unknown.

This list can be augmented by free-text values in a note or additional IANA-registered values (Section 7).

The <place-is> element contains other elements, e.g.,

```
<place-is>
  <audio>
    <noisy />
  </audio>
  <video>
    <dark />
  </video>
</place-is>
```

The <place-is> element MAY be qualified with the 'from' and 'until' attributes as described in Section 3.1.

### 3.7. Place-type Element

The <place-type> element describes the type of place the person is currently at. This offers the watcher an indication of what kind of communication is likely to be appropriate. The initial set of values is contained in RFC 4589 [12].

This list can be augmented by free-text values or additional IANA-registered values as described in RFC 4589.

The <place-type> element is a choice of elements, as in

```
<place-type>
  <pt:street/>
</place-type>
```

The <place-type> element MAY be qualified with the 'from' and 'until' attributes as described in Section 3.1.

### 3.8. Privacy Element

The <privacy> element indicates which types of communication third parties in the vicinity of the presentity are unlikely to be able to intercept accidentally or intentionally. This does not in any way describe the privacy properties of the electronic communication channel, e.g., properties of the encryption algorithm or the network protocol used.

audio: Inappropriate individuals are not likely to overhear audio communications.

text: Inappropriate individuals are not likely to see text communications.

unknown: This information is unknown.

video: Inappropriate individuals are not likely to see video communications.

The <privacy> element can be used by logic executing on the watcher or by a composer to filter, sort and label tuples. For example, a composer may have rules that limit the publication of tuples labeled "private" to a select subset of the watchers.

The <privacy> element MAY be qualified with the 'from' and 'until' attributes as described in Section 3.1.

Example:

```
<privacy>
  <text/>
  <audio/>
</privacy>
```

### 3.9. Relationship Element

The <relationship> element extends <tuple> and designates the type of relationship an alternate contact has with the presentity. This element is provided only if the tuple refers to somebody other than the presentity. Relationship values include "family", "friend", "associate" (e.g., for a colleague), "assistant", "supervisor", "self", and "unknown". The default is "self".

If a relationship is indicated, the URI in the <contact> element refers to the entity, such as the assistant, that has a relationship to the presentity, not the presentity itself.

Like tuples without a <relationship> qualifier, the <contact> element for tuples labeled with a relationship can contain either a communication URI such as "im", "sip", "sips", "h323", "tel", or "mailto", or a presence URI, such as "pres" or "sip".

Example:

```
<relationship>
  <friend/>
</relationship>
```

### 3.10. Service Class

The <service-class> element extends <tuple> and designates the type of service offered.

electronic: Delivery of information by electronic means, i.e., without delivering physical objects. Examples include telephone, fax, email, instant messaging, and SMS.

postal: Delivery by the postal service, e.g., as a letter, parcel, or postcard. Delivery could be to a post office box or central mailroom rather than the presentity's office location, for example.

courier: Delivery by messenger, overnight delivery, or courier. Courier-delivered messages are usually delivered to a receptionist rather than, say, a mailroom or receiving department.

freight: Delivery by freight carrier, typically of larger objects that are not sent by postal mail or courier. The recipient is often the shipping department or a loading dock.

in-person: Describes the coordinates for visits in person, as by a visitor, i.e., usually somebody's office or residence.

unknown: The type of service is unknown.

Electronic service is implied if omitted. The service types 'postal', 'courier', 'freight', and 'in-person' MUST NOT be used unless the contact URI is empty. Additional data elements defined elsewhere describe the physical service delivery address for the in-person, postal, or delivery services. Such addresses might be specified in geospatial coordinates, civic addresses, or some specialized address format, e.g., for interstellar addresses or a company-specific delivery system.

Example:

```
<service-class><postal/></service-class>
```

### 3.11. Sphere Element

The <sphere> element designates the current state and role that the person plays. For example, it might describe whether the person is in a work mode, at home, or participating in activities related to some other organization such as the IETF or a church. This document does not define names for these spheres except for two common ones, "work" and "home", as well as "unknown".

