



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

J.177

(05/2003)

SÉRIE J: RÉSEAUX CÂBLÉS ET TRANSMISSION DES
SIGNAUX RADIOPHONIQUES, TÉLÉVISUELS ET
AUTRES SIGNAUX MULTIMÉDIAS

IPCablecom

**Spécification de la fourniture de service à
l'abonné par le serveur de gestion d'appels
IPCablecom**

Recommandation UIT-T J.177

RECOMMANDATIONS UIT-T DE LA SÉRIE J
RÉSEAUX CÂBLÉS ET TRANSMISSION DES SIGNAUX RADIOPHONIQUES, TÉLÉVISUELS ET AUTRES
SIGNAUX MULTIMÉDIAS

Recommandations générales	J.1–J.9
Spécifications générales des transmissions radiophoniques analogiques	J.10–J.19
Caractéristiques de fonctionnement des circuits radiophoniques analogiques	J.20–J.29
Équipements et lignes utilisés pour les circuits radiophoniques analogiques	J.30–J.39
Codeurs numériques pour les signaux radiophoniques analogiques	J.40–J.49
Transmission numérique de signaux radiophoniques	J.50–J.59
Circuits de transmission télévisuelle analogique	J.60–J.69
Transmission télévisuelle analogique sur lignes métalliques et interconnexion avec les faisceaux hertziens	J.70–J.79
Transmission numérique des signaux de télévision	J.80–J.89
Services numériques auxiliaires propres aux transmissions télévisuelles	J.90–J.99
Prescriptions et méthodes opérationnelles de transmission télévisuelle	J.100–J.109
Services interactifs pour la distribution de télévision numérique	J.110–J.129
Transport des signaux MPEG-2 sur les réseaux par paquets	J.130–J.139
Mesure de la qualité de service	J.140–J.149
Distribution de la télévision numérique sur les réseaux locaux d'abonnés	J.150–J.159
IPCablecom	J.160–J.179
Divers	J.180–J.199
Application à la télévision numérique interactive	J.200–J.209

Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.

Recommandation UIT-T J.177

Spécification de la fourniture de service à l'abonné par le serveur de gestion d'appels IPCablecom

Résumé

La fourniture de service IPCablecom 1.0 peut être considérée comme deux opérations distinctes: celle d'adaptateur de terminal multimédia et celle de service à l'abonné par le serveur de gestion d'appels. La présente Recommandation définit l'interface utilisée entre les serveurs de gestion d'appels et de mise en service pour l'échange d'informations concernant la fourniture de service.

Source

La Recommandation J.177 de l'UIT-T a été approuvée par la Commission d'études 9 (2001-2004) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8 le 29 mai 2003.

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée mondiale de normalisation des télécommunications (AMNT), qui se réunit tous les quatre ans, détermine les thèmes d'étude à traiter par les Commissions d'études de l'UIT-T, lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution 1 de l'AMNT.

Dans certains secteurs des technologies de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

Le respect de cette Recommandation se fait à titre volontaire. Cependant, il se peut que la Recommandation contienne certaines dispositions obligatoires (pour assurer, par exemple, l'interopérabilité et l'applicabilité) et considère que la Recommandation est respectée lorsque toutes ces dispositions sont observées. Le futur d'obligation et les autres moyens d'expression de l'obligation comme le verbe "devoir" ainsi que leurs formes négatives servent à énoncer des prescriptions. L'utilisation de ces formes ne signifie pas qu'il est obligatoire de respecter la Recommandation.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

© UIT 2003

Tous droits réservés. Aucune partie de cette publication ne peut être reproduite, par quelque procédé que ce soit, sans l'accord écrit préalable de l'UIT.

TABLE DES MATIÈRES

		Page
1	Domaine d'application	1
	1.1 Objet de la présente Recommandation	1
	1.2 Domaine d'application de la présente Recommandation.....	1
	1.3 Prescriptions	1
2	Références.....	2
	2.1 Références normatives.....	2
	2.2 Obtention de documents de référence	3
3	Termes et définitions	3
4	Abréviations.....	5
5	Généralités	6
	5.1 Objectifs de service	6
	5.2 Architecture de référence IPCablecom.....	6
	5.3 Composants et interfaces.....	7
	5.4 Composants	8
	5.5 Descriptions d'interfaces.....	8
6	Hypothèses.....	9
7	Fourniture de service à l'abonné	9
	7.1 Relevé (facturation) d'abonné.....	9
	7.2 Montage et configuration des équipements.....	9
	7.3 Données statiques et dynamiques de fourniture de service à l'abonné.....	10
8	Prescriptions	10
	8.1 Prescriptions générales	10
	8.2 Prescriptions de transport	11
9	Modèle de données	12
	9.1 Aperçu général.....	12
	9.2 Les relations sont codées au moyen de l'entité PcspRelation.Codage XML..	14
10	Messagerie	15
	10.1 Aperçu général.....	15
	10.2 Prescriptions concernant le rôle d'émission de messages des serveurs CMS et PS	16
	10.3 Spécification WSDL.....	17
11	Sécurité	17
	Annexe A – Schéma XML de PCSP.....	18
	Annexe B – Spécification WSDL pour messagerie PCSP.....	38
	Appendice I – Exemples de codage d'entités	43
	I.1 Exemple d'objet PcspService.....	43
	I.2 Exemple d'objet PcspEndpoint.....	45

