

Groupe de travail Réseau
Request for Comments : 3863
 Catégorie : En cours de normalisation
 Traduction Claude Brière de L'Isle
 août 2004

H. Sugano & S. Fujimoto, Fujitsu
 G. Klyne, Nine by Nine
 A. Bateman, VisionTech
 W. Carr, Intel
 J. Peterson, NeuStar

Format de données pour les informations de présence (PIDF)

Statut du présent mémoire

Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Protocoles officiels de l'Internet" (STD 1) pour voir l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de Copyright

Copyright (C) The Internet Society (2004).

Résumé

Le présent mémoire spécifie le profil commun de présence (CPP, *Common Profile for Presence*) du format des données d'information de présence (PIDF, *Presence Information Data Format*) comme format commun de données de présence pour les protocoles de présence conformes à CPP, et définit aussi un nouveau type de support "application/pidf+xml" pour représenter l'entité MIME XML pour PIDF.

Table des Matières

| | |
|---|----|
| 1. Introduction..... | 1 |
| 1.1 Terminologie et conventions..... | 2 |
| 2. Décisions de conception..... | 2 |
| 2.1 Modèle minimal..... | 2 |
| 2.2 Caractéristiques ajoutées..... | 3 |
| 2.3 Décision de codage XML..... | 3 |
| 3. Généralités sur le format des informations de présence..... | 3 |
| 3.1 Type de contenu 'application/pidf+xml'..... | 3 |
| 3.2 Contenu des informations de présence..... | 3 |
| 4. Format des données de présence codées en XML..... | 4 |
| 4.1 Définitions du format XML..... | 4 |
| 4.2 Extensibilité des informations de présence..... | 6 |
| 4.3 Exemples..... | 9 |
| 4.4 Définitions de schéma XML..... | 10 |
| 5. Considérations relatives à l'IANA..... | 12 |
| 5.1 Enregistrement du type de contenu pour 'application/pidf+xml'..... | 12 |
| 5.2 Enregistrement du sous espace de nom d'URN pour 'urn:ietf:params:xml:ns:pidf'..... | 12 |
| 5.3 Enregistrement du sous espace de nom d'URN pour 'urn:ietf:params:xml:ns:pidf:status'..... | 13 |
| 6. Considérations pour la sécurité..... | 13 |
| 7. Considérations d'internationalisation..... | 14 |
| 8. Références..... | 14 |
| 8.1 Références normatives..... | 14 |
| 8.2 Références pour information..... | 15 |
| Appendice A. Définitions de type de document..... | 15 |
| Adresse des auteurs..... | 16 |
| Déclaration complète de droits de reproduction..... | 16 |

1. Introduction

Les spécifications des profils communs pour la messagerie instantanée (CPIM, *Common Profiles for Instant Messaging*) [RFC3860] et du profil commun de présence (CPP, *Common Profile for Presence*) [RFC3859] définissent un ensemble d'opérations et de paramètres pour réaliser l'interopérabilité entre différents protocoles de messagerie instantanée et de présence qui satisfont à la [RFC2779].

Le présent mémoire définit un peu plus le format des données d'information de présence (PIDF, *Presence Information Data Format*) comme un format commun de données de présence pour les protocoles de présence conformes à CPP, permettant aux informations de présence d'être transférées sans modification à travers les frontières d'un protocole conforme à CPP, en conservant le bénéfice de la sécurité et des performances.

Le format spécifié dans le présent mémoire définit le format de présence de base et l'extensibilité exigée par la RFC2779. Il définit un ensemble minimal de valeurs d'état de présence défini par le document Modèle IMPP [RFC2778]. Cependant, une application de présence est capable de définir ses propres valeurs d'état en utilisant le cadre d'extensibilité fourni par le présent mémoire. La définition de telles valeurs d'état étendues sort du domaine d'application du présent mémoire.

Noter aussi que ce mémoire ne définit que le format d'une charge utile de données de présence et son cadre d'extensibilité. Comment les données de présence sont transférées au sein d'une trame d'un protocole spécifique sera défini séparément dans une spécification de protocole.

1.1 Terminologie et conventions

Le présent mémoire utilise le vocabulaire défini dans le document "Modèle IMPP" [RFC2778]. Les termes comme Fermé, Message Instantané, Ouvert, Service de Présence, Présentité, Observateur, et Agent d'utilisateur observateur ont la même signification que celle qui y est définie.

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans le BCP 14, [RFC 2119].

2. Décisions de conception

On a adopté les documents de modèle et d'exigences IMPP [RFC2778], [RFC2779] comme point de départ de notre discussion. Les deux RFC contiennent un certain nombre de déclarations sur les informations de présence, qui peuvent être considérées comme l'ensemble des contraintes de base pour la conception du format. Aussi, on a pris l'approche minimaliste pour la conception qui se fonde sur elles. En commençant par le modèle minimal, seules les caractéristiques qui sont nécessaires pour résoudre des problèmes particuliers ont été incluses.

2.1 Modèle minimal

La présente spécification se fonde sur le modèle minimal extrait des documents sur le modèle et les exigences de IMPP. Le modèle consiste en les éléments qui suivent. Chacun d'eux est accompagné des RFC correspondantes et des numéros de leurs paragraphes, par exemple, (RFC2778:Sec.2.4) se réfère au paragraphe 2.4 de la RFC2778.

- (a) Informations de présence consiste en un ou plusieurs tuplets de présence, où un tuple de présence consiste en un État, une Adresse de Communication facultative, et une Balise d'Autre Présence facultative. Noter que l'adresse de contact dans une adresse de communications est comprise dans le présent document comme se référant seulement à un URI (RFC2778:Sec.3).
- (b) État a au moins les valeurs mutuellement exclusives Ouvert et Fermé, qui ont une signification pour l'acceptation des Messages Instantanés, et peuvent avoir une signification pour d'autres Moyens de Communication. Il peut y avoir d'autres valeurs d'État qui n'ont aucune implication sur l'acceptation des Messages Instantanés. Ces autres valeurs de État peuvent être combinées avec Ouvert et Fermé ou elles peuvent être mutuellement exclusives avec ces valeurs (RFC2778:Sec.3, RFC2779:Sec.4.4.1- 4.4.3).
- (c) État peut consister en une seule valeur ou en plusieurs. (RFC2778:Sec.2.4)
- (d) Il doit y avoir un moyen pour étendre le format commun de présence pour qu'il représente des informations supplémentaires non incluses dans le format commun. Les mécanismes d'extension et d'enregistrement doivent être définis pour le schéma d'informations de présence, incluant de nouvelles conditions d'État et de nouvelles formes pour des Balises d'Autre Présence (RFC2779:Sec.3.1.4-3.1.5).
- (e) Le format commun de présence doit inclure un moyen pour identifier de façon univoque la Présentité dont les

Informations de Présence sont rapportées (RFC2779:Sec.3.1.2).