Spheres allow the person to easily turn on or off certain rules that depend on what groups of people should be made aware of the person's status. For example, if the person is a Boy Scout leader, he might set the sphere to "scouting" and then have a rule set that allows other scout masters in his troop to see his presence status. As soon as he switches his status to "work", "home", or some other sphere, the fellow scouts would lose access.

The <sphere> element MAY be qualified with the 'from' and 'until' attributes as described in Section 3.1.

Example:

```
<sphere>
  <home/>
</sphere>
```

### 3.12. Status-Icon Element

The <status-icon> element includes a URI pointing to an image (icon) representing the current status of the person or service. The watcher MAY use this information to represent the status in a graphical user interface. Presentities SHOULD provide images of

sizes and aspect ratios that are appropriate for rendering as an icon. Support for JPEG, PNG, and GIF formats is RECOMMENDED.

Watchers resolving the URI MUST validate whether the local copy of the icon is current when receiving a notification, using the standard cache control mechanism in the URI-identified retrieval protocol.

Example:

```
<status-icon>http://www.example.com/playing.gif</status-icon>
```

### 3.13. Time Offset

The `<time-offset>` element describes the number of minutes of offset from UTC at the person's current location. A positive number indicates that the local time-of-day is ahead (i.e., east of) Universal Time, while a negative number indicates that the local time-of-day is behind (i.e., west of) Universal Time. Transitions into and out of daylight savings time may temporarily cause a difference between the true offset from UTC and the time offset element.

An optional attribute, `description`, can be used to describe the offset, e.g., by labeling the time zone. This description is meant for human consumption.

Publishers on mobile devices SHOULD NOT publish this information unless they know the time offset information to reflect the current location. (For example, many laptop users do not update their time zone when traveling.) Publishers SHOULD update the information whenever they discover that their UTC offset has changed.

Example:

```
<time-offset description="America/New_York">-300
</time-offset>
```

### 3.14. User-Input Element

The `<user-input>` element records the user-input or usage state of the service or device, based on human user input, e.g., keyboard, pointing device, or voice. If contained in a `<person>` element, it summarizes any user input activity across all services and devices operated by the present entity. The mechanism for such aggregation is beyond the scope of this document, but generally reflects the most recent user input across all devices and services. The element can assume one of two values, namely, 'active' or 'idle', with an optional 'last-input' attribute that records when the last user input

was received. An optional 'idle-threshold' element records how long the presentity will wait before reporting the service or device to be idle, measured in seconds.

(A two-state model was chosen since it would otherwise be necessary to send repeated last-input updates during continuous activity.)

A service that wants to indicate user input activity sends a <user-input> 'active' indication when the user has provided user input within a configurable interval of time, the idle-threshold. If the user ceases to provide input and the idle-threshold has elapsed, the tuple is marked with a <user-input> 'idle' indication instead, optionally including the time of last activity in the 'last-input' attribute. An example is below:

```
<user-input idle-threshold="600"
  last-input="2004-10-21T13:20:00.000-05:00">idle</user-input>
```

Depending on device or service capabilities, user input may be detected only for a particular application, i.e., when the application has user focus or when a user has sent a message or placed a call, or can be based on user input across all applications running on one end system.

The <user-input> element may be used by a watcher, typically in combination with other data, to estimate how likely a user is to answer when contacting the service. A tuple that has not been used in a while may still be OPEN, but a watcher may choose to first contact a URI in a tuple that is both OPEN and has been used more recently.

The <user-input> attribute can be omitted if the presentity wants to indicate that the device has not been used for a while, but does not want to reveal the precise duration, as in the following:

```
<user-input>idle</user-input>
```

Configuration MUST include the option to omit the 'last-input' attribute.