	Page
I.3 Exemple d'objet PfspMta	45
I.4 Exemple d'objet PfspCms	46
I.5 Exemple de relation PfspRelation.....	46
Appendice II – Exemple d'extension d'objet.....	46
II.1 Exemple d'objet étendu PfspService Object	46
II.2 Schéma d'extension	48
Appendice III – Evaluation du codage des données	48
III.1 XML	48
III.2 ASN.1/Règles de codage fondamentales (BER).....	49
III.3 ASCII propriétaire	49
III.4 Protocole de description de session (SDP).....	49
III.5 RADIUS	49
III.6 SQL.....	49
III.7 Récapitulation des options.....	49
III.8 Recommandation: XML.....	50
Appendice IV – Evaluation du protocole de transport.....	51
IV.1 Protocole TFTP offrant une sécurité Internet.....	51
IV.2 Enregistrement RADIUS multiple par lots en demande simple via messages d'événement	51
IV.3 Diameter	51
IV.4 Systèmes d'objets répartis.....	52
IV.5 HTTP	53
IV.6 Récapitulation des options.....	53
IV.7 Recommandation: HTTP 1.1	54
Appendice V – Bibliographie	54

Recommandation UIT-T J.177

Spécification de la fourniture de service à l'abonné par le serveur de gestion d'appels IPCablecom

1 Domaine d'application

1.1 Objet de la présente Recommandation

L'interface utilise un modèle de service Web. Spécifiée dans le langage WSDL 1.1 [9] l'interface achemine des objets codés XML dans des messages codés suivant le protocole SOAP 1.1 [10] (protocole simple d'accès d'objet) par un protocole de transport HTTP 1.1. Cette interface est sécurisée via IPSec.

Le modèle de données acheminé sur cette interface est spécifiquement conçu pour être extensible, permettant ainsi l'incorporation de fonctionnalités IPCablecom non définies à ce jour et d'extensions propres au vendeur.

1.2 Domaine d'application de la présente Recommandation

Le domaine d'application de la présente Recommandation est limité par la mise en service d'un serveur de gestion d'appels IPCablecom 1.0 par un unique fournisseur de services. En outre:

- l'interface de mise en service du serveur de gestion d'appels est limitée par l'échange de données d'activation du service entre le serveur CMS et le serveur de protocole. L'interface entre le serveur PS et le système d'assistance à l'exploitation ne relève pas du présent domaine d'application;
- la gestion d'éléments de serveur CMS et la fourniture d'éléments de réseau (plans de numérotation, etc.) ne relèvent pas du domaine d'application;
- la création d'enregistrements et/ou la facturation d'abonné est considérée comme faisant partie de l'application administrative d'assistance à l'exploitation et ne relève pas actuellement du domaine d'application;
- la présente Recommandation constitue un cadre de référence pour la fourniture de services à l'abonné par le serveur de gestion d'appels IPCablecom. Les implémentations doivent observer les exigences appropriées de la présente Recommandation, mais ne prennent pas toujours nécessairement en charge la totalité des éléments, des paramètres et des fonctionnalités indiqués dans les annexes.

1.3 Prescriptions

Si la présente Recommandation est implémentée, le mot clé "DOIT" est interprété comme indiquant un aspect "OBLIGATOIRE" de la présente Recommandation. Les mots clés indiquant un certain niveau d'importance de telle ou telle prescription utilisée dans la présente Recommandation sont résumés comme suit:

"DOIT"	Ce mot ainsi que l'adjectif "REQUIS" indiquent que l'article est une prescription absolue dans la présente Recommandation.
"NE DOIT PAS"	Cette expression indique que l'article est une interdiction absolue de la présente Recommandation.
"IL CONVIENT DE"	Cette expression ainsi que l'adjectif "RECOMMANDÉ" indiquent qu'il peut, dans des circonstances particulières, exister des raisons valables pour ignorer cet article, mais qu'il convient, avant de faire

ce choix, de prendre en considération la totalité des incidences et d'étudier soigneusement le cas.

"IL NE CONVIENT PAS DE" Cette expression indique qu'il peut, dans des circonstances particulières, exister des raisons valables pour que le comportement indiqué soit acceptable ou même utile, mais qu'il convient avant de faire ce choix de prendre en considération la totalité des incidences et d'étudier soigneusement le cas.

"PEUT" Ce mot, ainsi que l'adjectif "FACULTATIF" indiquent que cet article est effectivement facultatif. Un fournisseur peut choisir d'inclure l'article par exemple parce qu'il est requis sur un marché particulier ou parce qu'il améliore le produit, alors qu'un autre fournisseur peut choisir d'omettre ce même article.

2 Références

2.1 Références normatives

La présente Recommandation se réfère à certaines dispositions des Recommandations UIT-T et textes suivants qui de ce fait en font partie intégrante. Les versions indiquées étaient en vigueur au moment de la publication de la présente Recommandation. Toute Recommandation ou tout texte étant sujet à révision; les utilisateurs de la présente Recommandation sont invités à se reporter, si possible, aux versions les plus récentes des références normatives suivantes. La liste des Recommandations de l'UIT-T en vigueur est régulièrement publiée. La référence à un document figurant dans la présente Recommandation ne donne pas à ce document en tant que tel le statut d'une Recommandation.