- (f) Le format commun de présence doit permettre à la Présentité de sécuriser les informations de présence envoyées à un Observateur. Le format doit permettre d'appliquer les propriétés d'intégrité, de confidentialité et d'authentification aux informations de présence (RFC2779:Sec5.2.1, 5.2.4, 5.3.1, 5.3.3).

2.2 Caractéristiques ajoutées

En plus du modèle minimal décrit ci-dessus, le format spécifié dans le présent mémoire a les caractéristiques suivantes.

- (a) Les priorités relatives des adresses de contact sont spécifiables pour permettre à la source des Informations de Présence de dire au receveur (Agent d'Utilisateur Observateur) ses préférences entre plusieurs moyens de contact.
- (b) Le format de présence est capable de contenir les horodatages de la création des Informations de Présence. L'horodatage dans le document de présence permet au receveur de connaître l'heure de la création des données même si le message qui le contient est retardé. Il peut aussi être utilisé pour détecter une attaque en répétition, indépendamment du mécanisme de signature sous-jacent. Noter que ce mécanisme ne suppose aucun système mondial de synchronisation horaire pour les observateurs et les présentités (voir l'Appendice A de la RFC2779, 8.1.4 A7) mais suppose plutôt que la durée minimum qui pourrait s'écouler avant que les informations de présence soient considérées périmées est assez longue pour que des variations mineures entre les horloges système ne conduisent pas à des erreurs de jugement sur la fraîcheur des informations de présence .

2.3 Décision de codage XML

Le format des données d'informations de présence code les informations de présence en XML (langage de balisage extensible [XML]). Concernant les caractéristiques des Informations de Présence discutées ci-dessus, comme le fait qu'il ait une structure hiérarchique et devrait être pleinement extensible, XML est considéré comme le cadre le plus souhaitable par rapport à d'autres candidats comme vCard [RFC2426].

3. Généralités sur le format des informations de présence

Cette section décrit les généralités du format des données de présence défini dans le présent mémoire.

3.1 Type de contenu 'application/pidf+xml'

Le présent mémoire définit un nouveau type de contenu "application/pidf+xml" pour une entité MIME XML qui contient des informations de présence. La présente spécification suit les recommandations et conventions décrites dans la [RFC3023], incluant la convention de désignation du type ('+xml' suffix) et l'usage du paramètre 'charset'.

Bien qu'elle soit définie comme facultative, l'utilisation du paramètre 'charset' est RECOMMANDÉE. Si le paramètre 'charset' n'est pas spécifié, les processeurs conformes à XML DOIVENT suivre les exigences du paragraphe 4.3.3 de [XML].

3.2 Contenu des informations de présence

Ce paragraphe souligne les informations d'un document "application/pidf+xml". Une définition complète du contenu PIDF figure à la Section 4.

- o URL de Présentité : spécifie l'URL "pres" de la présentité.
- o Liste des Tuplets Présence
 - Identifiant : jeton pour identifier ce tuple dans les informations de présence.
 - État : Ouvert/Fermé et/ou valeurs d'état d'extension.
 - Adresse de Communication : Moyens de Communication et Adresse de Contact de ce tuple (facultatif).
 - Priorité relative : valeur numérique qui spécifie la priorité de cette Adresse de Communication (facultatif)
 - Horodatage : horodatage du changement de ce tuple (facultatif).
 - Commentaire lisible par l'homme : texte libre concernant ce tuple (facultatif).
- o Commentaire de Présentité lisible par l'homme : texte libre sur la Présentité (facultatif).

4. Format des données de présence codées en XML

Cette section définit un format des données d'information de présence codées en XML (PIDF, *presence information data format*) à utiliser avec les systèmes conformes à CPP. La production d'une charge utile de présence dans ce format est attendue de la part de la Présentité (la source des Informations de Présence) et elle est transportée aux Observateurs par les serveurs ou passerelles de présence sans aucune interprétation ni modification.

4.1 Définitions du format XML

Un objet PIDF est un document XML bien formé.

Il DOIT avoir la déclaration XML et il DEVRAIT contenir une déclaration de codage dans la déclaration XML, par exemple, "<?xml version='1.0' encoding='UTF-8'?>". Si le paramètre charset de la déclaration de type de contenu MIME est présent et si il est différent de la déclaration de codage, c'est le paramètre charset qui a la préséance.

Toute application conforme à la présente spécification DOIT accepter le codage de caractères UTF-8 pour assurer une interopérabilité minimale.

4.1.1 Élément <presence>

Les éléments PIDF sont associés au nom d'espace de noms XML 'urn:ietf:params:xml:ns:pidf', déclaré en utilisant un attribut xmlns, selon [XML-NS]. L'espace de noms peut être un espace de noms par défaut, ou il peut être associé à un préfixe d'espace de noms (voir des exemples au paragraphe 4.2.2).

La racine d'un objet "application/pidf+xml" est un élément <presence> associé à l'espace de nom des informations de présence. Elle contient un nombre quelconque (y compris 0) d'éléments <tuple>, suivi par un nombre quelconque (y compris 0) d'éléments <note>, suivi par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms.

L'élément <presence> DOIT avoir un attribut 'entité'. La valeur de l'attribut 'entité' est l'URL 'pres' de la Présentité qui publie ce document de présence.

L'élément <presence> DOIT contenir une déclaration d'espace de nom ('xmlns') pour indiquer l'espace de noms sur lequel se fonde le document de présence. Le document de présence conforme à la présente spécification DOIT avoir l'espace de noms 'urn:ietf:params:xml:ns:pidf:'.

Il PEUT contenir d'autres déclarations d'espace de noms pour les extensions utilisées dans le document XML présence.