#### 4. Example

The example below describes the presentity 'pres:someone@example.com', which has a SIP contact, 'sip:someone@example.com', representing a service. It also has a device contact, as an email box. The presentity is in a meeting, in a public office setting. The 'until' information indicates that he will be there until 5:30 pm local time. The presentity also has an

assistant, sip:secretary@example.com, who happens to be available for communications.

```
<?xml version="1.0" encoding="UTF-8"?>
  <presence xmlns="urn:ietf:params:xml:ns:pidf"
    xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
    xmlns:lt="urn:ietf:params:xml:ns:location-type"
    xmlns:rpid="urn:ietf:params:xml:ns:pidf:rpid"
    entity="pres:someone@example.com">

    <tuple id="bs35r9">
      <status>
        <basic>open</basic>
      </status>
      <dm:deviceID>urn:device:0003ba4811e3</dm:deviceID>
      <rpid:relationship><rpid:self/></rpid:relationship>
      <rpid:service-class><rpid:electronic/></rpid:service-class>
      <contact priority="0.8">im:someone@mobile.example.net</contact>
      <note xml:lang="en">Don't Disturb Please!</note>
      <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
      <timestamp>2005-10-27T16:49:29Z</timestamp>
    </tuple>

    <tuple id="ty4658">
      <status>
        <basic>open</basic>
      </status>
      <rpid:relationship><rpid:assistant/></rpid:relationship>
      <contact priority="1.0">mailto:secretary@example.com</contact>
    </tuple>

    <tuple id="eg92n8">
      <status>
        <basic>open</basic>
      </status>
      <dm:deviceID>urn:x-mac:0003ba4811e3</dm:deviceID>
      <rpid:class>email</rpid:class>
      <rpid:service-class><rpid:electronic/></rpid:service-class>
      <rpid:status-icon>http://example.com/mail.png</rpid:status-icon>
      <contact priority="1.0">mailto:someone@example.com</contact>
    </tuple>

    <note>I'll be in Tokyo next week</note>

    <dm:device id="pc147">
      <rpid:user-input idle-threshold="600"
        last-input="2004-10-21T13:20:00-05:00">idle</rpid:user-input>
      <dm:deviceID>urn:device:0003ba4811e3</dm:deviceID>
```

```

    <dm:note>PC</dm:note>
  </dm:device>

  <dm:person id="p1">
    <rpip:activities from="2005-05-30T12:00:00+05:00"
      until="2005-05-30T17:00:00+05:00">
      <rpip:note>Far away</rpip:note>
      <rpip:away/>
    </rpip:activities>
    <rpip:class>calendar</rpip:class>
    <rpip:mood>
      <rpip:angry/>
      <rpip:other>brooding</rpip:other>
    </rpip:mood>
    <rpip:place-is>
      <rpip:audio>
        <rpip:noisy/>
      </rpip:audio>
    </rpip:place-is>
    <rpip:place-type><lt:residence/></rpip:place-type>
    <rpip:privacy><rpip:unknown/></rpip:privacy>
    <rpip:sphere>bowling league</rpip:sphere>
    <rpip:status-icon>http://example.com/play.gif</rpip:status-icon>

    <rpip:time-offset>-240</rpip:time-offset>
    <dm:note>Scoring 120</dm:note>
    <dm:timestamp>2005-05-30T16:09:44+05:00</dm:timestamp>
  </dm:person>
</presence>

```

## 5. XML Schema Definitions

The RIPD schema is shown below. Due to limitations in composing schemas, not all XML documents that validate against the schema below are semantically valid RIPD documents. In particular, the schema allows each element to appear anywhere in PIDF or data-model elements; Table 1 restricts where these elements can appear for semantically valid RIPD documents. Elements that do not have from/until parameters MUST NOT appear more than once in each <person>, <tuple>, or <device>.