- [1] Recommandation UIT-T J.160 (2002), *Cadre architectural pour la fourniture de services à temps critique sur les réseaux de télévision par câble utilisant des câblo-modems.*
- [2] Recommandation UIT-T J.167 (2001), *Prescriptions pour les adaptateurs de support pour la fourniture de services en temps réel sur les réseaux de télévision par câble au moyen de câblo-modems.*
- [3] Recommandation UIT-T J.170 (2002), *Spécifications de la sécurité sur IPCablecom.*
- [4] ANSI/SCTE 24-13 (2001), *IPCablecom Part 13: Electronic Surveillance Standard.*
- [5] ISO 639-2:1998, *Codes pour la représentation des noms de langue – Partie 2: Code alpha-3.*
- [6] IETF RFC 1123 (1989), *Requirements for Internet Hosts – Application and Support.*
- [7] Recommandation UIT-T J.161 (2001), *Caractéristiques des codecs audio destinés au service audio bidirectionnel sur les réseaux de télévision par câble utilisant des câblo-modems.*
- [8] Recommandation UIT-T J.162 (2001), *Protocole réseau de signalisation d'appel pour la fourniture de services à temps critique sur les réseaux de télévision par câble utilisant des câblo-modems.*
- [9] Langage WSDL de description de services Web. <http://www.w3.org/TR/wsdl>.

[10] Protocole simple d'accès d'objet. <http://www.w3.org/TR/SOAP>.

[11] Protocole XML. <http://www.w3.org/2000/xp>.

2.2 Obtention de documents de référence

Rapport du Groupe de travail d'ingénierie Internet:

- Secrétariat du Groupe de travail d'ingénierie Internet (IETF, *Internet Engineering task force*) c/o Corporation for National Research Initiatives, 1895 Preston White Drive, Suite 100, Reston, VA 20191-5434, téléphone 703-620-8990; fax: 703-620-9071, Internet: www.ietf.org/.

Normes ISO:

- Organisation internationale de normalisation (ISO, *International Organization for Standardization*), 1 rue de Varembé, Case postale 56, CH-1211 Genève 20, Suisse, téléphone: +41-22-749-01-11; fax: +41-22-733-34-30, Internet: www.iso.ch/.

Documents Telcordia:

- Telcordia Technologies, 445 South Street, Morristown, NJ 07960-6438, téléphone 973-829-2000, Internet: www.telcordia.com/.

3 Termes et définitions

La présente Recommandation définit les termes suivants:

3.1 actif: un flux de service est déclaré actif lorsqu'il est habilité à acheminer des paquets de données. Avant d'être actif, un flux de service doit au préalable être autorisé.

3.2 système de terminaison de câble-modem: dispositif situé à l'extrémité d'un câble qui implémente le protocole DOCSIS RFI MAC et assure le raccordement aux câble-modems sur un réseau HFC.

3.3 serveur de gestion d'appel: dispositif de commande des connexions audio. Désigné également par l'expression agent d'appel suivant la terminologie MGCP/SGCP. Exemple de serveur d'application.

3.4 identificateur de circuit: identification exclusive d'un circuit ISUP DSO sur une passerelle média. Il s'agit d'une combinaison du code de point passerelle SS7 du circuit et du code d'identification de circuit (CIC). Le code du point de destination (DPC SS7) est associé à la passerelle de signalisation dont le domaine comprend le circuit en question.

3.5 qualité de service dynamique: attribuée instantanément à chaque communication en fonction de la qualité de service demandée.

3.6 point d'extrémité: terminal, passerelle ou unité de conférence multipoint.

3.7 passerelle: dispositifs servant de ponts entre l'environnement IPCablecom de communication locale IP et le RTPC; par exemple, la passerelle média qui comporte les interfaces des circuits supports avec le RTPC et transcode le flux média et la passerelle de signalisation qui émet et reçoit une signalisation de réseau à commutation de circuit au niveau de la frontière du réseau IPCablecom.

3.8 en-tête: information de commande de protocole située au début d'une unité de données de protocole.

3.9 groupe de travail d'ingénierie Internet: organisme responsable entre autres, d'élaborer des normes utilisées sur Internet. Voir indications détaillées sur le site <http://www.ietf.org/>.

3.10 échange de clés Internet: mécanisme de gestion des clés utilisées afin de négocier et de calculer la valeur des clés utilisées pour déterminer des adresses sources en sécurité IP.