4.1.2 Élément <tuple>

L'élément <tuple> porte un Tuplet Présence qui consiste en un élément <status> obligatoire, suivi par un nombre quelconque d'éléments d'extension facultatifs, éventuellement provenant d'autres espaces de noms) suivi par un élément facultatif <contact>, suivi par un nombre quelconque d'éléments facultatifs <note>, suivi par un élément <horodatage> facultatif.

Les tuplets fournissent un moyen pour segmenter les informations de présence. Les protocoles ou applications peuvent choisir de segmenter les informations de présence associées à une présentité pour toutes sortes de raisons - par exemple, parce que les composants des informations de présence complètes pour une présentité viennent d'un appareil distinct ou de différentes applications sur le même appareil, ou ont été générés à des moments différents. Les tuplets devraient être préférés aux autres manières de segmenter les informations de présence comme la création de multiples instances PIDF.

L'élément <tuple> DOIT contenir un attribut 'id' qui est utilisé pour distinguer ce tuple des autres dans la même Présentité. La valeur d'un attribut 'id' DOIT être unique parmi les valeurs d'attribut 'id' des autres tuplets pour la même Présentité. Une valeur 'id' est utilisée par les applications qui traitent le document de présence pour identifier le tuple correspondant dans les Informations de Présence acquises précédemment de la même Présentité. La valeur de l'attribut 'id' est une chaîne arbitraire, et n'a pas de signification autre que de fournir le moyen de distinguer les valeurs de <tuple>, comme noté ci-dessus.

L'élément <contact> est facultatif parce que une Présentité peut avoir besoin de cacher son Adresse de Communication ou il peut y avoir des tuplets sans relation avec aucun Moyen de Communication. Les tuplets qui contiennent un élément d'état <basique> DEVRAIENT contenir une adresse <contact>. Les tuplets PEUVENT contenir des états de présence conflictuels - un <tuplet> peut fournir un <état> <basique> de Ouvert, et un autre <tuplet> dans le même PIDF pourrait contenir un <état> <basique> de Fermé, même si ils contiennent tous deux la même adresse de <contact>.

La manière dont les informations de présence segmentées sont comprises par l'Agent d'Utilisateur Observateur dépend étroitement des capacités de celui-ci et de l'application de présence en question. En l'absence de toute compréhension spécifique de l'application ou du protocole de la signification des tuplets, les agents d'utilisateur observateur PEUVENT suivre les lignes directrices suivantes : ils devraient noter quels tuplets dans le PIDF ont changé leur état depuis la dernière notification en corrélant le 'id' de chaque <tuplet> avec ceux reçus dans les notifications précédentes et en comparant les deux valeurs <état> et <horodatage> dans les tuplets, si il en est de présentes.

4.1.3 Élément <état>

L'élément <état> contient un élément <basique> facultatif, suivi par un nombre quelconque d'éléments d'extension facultatifs (provenant éventuellement d'autres espaces de noms) avec la restriction qu'au moins un élément fils apparaisse dans l'élément <état>. Ces éléments fils de <état> contiennent des valeurs d'état de ce tuplet. En permettant plusieurs valeurs d'état dans un seul élément <tuplet>, différents types de valeurs d'état, par exemple, accessibilité et localisation, peuvent être représentées par un <tuplet>. Voir au paragraphe 4.3 un exemple de valeurs d'état multiples.

Le présent mémoire définit seulement l'élément de valeur d'état <basique>. Les autres valeurs d'état peuvent être incluses en utilisant le cadre standard d'extensibilité (voir le paragraphe 4.2.4). Les applications qui rencontrent des éléments non reconnus au sein de <état> peuvent les ignorer, sauf si ils portent un attribut `mustUnderstand="true"` ou `mustUnderstand="1"` (voir le paragraphe 4.2.3).

Noter que, bien que l'élément <état> DOIVE avoir au moins un élément de valeur d'état, cette valeur d'état peut n'être pas l'élément <basique>.

4.1.4 Élément <basique>

L'élément <basique> contient une des chaînes suivantes : "ouvert" ou "fermé".

Les valeurs "ouvert" et "fermé" indiquent la disponibilité pour recevoir des messages instantanés si le <tuplet> est pour une adresse de messagerie instantanée. Elles indiquent aussi la disponibilité générale pour d'autres moyens de communication, mais le présent mémoire n'en spécifie par le détail.

ouvert : dans le contexte des messages instantanés, cette valeur signifie que l'élément <contact> associé, s'il en est, correspond à une Boîte aux lettres instantanée qui est prête à accepter un Message Instantané.

fermé : dans le contexte des messages instantanés, cette valeur signifie que l'élément <contact> associé, s'il en est, correspond à une boîte aux lettres instantanée qui n'est pas capable d'accepter un message instantané.

4.1.5 Élément <contact>

L'élément <contact> contient un URL de l'adresse de contact. Il a facultativement un attribut 'priorité' dont la valeur signifie une priorité relative de cette adresse de contact sur les autres. La valeur de l'attribut DOIT être un nombre décimal entre 0 et 1 inclus avec au plus 3 chiffres après la virgule. Les valeurs supérieures indiquent une priorité supérieure. Des exemples de valeurs de priorité sont 0, 0,021, 0,5, 1,00. Si l'attribut 'priorité' est omis, les applications DOIVENT allouer l'adresse de contact de la plus faible priorité. Si la valeur de 'priorité' est hors de la gamme, les applications DEVRAIENT juste ignorer la valeur et la traiter comme si l'attribut n'était pas présent.

Les applications DEVRAIENT traiter les contacts qui ont une priorité supérieure comme si ils avaient la préséance sur ceux qui ont une priorité inférieure. Comment ils sont réellement traités sort du domaine d'application de cette spécification. Aussi, comment traiter les contacts avec la même priorité dépend de la mise en œuvre.

4.1.6 Élément <note>

L'élément <note> contient une valeur de chaîne, qui est usuellement destinée à un commentaire lisible par l'homme. L'élément <note> PEUT apparaître comme un élément fils de <présence> ou comme un élément fils de l'élément <tuplet>. Dans le premier cas, le commentaire porte sur la Présentité et dans le second cas, le commentaire concerne le tuplet particulier.

Noter que chaque fois qu'il apparaît, un élément <note> NE DEVRAIT PAS être utilisé, et interprété, comme un substitut non interopérable de l'état de son élément parent.