### 5.1. urn:ietf:params:xml:ns:pidf:ripd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:ietf:params:xml:ns:pidf:ripd"
  xmlns="urn:ietf:params:xml:ns:pidf:ripd"
  xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"

```

```
elementFormDefault="qualified"
attributeFormDefault="unqualified">

<xs:simpleType name="activeIdle">
  <xs:restriction base="xs:string">
    <xs:enumeration value="active"/>
    <xs:enumeration value="idle"/>
  </xs:restriction>
</xs:simpleType>

<xs:element name="activities">
  <xs:annotation>
    <xs:documentation>
      Describes what the person is currently doing, expressed as
      an enumeration of activity-describing elements. A person
      can be engaged in multiple activities at the same time,
      e.g., traveling and having a meal.
    </xs:documentation>
  </xs:annotation>

  <xs:complexType>
    <xs:sequence>
      <xs:element name="note" type="Note_t" minOccurs="0"
        maxOccurs="unbounded" />
      <xs:choice>
        <xs:element name="unknown" type="empty" minOccurs="0"/>
        <xs:sequence maxOccurs="unbounded">
          <xs:choice>
            <xs:element name="appointment"
              type="empty" />
            <xs:element name="away"
              type="empty" />
            <xs:element name="breakfast"
              type="empty" />
            <xs:element name="busy"
              type="empty" />
            <xs:element name="dinner"
              type="empty" />
            <xs:element name="holiday"
              type="empty" />
            <xs:element name="in-transit"
              type="empty" />
            <xs:element name="looking-for-work"
              type="empty" />
            <xs:element name="meal"
              type="empty" />
            <xs:element name="meeting"
              type="empty" />
          </xs:choice>
        </xs:sequence>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

    <xs:element name="on-the-phone"
      type="empty" />
    <xs:element name="performance"
      type="empty" />
    <xs:element name="permanent-absence"
      type="empty" />
    <xs:element name="playing"
      type="empty" />
    <xs:element name="presentation"
      type="empty" />
    <xs:element name="shopping"
      type="empty" />
    <xs:element name="sleeping"
      type="empty" />
    <xs:element name="spectator"
      type="empty" />
    <xs:element name="steering"
      type="empty" />
    <xs:element name="travel"
      type="empty" />
    <xs:element name="tv"
      type="empty" />
    <xs:element name="vacation"
      type="empty" />
    <xs:element name="working"
      type="empty" />
    <xs:element name="worship"
      type="empty" />
    <xs:element name="other"
      type="Note_t" />
    <xs:any namespace="##other"
      maxOccurs="unbounded" processContents="lax"/>
  </xs:choice>
</xs:sequence>
</xs:choice>
</xs:sequence>
<xs:attributeGroup ref="fromUntil"/>
<xs:attribute name="id" type="xs:ID"/>
<xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
</xs:element>

<xs:element name="class" type="xs:token">
  <xs:annotation>
    <xs:documentation>
      Describes the class of the service, device or person.
    </xs:documentation>
  </xs:annotation>

```

```
</xs:element>

<xs:element name="mood">
  <xs:annotation>
    <xs:documentation>
      Describes the mood of the presentity.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="note" type="Note_t" minOccurs="0"
        maxOccurs="unbounded" />
      <xs:choice>
        <xs:element name="unknown" type="empty"/>
        <xs:sequence maxOccurs="unbounded">
          <xs:choice>
            <xs:element name="afraid"
              type="empty"/>
            <xs:element name="amazed"
              type="empty"/>
            <xs:element name="angry"
              type="empty"/>
            <xs:element name="annoyed"
              type="empty"/>
            <xs:element name="anxious"
              type="empty" />
            <xs:element name="ashamed"
              type="empty" />
            <xs:element name="bored"
              type="empty" />
            <xs:element name="brave"
              type="empty" />
            <xs:element name="calm"
              type="empty" />
            <xs:element name="cold"
              type="empty" />
            <xs:element name="confused"
              type="empty" />
            <xs:element name="contented"
              type="empty" />
            <xs:element name="cranky"
              type="empty" />
            <xs:element name="curious"
              type="empty" />
            <xs:element name="depressed"
              type="empty" />
            <xs:element name="disappointed"
              type="empty" />
          </xs:choice>
        </xs:sequence>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="disgusted"
  type="empty" />
<xs:element name="distracted"
  type="empty" />
<xs:element name="embarrassed"
  type="empty" />
<xs:element name="excited"
  type="empty" />
<xs:element name="flirtatious"
  type="empty" />
<xs:element name="frustrated"
  type="empty" />
<xs:element name="grumpy"
  type="empty" />
<xs:element name="guilty"
  type="empty" />
<xs:element name="happy"
  type="empty" />
<xs:element name="hot"
  type="empty" />
<xs:element name="humbled"
  type="empty" />
<xs:element name="humiliated"
  type="empty" />
<xs:element name="hungry"
  type="empty" />
<xs:element name="hurt"
  type="empty" />
<xs:element name="impressed"
  type="empty" />
<xs:element name="in_awe"
  type="empty" />
<xs:element name="in_love"
  type="empty" />
<xs:element name="indignant"
  type="empty" />
<xs:element name="interested"
  type="empty" />
<xs:element name="invincible"
  type="empty" />
<xs:element name="jealous"
  type="empty" />
<xs:element name="lonely"
  type="empty" />
<xs:element name="mean"
  type="empty" />
<xs:element name="moody"
  type="empty" />
```