- 3.11 protocole Internet:** protocole de la couche Réseau Internet.
- 3.12 sécurité du protocole Internet:** ensemble de normes Internet destinées à protéger les paquets IP par des dispositifs de cryptage et d'authentification.
- 3.13 charge utile de sécurité d'encapsulation IPSec:** protocole assurant conjointement le cryptage des paquets IP et l'intégrité des messages à titre facultatif, ne couvrant cependant pas l'en-tête de paquet IP.
- 3.14 clé:** valeur mathématique introduite dans l'algorithme cryptographique choisi.
- 3.15 gestion de clés:** processus de distribution de clés symétriques partagées nécessaires à l'application d'un protocole de sécurité.
- 3.16 portabilité de numéro local:** permet à un usager de conserver le même numéro lorsqu'il passe d'un fournisseur de service local à un autre.
- 3.17 passerelle média:** fournit les interfaces circuit support au RTPC et transcode le flux média.
- 3.18 protocole de commande de passerelle média:** enchaînement de protocoles au SGCP. Voir publication IETF RFC 2705.
- 3.19 contrôleur de passerelle média:** fonction de contrôle général de la passerelle RTPC. Reçoit, contrôle et échange des informations de signalisation d'appel en IPCablecom et le RTPC.
- 3.20 adaptateur de terminal multimédia:** contient l'interface vers un dispositif vocal physique, une interface de réseau, des codecs, et toutes les fonctions de signalisation et d'encapsulation requises pour le transport de signaux téléphoniques utilisant le protocole Internet, la signalisation de fonctionnalités de classe et la signalisation de qualité de service.
- 3.21 systèmes support d'exploitation:** logiciel administratif utilisé pour les besoins de gestion de la configuration, de la qualité de fonctionnement, des défaillances, de la comptabilité et de la sécurité.
- 3.22 clé prépartagée:** clé secrète partagée transmise aux deux extrémités dans un flux de communication au moyen d'un mécanisme manuel non spécifié ou d'un mécanisme hors bande.
- 3.23 clé publique:** clé utilisée en cryptographie à clé publique qui appartient à une seule entité et est distribuée publiquement. Les autres entités utilisent cette clé pour chiffrer les données envoyées au détenteur de la clé.
- 3.24 modulation par impulsion et codage:** algorithme couramment utilisé afin de numériser un signal analogique (par exemple la voix humaine) en flux binaire numérique au moyen de simples techniques de conversion analogique numérique.
- 3.25 enregistrement, admission et statut:** le canal RAS est un canal non fiable servant à acheminer les messages RAS et les modifications de largeur de bande entre deux entités H.323.
- 3.26 demande de commentaires:** documents techniques d'orientation approuvés par l'IETF, disponible sur le site web suivant: <http://www.ietf.cnri.reston.va.us/rfc.html>.
- 3.27 point de transfert de signal:** nœud à l'intérieur d'un réseau SS7 qui achemine les messages de signalisation en fonction de leur adresse de destination. Il s'agit essentiellement d'un commutateur de paquets pour la signalisation SS7. Il peut en outre fournir des services supplémentaires d'acheminement, par exemple la traduction de titre global.
- 3.28 type-longueur-valeur:** n-uplet dans un fichier de configuration DOCSIS.
- 3.29 protocole datagramme d'utilisateur:** protocole de connexion fondé sur le protocole Internet (IP).

4 Abréviations

La présente Recommandation définit les abréviations suivantes:

AAA	authentification, autorisation et comptabilité (<i>authentication, authorization and accounting</i>)
BPP	fourniture de service de base (<i>basic POTS provisioning</i>)
CFP	fourniture de fonctionnalité d'appel (<i>call feature provisioning</i>)
CID	identificateur de circuit (<i>circuit ID</i>)
CM	câblo-modem DOCSIS
CMS	syntaxe de message cryptographique (<i>cryptographic message syntax</i>)
CMS	serveur de gestion d'appel (<i>call management server</i>)
CMTS	système de terminaison de câblo-modem (<i>cable modem termination system</i>)
Codec	codeur-décodeur
CSR	représentant du service client (<i>customer service representative</i>)
DOCSIS TM	spécifications d'interface de service de transmission de données par câble (<i>data-over-cable service interface specifications</i>)
DQoS	qualité de service dynamique (<i>dynamic quality of service</i>)
ESP	charge utile de sécurité d'encapsulation IPsec (<i>IPsec encapsulating security payload</i>)
FQDN	nom de domaine complet (<i>fully qualified domain name</i>). Voir IETF RFC 821 pour de plus amples détails.
HTTP	protocole de transfert hypertexte (<i>hypertext transfer protocol</i>). Voir IETF RFC 1945 et RFC 2068.
IC	porteuse intercirconscriptions (<i>inter-exchange carrier</i>)
IETF	groupe de travail d'ingénierie Internet (<i>Internet Engineering task force</i>)
IKE	échange de clés Internet (<i>Internet key exchange</i>)
IP	protocole Internet (<i>Internet protocol</i>)
IPsec	sécurité du protocole Internet (<i>Internet protocol security</i>)
LNP	portabilité de numéro local (<i>local number portability</i>)
MC	contrôleur multipoint (<i>multipoint controller</i>)
MG	passerelle média (<i>media gateway</i>)
MGC	contrôleur de passerelle média (<i>media gateway controller</i>)
MGCP	protocole de commande de passerelle média (<i>media gateway control protocol</i>)
MIB	base d'informations de gestion (<i>management information base</i>)
MIC	modulation par impulsion et codage
MTA	adaptateur de terminal multimédia (<i>multimedia terminal adapter</i>)
NCS	signalisation d'appel de réseau (<i>network call signalling</i>)
OSS	systèmes support d'exploitation (<i>operations support system</i>)

PCSP	fourniture de service à l'abonné par le serveur de gestion d'appels IPCablecom (<i>IPCablecom CMS subscriber provisioning</i>)
PS	serveur de mise en service (<i>provisioning server</i>)
QCIF	format quart du CIF (<i>quarter common intermediate format</i>)
RAS	enregistrement, admission et statut (<i>registration, admissions and status</i>)
RFC	demande de commentaires (<i>request for comments</i>)
RTPC	réseau téléphonique public commuté
SDP	protocole de description de session (<i>session description protocol</i>)
SNMP	protocole simple de gestion de réseau (<i>simple network management protocol</i>)
SOAP	protocole simple d'accès aux objets (<i>simple object access protocol</i>)
STP	point de transfert de signal (<i>signalling transfer point</i>)
TD	temporisation de déconnexion
TFTP	protocole trivial de transfert de fichier (<i>trivial file transfer protocol</i>)
TLV	type-longueur-valeur
UDP	protocole datagramme d'utilisateur (<i>user datagram protocol</i>)
UIT	Union internationale des télécommunications
UIT-T	Union internationale des télécommunications – Bureau de normalisation des télécommunications
VoIP	téléphonie utilisant le protocole Internet (<i>voice over Internet protocol</i>)

5 Généralités

5.1 Objectifs de service

5.2 Architecture de référence IPCablecom

La Figure 1 représente l'architecture de référence du réseau IPCablecom 1.0. Des indications plus détaillées concernant cette architecture de référence figurent dans le document d'architecture IPCablecom [1].