L'élément <note> DEVRAIT avoir un attribut 'xml:lang' spécial pour spécifier le langage utilisé dans le contenu de cet élément comme défini au paragraphe 2.12 de [XML]. La valeur de cet attribut est l'identifiant du langage défini par la [RFC3066]. Il PEUT être omis lorsque le langage utilisé est impliqué par un contexte plus large comme les informations de codage du contenu, comme un attribut xml:lang sur un élément XML qui l'inclut, ou un en-tête Langage du contenu [RFC3282] sur une enveloppe MIME.

4.1.7 Élément <timestamp>

L'élément <timestamp> contient une chaîne qui indique la date et l'heure du changement d'état de ce tuple. La valeur de cet élément DOIT suivre le format IMPP datetime [RFC3339]. Les horodatages qui contiennent 'T' ou 'Z' DOIVENT utiliser des formes majuscules.

Comme mesure de sécurité, l'élément <timestamp> DEVRAIT être inclus dans tous les tuples sauf si l'heure exacte du changement d'état ne peut pas être déterminée. Pour les lignes directrices de sécurité pour les observateurs qui reçoivent des informations de présence avec des horodatage, voir les Considérations sur la sécurité (Section 6).

Une Présence NE DOIT PAS générer des éléments <presence> successifs contenant le même horodatage.

4.2 Extensibilité des informations de présence

Le cadre d'extensibilité des informations de présence se fonde sur l'espace de noms XML [XML-NS]. La RFC2779 exige que PIDF ait un moyen d'étendre les valeurs de <status> au delà de <basique>. Ces extensions NE DOIVENT PAS modifier la façon dont <basique> est compris, ni changer la structure ou la sémantique des corps PIDF eux-mêmes. Ces extensions permettent simplement aux protocoles et applications de définir des données de présence plus riches.

4.2.1 Fondements des espaces de nom XML

Tous les éléments et certains attributs sont associés à un "espace de noms", qui à son tour est associé à un URI unique au monde. Tout développeur peut introduire ses propres noms d'élément, en évitant les conflits par le choix d'un URI d'espace de noms approprié.

Dans les données de présence, les noms d'élément ou d'attribut sont associés à un espace de noms particulier par un préfixe d'espace de noms, qui est la première partie du nom, suivi par deux points (":"); par exemple,

```
<préfixe:nom-d'élément ...> ... </préfixe:nom-d'élément>
```

Où 'préfixe' est le préfixe du nom d'en-tête, 'nom-d'élément' est un nom qui a sa portée délimitée par l'espace de noms associé à 'préfixe'. Noter que le choix de 'préfixe' est assez arbitraire ; c'est l'URI correspondant qui définit la portée de la dénomination. Deux préfixes différents associés au même URI d'espace de nom se réfèrent au même espace de noms.

Un espace de noms par défaut peut être déclaré pour des éléments XML sans préfixe d'espace de noms. L'espace de noms par défaut NE s'applique PAS aux noms d'attribut, mais l'interprétation d'un attribut sans préfixe peut être déterminée par les éléments qui le contiennent.

Un espace de noms est identifié par un URI. Dans cet usage, l'URI est utilisé simplement comme un identifiant unique au monde, et il n'est pas exigé qu'il puisse être utilisé pour restituer une ressource de la Toile, ou pour aucun autre objet. Tout URI légal unique au monde PEUT être utilisé pour identifier un espace de noms. (Par "unique au monde", on veut dire construit conformément à un ensemble de règles afin qu'il soit raisonnable de s'attendre à ce que personne d'autre n'utilise le même URI pour un objet différent.)

Pour les détails, voir la spécification d'espace de noms XML [XML-NS].

4.2.2 Espaces de nom XML dans les informations de présence

Un URI utilisé comme identifiant d'espace de noms dans des données d'Informations de Présence DOIT être un URI absolu complet, selon la [RFC2396]. (Les URI relatifs et les références d'URI qui contiennent des fragments d'identifiants NE DOIVENT PAS être utilisés à cette fin.)

L'URI d'espace de noms pour les éléments définis dans la présente spécification est un URN [RFC2141], qui utilise l'identifiant d'espace de nom de 'ietf' défini par la [RFC2648] et étendue par la [RFC3688] :

```
urn:ietf:params:xml:ns:pidf
```

Donc, les données de simple présence peuvent être :

```
<?xml version="1.0" encoding="UTF-8"?>
<impp:presence xmlns:impp="urn:ietf:params:xml:ns:pidf"
  entity="pres:someone@example.com">
  <impp:tuple id="sg89ae">
    <impp:status>
      <impp:basic>open</impp:basic>
    </impp:status>
    <impp:contact priority="0.8">tel:+09012345678</impp:contact>
  </impp:tuple>
</impp:presence>
```

Si on utilise un espace de noms XML par défaut :