```
<xs:element name="nervous"
  type="empty" />
<xs:element name="neutral"
  type="empty" />
<xs:element name="offended"
  type="empty" />
<xs:element name="playful"
  type="empty" />
<xs:element name="proud"
  type="empty" />
<xs:element name="relieved"
  type="empty" />
<xs:element name="remorseful"
  type="empty" />
<xs:element name="restless"
  type="empty" />
<xs:element name="sad"
  type="empty" />
<xs:element name="sarcastic"
  type="empty" />
<xs:element name="serious"
  type="empty" />
<xs:element name="shocked"
  type="empty" />
<xs:element name="shy"
  type="empty" />
<xs:element name="sick"
  type="empty" />
<xs:element name="sleepy"
  type="empty" />
<xs:element name="stressed"
  type="empty" />
<xs:element name="surprised"
  type="empty" />
<xs:element name="thirsty"
  type="empty" />
<xs:element name="worried"
  type="empty" />
<xs:element name="other"
  type="Note_t" />
<xs:any namespace="##other"
  maxOccurs="unbounded" processContents="lax"/>
</xs:choice>
</xs:sequence>
</xs:choice>
</xs:sequence>
<xs:attributeGroup ref="fromUntil"/>
<xs:attribute name="id" type="xs:ID"/>
```

```

    <xs:anyAttribute namespace="##any" processContents="lax"/>
  </xs:complexType>
</xs:element>

<xs:element name="place-is">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="note" type="Note_t" minOccurs="0"
        maxOccurs="unbounded" />
      <xs:element name="audio" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="noisy" type="empty" />
            <xs:element name="ok" type="empty" />
            <xs:element name="quiet" type="empty" />
            <xs:element name="unknown" type="empty" />
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="video" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="toobright" type="empty" />
            <xs:element name="ok" type="empty" />
            <xs:element name="dark" type="empty" />
            <xs:element name="unknown" type="empty" />
          </xs:choice>
        </xs:complexType>
      </xs:element>
      <xs:element name="text" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="uncomfortable" type="empty" />
            <xs:element name="inappropriate" type="empty" />
            <xs:element name="ok" type="empty" />
            <xs:element name="unknown" type="empty" />
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attributeGroup ref="fromUntil"/>
    <xs:attribute name="id" type="xs:ID"/>
    <xs:anyAttribute namespace="##any" processContents="lax"/>
  </xs:complexType>
</xs:element>

<xs:element name="place-type">
  <xs:annotation>

```

```

    <xs:documentation>
      Describes the type of place the person is currently at.
    </xs:documentation>
  </xs:annotation>
</xs:complexType>
<xs:sequence>
  <xs:element name="note" type="Note_t" minOccurs="0"
    maxOccurs="unbounded" />
  <xs:choice>
    <xs:element name="other" type="Note_t"/>
    <xs:any namespace="##other" maxOccurs="unbounded"
      processContents="lax"/>
  </xs:choice>
</xs:sequence>
<xs:attributeGroup ref="fromUntil"/>
<xs:attribute name="id" type="xs:ID"/>
<xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
</xs:element>