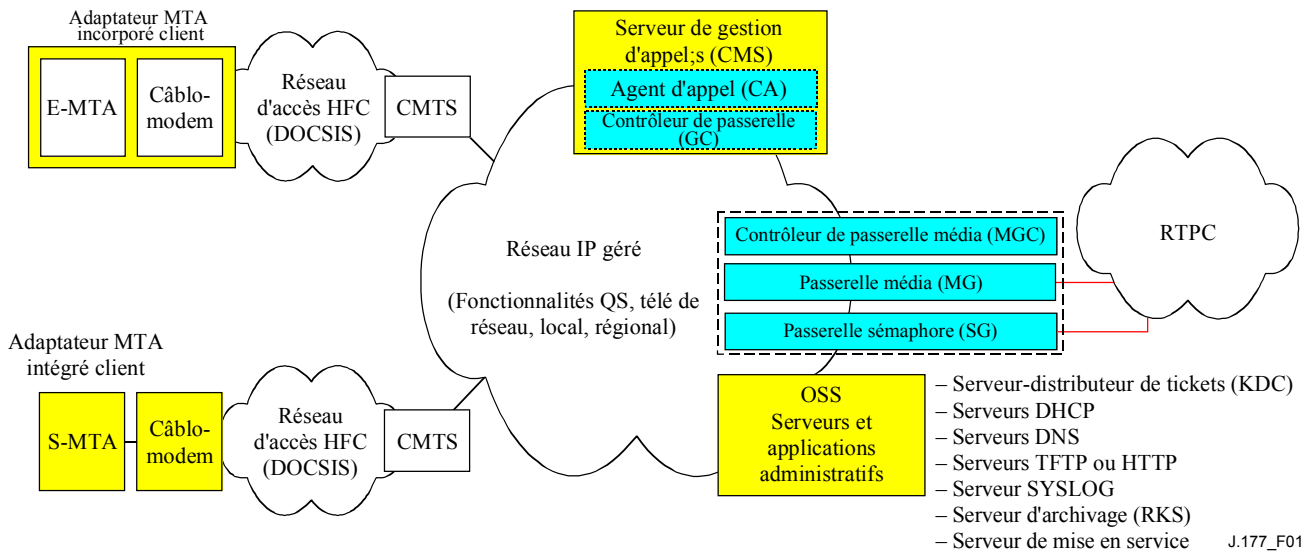


Figure 1/J.177– Modèle de référence de composant de réseau IPCablecom 1.0 (partiel)

5.3 Composants et interfaces

La mise en service est définie comme l'ensemble des activités nécessaires à la fourniture d'un service déterminé à un abonné. La fourniture du service IPCablecom peut être considérée comme deux opérations distinctes: fourniture de l'adaptateur de terminal média et fourniture de service à l'abonné par le serveur de gestion d'appel. La Figure 2 représente les interfaces de fourniture de service assurées par le serveur de mise en service par rapport aux différents éléments IPCablecom. Les interfaces qui ne sont pas explicitement désignées ne sont pas définies et ne relèvent pas de IPCablecom.

La présente Recommandation vise à définir les spécifications concernant l'interface de mise en service entre le serveur de gestion d'appels et le serveur de mise en service (Pkt-prov-p1).

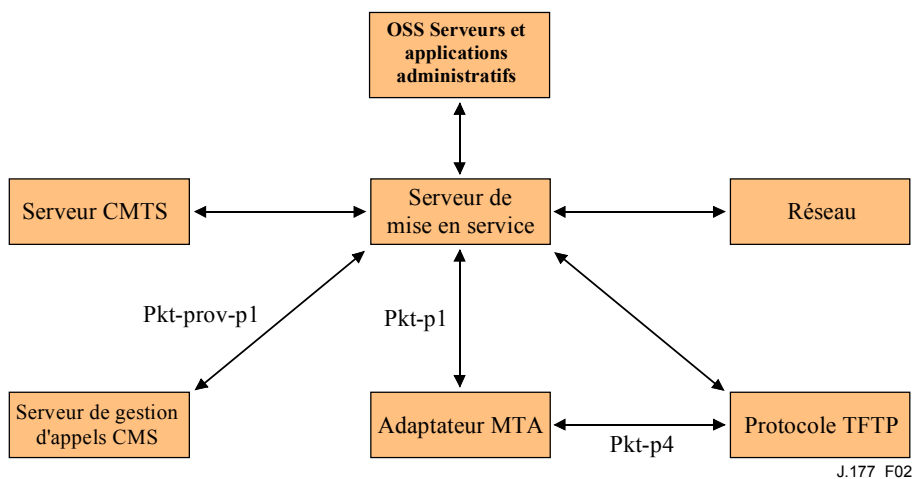


Figure 2/J.177 – Interfaces des composants de mise en service

5.4 Composants

5.4.1 Serveurs et applications administratifs (systèmes de gestion commerciaux de services à la clientèle du fournisseur de services)

Il s'agit des serveurs et des applications administratifs utilisés par un fournisseur de services pour gérer les usagers et les différents composants qui font l'objet de son activité commerciale. Ces systèmes dotent le processus de mise en service IPCablecom de commande d'activation des services à l'attention des abonnés. Ces systèmes peuvent également recevoir des données de comptabilité ou d'utilisation permettant de créer des événements de facturation des abonnés.