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  entity="pres:someone@example.com">
  <tuple id="sg89ae">
    <status>
      <basic>open</basic>
    </status>
    <contact priority="0.8">tel:+09012345678</contact>
  </tuple>
</presence>
```

Comme c'est généralement le cas dans XML avec les espaces de noms, l'attribut xmlns peut être utilisé sur tout élément dans les informations de présence pour définir soit l'espace de noms par défaut, soit un espace de noms associé à un préfixe d'espace de noms.

4.2.3 Traitement des noms d'éléments non reconnus

Sauf comme noté ci-dessus, un processeur d'Informations de Présence DOIT ignorer tout élément XML avec un nom non reconnu (c'est-à-dire, qui a un URI d'espace de noms non reconnu, ou un nom local non reconnu au sein de cet espace de noms). Cela inclut tous les contenus d'élément, même si il paraît contenir des éléments avec des noms reconnus.

Les extensions à PIDF sont informationnelles par nature – elles fournissent des informations supplémentaires au delà de l'état <basique>. Cependant, afin de comprendre une extension complexe, des éléments incorporés au sein d'un élément d'extension peuvent devoir être marqués comme obligatoires. Dans ce cas, le nom de l'élément est qualifié avec un attribut mustUnderstand='true' (*doit comprendre = vrai*) ou mustUnderstand='1'. Voir un exemple au paragraphe 4.3.3.

Note : un attribut mustUnderstand='true' ou mustUnderstand='1' au sein d'un élément qui est ignoré est lui-même ignoré. L'auteur d'information incorporées de compréhension obligatoire est chargé de s'assurer que tout élément incorporant est aussi étiqueté avec un attribut mustUnderstand='true' ou mustUnderstand='1', si nécessaire.

La présente spécification définit (paragraphe 4.1) des éléments au sein de l'espace de noms 'urn:ietf:params:xml:ns:pidf' qui DOIVENT être reconnus dans les données de présence CPP. Les processeurs DOIVENT les traiter comme décrit, même si ils ne portent pas un attribut mustUnderstand. La définition de schéma XML (paragraphe 4.4) indique les éléments qui DOIVENT être présents dans un document valide d'informations de présence.

Si un agent reçoit des Informations de Présence avec un bloc <état> contenant un élément non reconnu avec un attribut mustUnderstand='true' (ou '1') il devrait traiter cet élément entier et tout son contenu comme non reconnu et ne pas tenter de le traiter.

Afin de s'assurer que les mises en œuvre minimales peuvent traiter correctement les informations PIDF de base, l'attribut mustUnderstand DOIT être utilisé seulement au sein des éléments facultatifs incorporés dans un élément <état>. Cela va assurer que les problèmes de traitement d'une extension se restreignent à cette extension et n'affectent pas le traitement des

informations PIDF de base définies dans la présente spécification.

4.2.4 Extensibilité de valeur d'état

Le présent mémoire définit seulement la valeur de l'état <basique> avec les valeurs de "ouvert" et "fermé". D'autres valeurs d'état sont possibles en utilisant les règles standard d'extensibilité fondées sur l'espace de noms définies ci-dessus.

Par exemple, une valeur d'état de localisation pourrait être incluse ainsi :

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:local="urn:example-com:pidf-status-type"
  entity="pres:someone@example.com">
  <tuple id="ub93s3">
    <status>
      <basic>open</basic>
      <local:location>home</local:location>
    </status>
    <contact>im:someone@example.com</contact>
  </tuple>
</presence>
```

Certaines nouvelles valeurs d'état vont 'étendre' la valeur de l'élément <basique>. Par exemple, une valeur d'état définie pour être utilisée avec la messagerie instantanée peut inclure des valeurs telles que 'away' (*absent*), 'busy' (*occupé*) et 'offline' (*hors ligne*). Afin qu'un certain niveau d'interopérabilité soit maintenu avec les agents d'utilisateur qui ne reconnaissent pas les nouvelles extensions, la valeur d'état <basique> doit aussi être incluse. Cela signifie que les extensions ne sont pas obligées de définir une transposition de chacune de leurs valeurs en Ouvert ou Fermé.

4.2.5 Normalisation des extensions d'état

Bien que la définition existante de PIDF permette que des éléments arbitraires apparaissent dans l'élément <état>, il peut parfois être souhaitable de normaliser les éléments d'état d'extension et leur sémantique (la signification des états particuliers, comment ils devraient être interprétés). L'URN 'urn:ietf:params:xml:ns:pidf:status' a été spécifié comme espace de nom parapluie sous lequel des extensions à l'élément <état> PIDF devraient être spécifiées (par exemple, urn:ietf:params:xml:ns:pidf:status:my-extension). Les nouvelles valeurs sous cet espace de noms DOIVENT être définies par une RFC en cours de normalisation.

L'exemple suivant de schéma XML définit une extension pour les informations de présence <location> (*localisation*), qui peut avoir les valeurs de 'home' (*à la maison*), 'office' (*au bureau*), ou 'car' (*en voiture*). Si l'élément <location> était normalisé, le présent document serait rendu disponible dans une RFC avec les informations sur l'utilisation de l'extension. Ces extensions devraient utiliser l'espace de noms 'urn:ietf:params:xml:ns:pidf:status', et chaque RFC qui définit une extension devrait enregistrer un nom d'extension dans cet espace de noms auprès de l'IANA.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:ietf:params:xml:ns:pidf:status"
  xmlns:tns="urn:ietf:params:xml:ns:pidf:status"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:simpleType name="location">
    <xs:restriction base="xs:string">
      <xs:enumeration value="home"/>
      <xs:enumeration value="office"/>
      <xs:enumeration value="car"/>
    </xs:restriction>
  </xs:simpleType>

</xs:schema>
```

En plus du schéma XML pour valider l'extension, l'enregistrement du nom de l'extension auprès de l'IANA, les RFC qui définissent des extensions DOIVENT préciser :

- Le domaine d'applicabilité de l'extension. Cette extension est elle exclusivement valable pour les clients IM, les téléphones, géolocalisateurs, etc. ? Quelles sortes d'applications de présence utiliseraient cette extension et dans quelles circonstances ?
- La sémantique des états de présence définie dans l'extension. Quelle disposition provoque la déclaration automatique par une présentité qu'elle est dans l'état X, ou est-ce qu'une personne a choisi X à partir d'un menu déroulant ? Y a-t il des lignes directrices générales pour les observateurs des informations de présence pour l'état Y (par exemple, comment ils peuvent le mieux tenter de communiquer avec la présentité, si c'est possible, lorsque le principal est dans l'état Y ?).

Les extensions DEVRAIENT aussi préciser :

- Comment, si c'est possible, un état de présence défini dans l'extension se rapporte à <basique>, ou à toute extension pertinente précédemment publiée dans une RFC. Par exemple, "l'état Z implique Ouvert, de sorte qu'il NE DOIT PAS être utilisé si un état de base de Fermé est exprimé", ou "vous devriez utiliser l'extension de ce document, et non l'extension de la RFCQQQQ, si les circonstances sont les suivantes...."

4.3 Exemples

4.3.1 Espace de noms par défaut avec les extensions d'état

L'instance de document suivante utilise un espace de noms hypothétique 'pidf:im' XML comme exemple de la sorte d'extension d'état qui peut être développée pour PIDF.

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:im="urn:ietf:params:xml:ns:pidf:im"
  xmlns:myex="http://id.example.com/presence/"
  entity="pres:someone@example.com">
  <tuple id="bs35r9">
    <status>
      <basic>open</basic>
      <im:im>busy</im:im>
      <myex:location>home</myex:location>
    </status>
    <contact priority="0.8">im:someone@mobilecarrier.net</contact>
    <note xml:lang="en">Don't Disturb Please!</note>
    <note xml:lang="fr">Ne pas déranger, s'il vous plait</note>
    <timestamp>2001-10-27T16:49:29Z</timestamp>
  </tuple>
  <tuple id="eg92n8">
    <status>
      <basic>open</basic>
    </status>
    <contact priority="1.0">mailto:someone@example.com</contact>
  </tuple>
  <note>Je serai à Tokyo la semaine prochaine</note>
</presence>
```