<xs:element name="privacy">
  <xs:annotation>
    <xs:documentation>
      Indicates which type of communication third parties in the
      vicinity of the presentity are unlikely to be able to
      intercept accidentally or intentionally.
    </xs:documentation>
  </xs:annotation>
</xs:complexType>
<xs:sequence>
  <xs:element name="note" type="Note_t" minOccurs="0"
    maxOccurs="unbounded" />
  <xs:choice>
    <xs:element name="unknown" type="empty"/>
    <xs:sequence minOccurs="1">
      <xs:element name="audio" type="empty" minOccurs="0"/>
      <xs:element name="text" type="empty" minOccurs="0"/>
      <xs:element name="video" type="empty" minOccurs="0"/>
      <xs:any namespace="##other" minOccurs="0"
        maxOccurs="unbounded" processContents="lax"/>
    </xs:sequence>
  </xs:choice>
</xs:sequence>
<xs:attributeGroup ref="fromUntil"/>
<xs:attribute name="id" type="xs:ID"/>
<xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
</xs:element>

```

```
<xs:element name="relationship">
  <xs:annotation>
    <xs:documentation>
      Designates the type of relationship an alternate contact
      has with the presentity.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="note" type="Note_t" minOccurs="0"
        maxOccurs="unbounded" />
      <xs:choice>
        <xs:element name="assistant" type="empty" />
        <xs:element name="associate" type="empty" />
        <xs:element name="family" type="empty" />
        <xs:element name="friend" type="empty" />
        <xs:element name="other" type="Note_t" minOccurs="0" />
        <xs:element name="self" type="empty" />
        <xs:element name="supervisor" type="empty" />
        <xs:element name="unknown" type="empty" />
        <xs:any namespace="##other" maxOccurs="unbounded"
          processContents="lax"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="service-class">
  <xs:annotation>
    <xs:documentation>
      Designates the type of service offered.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="note" type="Note_t" minOccurs="0"
        maxOccurs="unbounded" />
      <xs:choice>
        <xs:element name="courier" type="empty" />
        <xs:element name="electronic" type="empty" />
        <xs:element name="freight" type="empty" />
        <xs:element name="in-person" type="empty" />
        <xs:element name="postal" type="empty" />
        <xs:element name="unknown" type="empty" />
        <xs:any namespace="##other" maxOccurs="unbounded"
          processContents="lax"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
</xs:complexType>
</xs:element>

<xs:element name="sphere">
  <xs:annotation>
    <xs:documentation>
      Designates the current state and role that the person plays.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:choice minOccurs="0">
      <xs:element name="home" type="empty" />
      <xs:element name="work" type="empty" />
      <xs:element name="unknown" type="empty" />
      <xs:any namespace="##other" maxOccurs="unbounded"
        processContents="lax"/>
    </xs:choice>
    <xs:attributeGroup ref="fromUntil"/>
    <xs:attribute name="id" type="xs:ID"/>
    <xs:anyAttribute namespace="##any" processContents="lax"/>
  </xs:complexType>
</xs:element>

<xs:element name="status-icon">
  <xs:annotation>
    <xs:documentation>
      A URI pointing to an image (icon) representing the current
      status of the person or service.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:anyURI">
        <xs:attributeGroup ref="fromUntil"/>
        <xs:attribute name="id" type="xs:ID"/>
        <xs:anyAttribute namespace="##any" processContents="lax"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

<xs:element name="time-offset">
  <xs:annotation>
    <xs:documentation>
      Describes the number of minutes of offset from UTC at the
      user's current location.
    </xs:documentation>
  </xs:annotation>
```

```

<xs:complexType>
  <xs:simpleContent>
    <xs:extension base="xs:integer">
      <xs:attributeGroup ref="fromUntil"/>
      <xs:attribute name="description"
        type="xs:string"/>
      <xs:attribute name="id" type="xs:ID"/>
      <xs:anyAttribute namespace="##any" processContents="lax"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
</xs:element>