5.4.2 Serveur de mise en service

Ce système constitue l'interface entre les serveurs et les applications administratifs du prestataire et les éléments IPCablecom. La mise en place de ce système ou de ces relations avec d'autres serveurs et applications administratifs éventuellement utilisés par un fournisseur de service ne relève pas de IPCablecom.

Le serveur de mise en service (ou de configuration) est défini dans [2] comme étant constitué d'une application de mise en service contenant une logique de mise en service ainsi qu'une entité SNMP de mise en service, qui permet d'accéder aux composants actifs. Sera mentionné ci-après le serveur de mise en service sans distinguer entre ses deux entités constitutives.

5.4.3 Serveur CMS

La description du composant serveur de gestion d'appels figure dans [1]. Ce composant fournit des services de commande d'appel et de signalisation aux composants MTA et CMTS du réseau IPCablecom.

5.4.4 Adaptateur MTA

Un adaptateur de terminal média est un dispositif client IPCablecom qui contient une interface côté abonné avec les équipements des locaux clients (CPE, *customer premises equipment*) (par exemple, téléphone) et une interface de signalisation côté réseau avec les éléments de commande d'appel du réseau. La description de ce composant figure dans [1].

5.4.5 Protocole TFTP

Ce service de fichier de configuration est à la base de la configuration de la plupart des dispositifs d'un réseau IPCablecom. Il peut s'agir d'un service TFTP autonome qui fournit des fichiers définis de façon statique à différents dispositifs ou encore d'un service dynamique qui crée des configurations à la volée à partir d'autres sources de données.

5.4.6 Système CMTS

Le système de terminaison de câblo-modem assure la commande des modems et l'acheminement de paquets à l'entrée et à la sortie d'un réseau HF. Le système CMTS assure également la connectivité des données avec des réseaux IP étendus. La description de ce composant figure dans [1].

5.4.7 Réseau

Ensemble des éléments de réseaux pouvant être nécessaires afin d'assurer la signalisation, la gestion et le transport de paquets IPCablecom à destination et en provenance de réseaux fournisseurs de service.

5.5 Descriptions d'interfaces

5.5.1 Pkt-p1

La définition de cette interface figure dans [2].

5.5.2 Pkt-p4

La définition de cette interface figure dans [2].

5.5.3 Pkt-prov-p1

La définition de cette interface figure dans la présente Recommandation.

6 Hypothèses

- Le système administratif OSS est chargé de coordonner les mises à jour de point terminal avec les entités réseaux concernées (adaptateurs MTA, systèmes CMTS, etc.) et le serveur CMS.
- Le serveur de gestion d'appels CMS ne doit pas jouer un rôle de gestion, ni spécifier des communications de protocole SNMP vers un adaptateur MTA au cours d'une mise en service de serveur de gestion d'appels.
- Le serveur de gestion d'appels et le serveur de mise en service résident dans le même domaine de mise en service sécurisé. Les informations concernant la sécurité seront définies dans la Recommandation de sécurité IPCablecom [3].

7 Fourniture de service à l'abonné

La fourniture de service à l'abonné comprend:

- la prise en charge des relevés/ou de la facturation des abonnés;
- le démarrage/la configuration des équipements.

7.1 Relevé (facturation) d'abonné

L'établissement d'un relevé d'abonné contenant les informations requises pour fournir le service facturé et recueillir le paiement d'un abonné. La création de relevé/facturation des abonnés est considérée comme faisant partie de l'application administrative OSS et ne relève pas actuellement de IPCablecom.

7.2 Montage et configuration des équipements

Cette activité peut comporter l'installation matérielle et/ou le raccordement de l'équipement ainsi que des logiciels et/ou des mises à jour de base de données nécessaires pour assurer effectivement la fourniture du service à l'utilisateur. Le montage de l'équipement affecte deux composants importants dans un environnement IPCablecom.

- Equipement des locaux client constitué pour IPCablecom par l'adaptateur de terminal de média. Les conditions de mise en service de l'adaptateur de terminal de média sont définies dans [2] et ne seront pas examinées dans la présente Recommandation.
- Le serveur de gestion d'appels. La mise en service du serveur de gestion d'appels proprement dit peut être scindée en deux points essentiels: celle du service téléphonique ordinaire de base et celle des fonctionnalités d'appel.

7.2.1 Fourniture du service ordinaire (BPP, *basic POTS provisioning*) au serveur CMS

L'opération BPP fournit au serveur CMS le jeu de données minimales nécessaires pour acheminer le service ordinaire à l'intérieur du réseau IPCablecom. Ce jeu de données minimales est constitué d'un numéro de téléphone mappé avec le nom de domaine complet de l'adaptateur de terminal de média qui lui est associé et l'identificateur de point terminal de signalisation d'appel par le réseau. Ces données serviront à établir des tables de traduction de configuration permettant au service CMS d'acheminer les appels vers le dispositif/port approprié, affectés à un numéro de téléphone

spécifique. Un processus BPP est indispensable pour chaque abonné avant qu'il puisse recevoir un appel quelconque dans un réseau IPCablecom.

7.2.2 Fourniture de fonctionnalité d'appel au serveur CMS (CFP)

Outre l'opération BPP, l'opération de fourniture de fonctionnalité d'appel CFP vise à fournir des fonctionnalités d'appel à un abonné. Cette opération est néanmoins plus complexe, dans la mesure où les paramètres transmis peuvent varier d'une fonctionnalité à l'autre et peuvent également dépendre des implémentations propres au fournisseur. Une analyse plus détaillée de l'opération CFP peut être envisagée dans les versions futures du présent document.