4.3.2 Presence avec d'autres éléments d'extension

```
<?xml version="1.0" encoding="UTF-8"?>
<impp:presence xmlns:impp="urn:ietf:params:xml:ns:pidf"
  xmlns:myex="http://id.example.com/presence/"
  entity="pres:someone@example.com">
  <impp:tuple id="ck38g9">
    <impp:status>
      <impp:basic>open</impp:basic>
    </impp:status>
    <myex:mytupletag>Extended value in tuple</myex:mytupletag>
    <impp:contact priority="0.65">tel:+09012345678</impp:contact>
  </impp:tuple>
  <impp:tuple id="md66je">
```

```

<impp:status>
  <impp:basic>open</impp:basic>
</impp:status>
<impp:contact priority="1.0">
  im:someone@mobilecarrier.net</impp:contact>
</impp:tuple>
<myex:mytag>Mes informations de présentité étendues</myex:mytag>
</impp:presence>

```

4.3.3 Exemple d'éléments de compréhension obligatoire

```

<?xml version="1.0" encoding="UTF-8"?>
<impp:presence xmlns:impp="urn:ietf:params:xml:ns:pidf"
  xmlns:myex="http://id.mycompany.com/presence/"
  entity="pres:someone@example.com">
<impp:tuple id="tj25ds">
  <impp:status>
    <impp:basic>open</impp:basic>
  </impp:status>
  <myex:complexExtension>
    <myex:ex1 impp:mustUnderstand="1">val1</myex:ex1>
    <myex:ex2>val2</myex:ex2>
  </myex:complexExtension>
  <impp:contact priority="0.725">tel:+09012345678</impp:contact>
</impp:tuple>
<myex:mytag>Mes informations de présentité étendues</myex:mytag>
</impp:presence>

```

Ici, <myex:ex1> doit être compris et, si il n'est pas reconnu, <myex:complexExtension> DOIT être ignoré. <myex:mytag> et <myex:ex2> PEUVENT être ignorés si ils ne sont pas reconnus.

4.4 Définitions de schéma XML

Ce paragraphe donne la définition du schéma XML [XMLSchema] du format "application/pidf+xml". Elle est présentée comme une définition formelle du format "application/pidf+xml". Noter que la définition du schéma XML n'est pas destinée à être utilisée avec la validation au vol du document de présence XML.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema targetNamespace="urn:ietf:params:xml:ns:pidf"
  xmlns:tns="urn:ietf:params:xml:ns:pidf"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

<!--Cet emprunt amène l'attribut de langage XML xml:lang-->
<xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>

<xs:element name="presence" type="tns:presence"/>

<xs:complexType name="presence">
  <xs:sequence>
    <xs:element name="tuple" type="tns:tuple" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="note" type="tns:note" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="entity" type="xs:anyURI" use="required"/>
</xs:complexType>

```

```

<xs:complexType name="tuple">
  <xs:sequence>
    <xs:element name="status" type="tns:status"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="contact" type="tns:contact" minOccurs="0"/>
    <xs:element name="note" type="tns:note" minOccurs="0"
      maxOccurs="unbounded"/>
    <xs:element name="timestamp" type="xs:dateTime" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:ID" use="required"/>
</xs:complexType>

<xs:complexType name="status">
  <xs:sequence>
    <xs:element name="basic" type="tns:basic" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
      maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="basic">
  <xs:restriction base="xs:string">
    <xs:enumeration value="open"/>
    <xs:enumeration value="closed"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="contact">
  <xs:simpleContent>
    <xs:extension base="xs:anyURI">
      <xs:attribute name="priority" type="tns:qvalue"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType name="note">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:simpleType name="qvalue">
  <xs:restriction base="xs:decimal">
    <xs:pattern value="0(\.[0-9]{0,3})?"/>
    <xs:pattern value="1(\.0{0,3})?"/>
  </xs:restriction>
</xs:simpleType>

<!--Attributs globaux -->
<xs:attribute name="mustUnderstand" type="xs:boolean" default="0">
  <xs:annotation>
    <xs:documentation>
Cet attribut peut être utilisé sur tout élément dans une extension PIDF facultative pour indiquer que l'élément
correspondant doit être compris par le processeur PIDF si l'élément englobant facultatif est à traiter.
    </xs:documentation>
  </xs:annotation>
</xs:attribute>
</xs:schema>

```

5. Considérations relatives à l'IANA

Le présent mémoire invite l'IANA à :

- enregistrer un nouveau type de contenu MIME application/pidf+xml, selon la [RFC2045],
- enregistrer un nouvel URN d'espace de noms XML selon la [RFC3688].
- enregistrer un nouvel URN d'espace de noms XML pour les extensions d'état selon la [RFC3688].

On donne ci-dessous le gabarit d'enregistrement. Pour plus d'informations sur les extensions d'état, voir le paragraphe 4.2.5.

5.1 Enregistrement du type de contenu pour 'application/pidf+xml'

To: ietf-types@iana.org

Sujet : Enregistrement du type de support MIME application/pidf+xml

Nom du type de support MIME : application

Nom du sous type MIME : pidf+xml

Paramètres exigés : aucun

Paramètres facultatifs : charset. Indique le codage de caractères du XML inclus. Par défaut, UTF-8.

Considérations de codage : Utilise XML, qui peut employer des caractères de 8 bits, selon le codage de caractères utilisé. Voir la [RFC3023], paragraphe 3.2.

Considérations de sécurité : Ce type de contenu est conçu pour porter les données de présence qui peuvent être considérées comme des informations privées. Des précautions appropriées devraient être prises pour limiter la divulgation de ces informations.

Considérations d'interopérabilité : Ce type de contenu fournit un format commun pour l'échange d'informations de présence à travers différents protocoles conformes à CPP.

Spécification publiée : RFC 3863

Applications qui utilisent ce type de support : systèmes de présence et de messagerie instantanée.

Informations supplémentaires : Numéro magique :

Extensions de fichier:

Code de type de fichier Macintosh :

Personne & adresse de messagerie à contacter pour plus d'informations : Hiroyasu Sugano, sugano.h@jp.fujitsu.com

Usage prévu : utilisation limitée

Auteur/contrôleur des changements : Cette spécification est un sujet d'étude du groupe de travail IETF IMPP dont la liste de diffusion est à l'adresse <impp@iastate.edu>.

Autres informations : Ce type de support est une spécialisation de application/xml [RFC3023], et beaucoup des considérations qui y sont décrites s'appliquent aussi à application/pidf+xml.

5.2 Enregistrement du sous espace de nom d'URN pour 'urn:ietf:params:xml:ns:pidf'

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

Description : Ceci est l'URI d'espace de noms XML pour les éléments XML définis par la RFC3863 pour décrire les informations de présence CPP dans le type de contenu application/pidf+xml.

Contact du déposant :

IETF, groupe de travail IMPP, <impp@iastate.edu>

Hiroyasu Sugano, <sugano.h@jp.fujitsu.com>

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=utf-8"/>
  <title>Espace de noms pour informations de présence CPP</title>
</head>
<body>
```

```

<h1>Espace de noms pour informations de présence CPP</h1>
<h2>urn:ietf:params:xml:ns:pidf</h2>
<p>See <a
  href="[[[ftp://ftp.rfc-editor.org/in-notes/rfc3863.txt]]]">
  RFC3863</a>.</p>
</body>
</html>
END