<xs:element name="user-input">
  <xs:annotation>
    <xs:documentation>
      Records the user-input or usage state of the service or
      device.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="activeIdle">
        <xs:attribute name="idle-threshold"
          type="xs:positiveInteger"/>
        <xs:attribute name="last-input" type="xs:dateTime"/>
        <xs:attribute name="id" type="xs:ID"/>
        <xs:anyAttribute namespace="##any"
          processContents="lax"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
</xs:schema>

```

## 6. Extending RIPID

Any developer can introduce their own element names, avoiding conflict by choosing an appropriate namespace URI. To add new standardized elements to the enumerations <activities>, <mood>, <privacy>, <relationship> and <service-class>, the extension process described in PIDF [9] is followed, i.e., such extensions would use namespace designators such as urn:ietf:params:xml:ns:pidf:ext, where 'ext' is the name of the extension. Any new values for the <place-type> element are assigned according to [12] and are given a namespace designator at their time of registration.

To avoid the unnecessary proliferation of XML namespaces containing a single element, groups of element registrations for each of these enumerations, such as <privacy>, SHOULD be bundled into a single namespace rather than assigning a new namespace to each new element.

## 7. IANA Considerations

### 7.1. URN Sub-Namespace Registration for 'urn:ietf:params:xml:ns:pidf:rpid'

URI: urn:ietf:params:xml:ns:pidf:rpid

Description: This is the XML namespace for XML elements defined by RFC 4480 to describe rich presence information extensions for the status element in the PIDF presence document format in the application/pidf+xml content type.

Registrant Contact: IETF, SIMPLE working group, simple@ietf.org,  
Henning Schulzrinne, hgs@cs.columbia.edu

XML:

BEGIN

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
"http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>RPID: Rich Presence Extensions to the Presence
    Information Data Format (PIDF)</title>
</head>
<body>
  <h1>Namespace for rich presence extension</h1>
  <h2>urn:ietf:params:xml:ns:pidf:rpid</h2>
  <p>See <a href="http://www.rfc-editor.org/rfc/rfc4480.txt">
    RFC&4480;</a>.</p>
</body>
</html>
END
```

## 7.2. Schema Registration for Schema

`'urn:ietf:params:xml:ns:pidf:status:rpid'`

URI: `urn:ietf:params:xml:ns:pidf:status:rpid`

Registrant Contact: IESG

XML: See Section 5

Note that this document does not need a new content type. It inherits the content type from [8], namely, `application/pidf+xml`.

## 8. Internationalization Considerations

RPID contains mostly tokens that are meant for consumption by programs, not directly by humans. Programs are expected to translate those tokens into language-appropriate text strings according to the preferences of the watcher.

Some elements may contain `<note>` and `<other>` elements that can contain free text. These elements SHOULD be labeled with the `'xml:lang'` attribute to indicate their language and script. The specification allows multiple occurrences of these elements so that the presentity can convey `<note>` and `<other>` elements in multiple scripts and languages. If no `'xml:lang'` attribute is provided, the default value is "i-default" [3].

Since RPID is represented in XML, it provides native support for encoding information using the Unicode character set and its more compact representations including UTF-8. Conformant XML processors recognize both UTF-8 and UTF-16. Though XML includes provisions to identify and use other character encodings through use of an "encoding" attribute in an `<?xml?>` declaration, use of UTF-8 is RECOMMENDED in environments where parser encoding support incompatibility exists.

A description of time-zone considerations can be found in Section 3.13.

## 9. Security Considerations

The security considerations in [8] apply, as well as [7]. Compared to PIDF, this presence document format reveals additional information about presentities that can be highly sensitive. Beyond traditional security measures to protect confidentiality and integrity, systems should offer a means to selectively reveal information to particular watchers and to inspect the information that is being published, particularly if it is generated automatically from other sources, such as calendars or sensors.

Like any reference to an external object, the <status-icon> may allow the presentity to induce the watcher to retrieve data from a third party (content indirection attack), thus either retrieving harmful content or adding to the server load of the referenced resource.