7.3 Données statiques et dynamiques de fourniture de service à l'abonné

Les données requises par le serveur CMS à cet effet sont classées dans deux catégories:

- 1) données d'état du service à caractère statique, facturées et attribuées de façon permanente. Ces données ne changent pas d'un appel à l'autre. Par exemple, les réglages de qualité de service, les états d'abonnement/non-abonnement aux fonctionnalités d'appel, les informations d'identification du demandeur, etc.;
- 2) données d'état du service, à caractère dynamique, non facturées et semi-permanentes. Il s'agit dans nombre de cas d'informations pouvant être modifiées par l'abonné, soit à un point d'extrémité via un code *XX, soit par une interface web, à l'intérieur du serveur de gestion d'appels. On peut citer à titre d'exemple les paramètres réglables par l'utilisateur d'une fonctionnalité d'appel, telle que le renvoi automatique en cas d'occupation (CFBL, *call forward busy line*). Le numéro de renvoi utilisé par cette fonctionnalité est une donnée d'état de service de type dynamique et non facturée. L'abonnement/non-abonnement au renvoi d'appel sur occupation est une donnée à caractère statique tenue à jour par le serveur de mise en service.

Dans le contexte des serveurs CMS/PS IPCablecom, les serveurs PS contiennent toutes les données respectivement statiques et dynamiques de mise en service.

8 Prescriptions

8.1 Prescriptions générales

- L'interface DOIT ne faire aucune hypothèse quant aux techniques d'implémentation des serveurs PS et CMS.

Les fournisseurs faisant appel à des partenariats multiples proposeront certainement des implémentations de serveurs CMS et PS utilisant des matériels et des logiciels variés, ainsi que différentes plates-formes de langage d'élaboration de programmes. Une plate-forme ainsi qu'une interface neutres de langage sont indispensables.

- L'interface doit prendre en charge la fourniture de l'accès au service ordinaire.
- Le modèle de données de l'interface DOIT contenir la quantité minimale d'informations requises pour prendre en charge le service ordinaire.
- L'interface DOIT prendre en charge la fourniture de fonctionnalité d'appel.

Le modèle de données de l'interface DOIT prendre en charge l'abonnement à toute fonctionnalité d'appel IPCablecom définie dans [12] et [13].

- Le modèle de données de l'interface DOIT être extensible.

Actuellement l'interface porte essentiellement sur les données téléphoniques. Toutefois, dans la mesure des possibilités, il conviendrait de prévoir une interface extensible aux futurs services multimédias IPCablecom. Il y a intérêt à disposer d'un modèle de données unique et extensible pour la fourniture de services et à assurer son transport de façon à

prendre en charge la totalité des fonctionnalités IPCablecom, dont certaines ne sont pas encore définies.

- L'interface ne DOIT affecter aucune opération en cours de l'adaptateur MTA.

Les données propres au point d'extrémité peuvent être ajoutées, supprimées ou modifiées sur l'adaptateur MTA sans affecter d'autres points d'extrémité de l'adaptateur MTA ou d'autres sessions en cours. Il faut soigneusement documenter les scénarios de mise en service de point d'extrémité par le serveur de gestion d'appels, qui entraîneraient la mise hors service d'un point d'extrémité/adaptateur de terminaison média.

- L'interface DOIT pouvoir prendre en charge les protocoles actuels (NCS) et futurs de signalisation.

8.2 Prescriptions de transport

- Le transport ne DOIT impliquer aucune hypothèse concernant l'infrastructure physique de mise en réseau entre un service de mise en service et un serveur de gestion d'appels.

Il est prévu que plusieurs fournisseurs de services opéreront en interfonctionnement sur un réseau géré unique. Par conséquent, plusieurs entreprises communiqueront, éventuellement au moyen d'implémentations de serveurs CMS et PS provenant de différents fournisseurs et utilisant différentes infrastructures de réseaux (passerelles, serveurs mandataires, etc.). Le protocole de transport CMS/PS doit faciliter la capacité de pénétration d'une infrastructure de réseau arbitraire.

- Le transport DOIT prendre en charge le transfert unidirectionnel d'objets uniques de modèles de données, depuis le serveur de mise en service jusqu'au serveur de gestion d'appels.
- Le transport DOIT prendre en charge des transactions efficaces d'objets multiples de modèles de données, depuis le serveur de mise en service jusqu'au serveur de gestion d'appels.
- Le transport PEUT prendre en charge le transfert unidirectionnel d'objets uniques de modèles de données, depuis le serveur de gestion d'appels jusqu'au serveur de mise en service.
- Le transport PEUT prendre en charge une émission efficace en continu de plusieurs objets de modèles de données depuis le serveur de gestion d'appels jusqu'au serveur de mise en service.
- Le transport DOIT comporter une sémantique propre à prendre en charge des objets de modèles de données nouveaux, mis à jour et supprimés.
- Le transport DOIT prendre en charge les demandes d'informations formulées entre le serveur PS et le serveur CMS.
- Le transport DOIT traiter les situations telles que l'occupation du serveur CMS, les erreurs, etc.
- Le transport DOIT assurer l'accusé de réception positif/négatif des opérations reçues.
- Le transport DOIT implémenter une sémantique de messages de type au moins une fois. L'expéditeur ne DOIT pas supprimer sa demande tant que le destinataire n'en a pas accusé réception (les accusés de réception ne sont pas reconnus). Le transport doit pouvoir déceler les données corrompues en cours d'acheminement, etc. et en informer l'expéditeur.
- Le transport DOIT fournir un accusé de réception positif/négatif des opérations traitées.
- Le serveur de mise en service DOIT pouvoir amorcer un transfert d'objets de modèles de données.
- Le transport DOIT être sécurisé.