```

5.3 Enregistrement du sous espace de nom d'URN pour 'urn:ietf:params:xml:ns:pidf:status'

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

Description : Ceci est l'URI d'espace de noms XML pour les éléments XML définis par la RFC3863 pour décrire les extensions à l'état des informations de présence CPP dans le type de contenu application/pidf+xml.

Contact du déposant :

IETF, groupe de travail IMPP, <impp@iastate.edu>
 Hiroyasu Sugano, <sugano.h@jp.fujitsu.com>

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=utf-8"/>
  <title>Espace de noms pour extensions d'état CPP</title>
</head>
<body>
  <h1>Espace de noms pour extensions d'informations de présence CPP</h1>
  <h2>urn:ietf:params:xml:ns:pidf:status</h2>
  <p>See <a
    href="[[[ftp://ftp.rfc-editor.org/in-notes/rfc3863.txt]]]">
    RFC3863</a>.</p>
</body>
</html>
END

```

6. Considérations pour la sécurité

Comme les informations de présence sont très sensibles à la confidentialité, le protocole pour les informations de présence DOIT avoir une capacité de protection du PIDF contre les menaces possibles, comme l'espionnage, la corruption, l'altération et les attaques en répétition. Ces mécanismes de sécurité doivent être capables d'une utilisation de bout en bout entre les présentités et les observateurs, même si l'observateur et la présentité emploient des protocoles de présence différents et communiquent à travers une passerelle CPP. Comme le type MIME 'application/pidf+xml' est défini pour des documents en PIDF, prévoir la sécurité pour PIDF au niveau de MIME (avec S/MIME [RFC3851]) semble approprié. Donc, PIDF devrait suivre les recommandations normatives pour l'utilisation de S/MIME (incluant les suites de chiffrement minimum) données dans la spécification principale de CPP.

Noter que l'utilisation d'horodatages dans PIDF (voir au paragraphe 4.1.7) peut apporter une protection rudimentaire contre les attaques en répétition. Si un observateur reçoit des informations de présence qui sont périmées, elles DEVRAIENT être ignorées. Un observateur peut déterminer que des informations de présence sont périmées d'un certain nombre de façons. Le plus significatif, si le nouvel horodatage dans les informations de présence est plus ancien que le plus récent horodatage dans les informations de présence reçues en dernier, elles devraient être considérées comme périmées. Les applications et protocoles devraient aussi adopter leurs propres règles pour déterminer la fréquence à laquelle les informations de présence devraient être rafraîchies. Par exemple, si des informations de présence apparaissent comme ayant plus d'une heure, elles devraient être considérées comme périmées (une notification générée pour ces informations de présence ne va prendre aussi

longtemps pour atteindre un observateur, et si une présentité n'a pas rafraîchi son état de présence dans la dernière heure, elle est probablement hors ligne).

7. Considérations d'internationalisation

Tous les processeurs qui se conforment à la présente spécification DOIVENT être capables de générer et accepter le codage UTF-8, qui est l'un des codages de caractère obligatoires pour les processeurs conformes à XML, et est aussi exigé par les politiques établies dans la [RFC2277].

D'autres codages de caractères PEUVENT être acceptés (mais il est fortement déconseillé aux processeurs conformes à CPP d'émettre autre chose que de l'UTF-8).

8. Références

8.1 Références normatives

- [RFC2045] N. Freed et N. Borenstein, "[Extensions de messagerie Internet](#) multi-objets (MIME) Partie 1 : Format des corps de message Internet", novembre 1996. (*D. S., MàJ par 2184, 2231, 5335.*)
- [RFC2046] N. Freed et N. Borenstein, "[Extensions de messagerie Internet](#) multi-objets (MIME) Partie 2 : Types de support", novembre 1996. (*D. S., MàJ par 2646, 3798, 5147, 6657.*)
- [RFC2047] K. Moore, "MIME ([Extensions de messagerie Internet](#) multi-objets) Partie trois : extensions d'en-tête de message pour texte non ASCII", novembre 1996. (*MàJ par RFC2184, RFC2231*) (*D.S.*)
- [RFC2048] N. Freed, J. Klensin et J. Postel, "Extensions multi-objets de la messagerie Internet (MIME) Partie 4 : Procédures d'enregistrement", BCP 13, novembre 1996. (*Obsolète, voir RFC 4288-4289*)
- [RFC2049] N. Freed, N. Borenstein, "[Extensions multi-objets de la messagerie Internet](#) (MIME) Partie cinq : critères de conformité et exemples", novembre 1996. (*Remplace RFC1521, RFC1522, RFC1590*) (*D.S.*)
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.
- [RFC2141] R. Moats, "[Syntaxe des URN](#)", mai 1997.
- [RFC2277] H. Alvestrand, "Politique de l'IETF en matière de [jeux de caractères et de langages](#)", BCP 18, janvier 1998.
- [RFC2396] T. Berners-Lee, R. Fielding et L. Masinter, "Identifiants de ressource uniformes (URI) : Syntaxe générique", août 1998. (*Obsolète, voir RFC3986*)
- [RFC2648] R. Moats, "Espace de nom d'URN pour les documents de l'IETF", août 1999. (*Information*)
- [RFC3023] M. Murata, S. St.Laurent et D. Kohn, "Types de support XML", janvier 2001. (*Obsolète, voir RFC7303*)
- [RFC3066] H. Alvestrand, "Étiquettes pour l'identification des langues", BCP 47, janvier 2001. (*Obsolète, voir RFC4646.*)
- [RFC3339] G. Klyne, C. Newman, "[La date et l'heure sur l'Internet](#) : horodatages", juillet 2002. (*P.S.*)
- [RFC3688] M. Mealling, "[Registre XML de l'IETF](#)", BCP 81, janvier 2004.
- [XML] Bray, T., Paoli, J., Sperberg-McQueen, C., and E. Maler, "Extensible Markup Language (XML) 1.0 (Second Edition)", W3C Recommendation, octobre 2000, < <http://www.w3.org/TR/2000/REC-xml-20001006> >
- [XML-NS] Bray, T., Hollander, D., and A. Layman "Namespaces in XML", W3C recommendation: xml-names, 14 janvier 1999, < <http://www.w3.org/TR/REC-xml-names> >
- [XMLSchema] Thompson, H., Beech, D., Maloney, M., and N. Mendelsohn, "XML Schema Part 1: Structures", W3C REC-xmlschema-1, mai 2001, < <http://www.w3.org/TR/xmlschema-1/> >.

8.2 Références pour information

- [RFC2426] F. Dawson, T. Howes, "Profil de répertoire MIME vCard", septembre 1998. (*Obsolète, voir RFC6350*) (*P.S.*)
- [RFC2778] M. Day, J. Rosenberg et H. Sugano, "[Modèle pour Presence et la messagerie instantanée](#)", février 2000.
- [RFC2779] M. Day et autres, "[Exigences des protocoles Messagerie instantanée / Presence](#)", février 2000. (*Information*)
- [RFC3282] H. Alvestrand, "[En-têtes de langage de contenu](#)", mai 2002. (*D.S.*)
- [RFC3851] B. Ramsdell, "Spécification du message d'extensions de messagerie Internet multi-objets/sécurisé (S/MIME) version 3.1", juillet 2004. (*Remplacée par RFC5751*)
- [RFC3859] J. Peterson, "[Profil commun pour les services de présence \(CPP\)](#)", août 2004. (*P.S.*)
- [RFC3860] J. Peterson, "[Profil commun pour la messagerie instantanée \(CPIM\)](#)", août 2004. (*P.S.*)

Appendice A. Définitions de type de document

On décrit la définition de type de document (DTD, *Document Type Definition*) pour le format "application/pidf+xml". Le DTD n'est présenté ici que pour l'information de ceux qui ne seraient pas familiarisés avec la définition de schéma XML.

Note : la DTD ne montre pas où les éléments d'extension peuvent être ajoutés. On trouvera cette information dans le schéma XML.