## 10. References

### 10.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [2] Moats, R., "URN Syntax", RFC 2141, May 1997.
- [3] Alvestrand, H., "IETF Policy on Character Sets and Languages", BCP 18, RFC 2277, January 1998.
- [4] Moats, R., "A URN Namespace for IETF Documents", RFC 2648, August 1999.
- [5] Day, M., Rosenberg, J., and H. Sugano, "A Model for Presence and Instant Messaging", RFC 2778, February 2000.
- [6] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [7] Rosenberg, J., "A Presence Event Package for the Session Initiation Protocol (SIP)", RFC 3856, August 2004.
- [8] Sugano, H., Fujimoto, S., Klyne, G., Bateman, A., Carr, W., and J. Peterson, "Presence Information Data Format (PIDF)", RFC 3863, August 2004.
- [9] Yergeau, F., Paoli, J., Sperberg-McQueen, C., Bray, T., and E. Maler, "Extensible Markup Language (XML) 1.0 (Third Edition)", W3C REC REC-xml-20040204, February 2004.
- [10] Maloney, M., Beech, D., Thompson, H., and N. Mendelsohn, "XML Schema Part 1: Structures Second Edition", W3C REC REC-xmlschema-1-20041028, October 2004.
- [11] Malhotra, A. and P. Biron, "XML Schema Part 2: Datatypes Second Edition", W3C REC REC-xmlschema-2-20041028, October 2004.
- [12] Schulzrinne, H. and H. Tschofenig, "Location Types Registry", RFC 4589, July 2006.

## 10.2. Informative References

- [13] Dawson, F. and D. Stenerson, "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", RFC 2445, November 1998.
- [14] Peterson, J., "Common Profile for Instant Messaging (CPIM)", RFC 3860, August 2004.
- [15] Lennox, J., Wu, X., and H. Schulzrinne, "Call Processing Language (CPL): A Language for User Control of Internet Telephony Services", RFC 3880, October 2004.
- [16] Rosenberg, J., "A Data Model for Presence", RFC 4479, July 2006.
- [17] Lisetti, C., "Personality, Affect, and Emotion Taxonomy for Socially Intelligent Agents", Proceedings of FLAIRS 2002, 2002.
- [18] Open Mobile Alliance, "The Wireless Village Initiative: Presence Attributes 1.1", Recommendation WV-29, 2004.

## Appendix A. Acknowledgements

The document reflects the discussion on the SIMPLE mailing list, with contributions from many individuals. David L. Black, Miguel Garcia, Avshalom Houri, Markus Isomaki, Rick Jones, Hisham Khartabil, Jonathan Lennox, Eva-Maria Leppanen, Mikko Lonnfors, Rohan Mahy, Miguel Marcia, Andrew Newton, Aki Niemi, Jon Peterson, and Brian Rosen provided detailed comments and suggestions. Xiaotao Wu assisted with schema testing. Jari Urpalainen provided valuable advice on XML schema issues.

## Authors' Addresses

Henning Schulzrinne  
Columbia University  
Department of Computer Science  
450 Computer Science Building  
New York, NY 10027  
US

Phone: +1 212 939 7042  
EMail: [hgs+simple@cs.columbia.edu](mailto:hgs+simple@cs.columbia.edu)  
URI: <http://www.cs.columbia.edu>

Vijay Gurbani  
Lucent  
2000 Naperville Rd.  
Room 6G-440  
Naperville, IL 60566-7033  
US

EMail: [vkg@lucent.com](mailto:vkg@lucent.com)

Paul Kyzivat  
Cisco Systems  
BXB500 C2-2  
1414 Massachusetts Avenue  
Boxborough, MA 01719  
US

EMail: [pkyzivat@cisco.com](mailto:pkyzivat@cisco.com)

Jonathan Rosenberg  
Cisco Systems  
600 Lanidex Plaza  
Parsippany, NJ 07054-2711  
US

EMail: [jdrosen@cisco.com](mailto:jdrosen@cisco.com)

## Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM 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.

## Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

## Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