```

<!ENTITY % URL      "CDATA">
<!ENTITY % URI      "CDATA">
<!ENTITY % TUPLEID  "CDATA">
<!ENTITY % DATETIME "CDATA">
<!ENTITY % VALUETYPE "CDATA">
<!ENTITY % PRIORITY "CDATA">
<!ENTITY % NOTE     "CDATA">

<!ELEMENT presence ((tuple*),note?)>
<!ATTLIST presence
  xmlns %URI; #REQUIRED
  entity %URL; #REQUIRED
>

<!ELEMENT tuple (status,contact?,note?,timestamp?)>
<!ATTLIST tuple
  id %TUPLEID; #REQUIRED
>

<!ELEMENT status (basic?)>
<!ELEMENT basic CDATA>

<!ELEMENT contact %URL;>
<!ATTLIST contact
  priority %PRIORITY; #IMPLIED
>

<!ELEMENT note %NOTE;>

<!ELEMENT timestamp %DATETIME;>

```

Adresse des auteurs

Hiroyasu Sugano
Fujitsu Laboratories Ltd.
64, Nishiwaki
Ohkubo-cho
Akashi 674-8555
Japan
mél : sugano.h@jp.fujitsu.com

Adrian Bateman
VisionTech Limited
Colton, Staffordshire, WS15 3LD
United Kingdom
mél : bateman@acm.org

Graham Klyne
Nine by Nine
mél : GK@ninebynine.org

Shingo Fujimoto
Fujitsu Laboratories Ltd.
mél : shingo_fujimoto@jp.fujitsu.com

Wayne Carr
Intel Corporation
2111 NE 25th Avenue
Hillsboro, OR 97124
USA
mél : wayne.carr@intel.com

Jon Peterson
NeuStar, Inc.
1800 Sutter St
Suite 570
Concord, CA 94520
USA
mél : jon.peterson@neustar.biz

Déclaration complète de droits de reproduction

Copyright (c) The Internet Society (2001). Tous droits réservés.

Le présent document et ses traductions peuvent être copiés et fournis aux tiers, et les travaux dérivés qui les commentent ou les expliquent ou aident à leur mise en œuvre peuvent être préparés, copiés, publiés et distribués, en tout ou partie, sans restriction d'aucune sorte, pourvu que la déclaration de droits de reproduction ci-dessus et le présent paragraphe soient inclus dans toutes ces copies et travaux dérivés. Cependant, le présent document lui-même ne peut être modifié d'aucune façon, en particulier en retirant la notice de droits de reproduction ou les références à la Internet Society ou aux autres organisations Internet, excepté autant qu'il est nécessaire pour le besoin du développement des normes Internet, auquel cas les procédures de droits de reproduction définies dans les procédures des normes Internet doivent être suivies, ou pour les besoins de la traduction dans d'autres langues que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Internet Society, ses successeurs ou ayant droits.

Le présent document et les informations contenues sont fournis sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.