9 Modèle de données

Le présent paragraphe présente une description de haut niveau du modèle de données PCSP et de son codage XML. La définition normative obligatoire du modèle de données et de son codage figure dans le schéma XML de protocole PCSP à l'Annexe A.

9.1 Aperçu général

Le modèle de données applicable à la fourniture de services par serveur CMS IPCablecom est représenté à la Figure 3. Il comprend deux catégories d'entités:

- objets;
- relations entre objets.

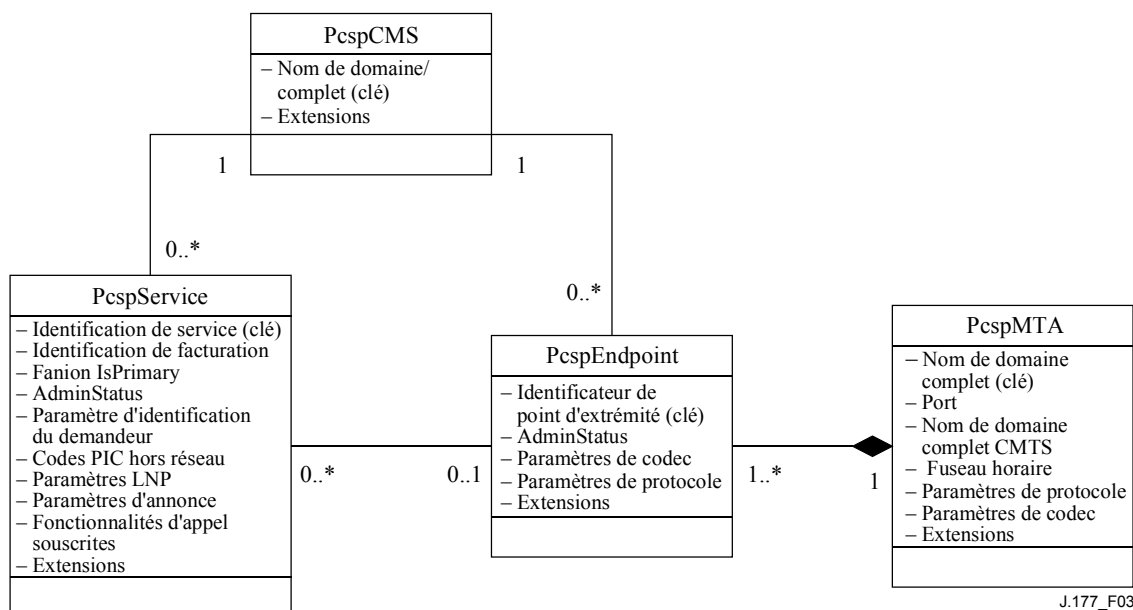


Figure 3/J.177 – Modèle de données de fourniture de services par serveur CMS¹

Les entités suivantes DOIVENT être prises en charge:

- l'objet PcpService est l'entité à laquelle un nœud usagé IPCablecom 1.0 s'abonne. Il est constitué par un numéro de téléphone et par toutes les fonctionnalités connexes (fonctionnalités d'appel, etc.);
- un objet PcpMTA est constitué par un adaptateur de terminal de média, qui regroupe plusieurs points d'extrémité contenus physiquement dans l'adaptateur MTA;
- l'objet PcpEndpoint est constitué par un point d'extrémité matériel sur une passerelle/ un MTA;
- un objet PcpCMS maintient les associations entre points d'extrémité/serveurs CMS et services/serveurs CMS;
- les entités PcpRelations sont constituées par des associations entre objets. Elles sont représentées à la Figure 3 par des connexions entre objets.

¹ Un projet contient une ou plusieurs activités, les différentes chaînes notamment "1..*" associées au projet et aux associations d'activités définissent la *multiplicité* de la classe d'activité à l'intérieur de la relation. Des indications détaillées complémentaires figurent dans la référence à un langage de modélisation harmonisé (UML, *unified modelling language*).

Les objets PcpsService et PcpsEndpoint sont distincts, afin de pouvoir prendre en charge plusieurs services (numéros de téléphone) par point d'extrémité. Les objets distincts PcpsMta et PcpsEndpoint permettent de gérer les points d'extrémité d'un adaptateur MTA par différents fournisseurs de services. L'objet PcpsCms sert essentiellement à maintenir un ensemble de points d'extrémité et de services.

Tous les objets sont extensibles.

9.1.1 Objet PcpsService

L'objet service est constitué par l'entité à laquelle un client IPCablecom 1.0 est abonné. Il correspond à un numéro de téléphone et à toutes les fonctionnalités connexes. Le modèle de données permet de fournir plusieurs services en un point d'extrémité unique.

L'objet PcpsService contient les informations génériques suivantes (les indications détaillées complètes figurent dans le schéma XML de PCSP):

- ServiceId – identificateur propre au service;
- BillingId – identificateur d'un autre service, qui sera facturé au titre de l'activité correspondant à ce service;
- Fanion IsPrimary – dans le cas de services multiples fournis en un point d'extrémité, un service DOIT comporter ce fanion positionné de façon à indiquer le service par défaut à utiliser pour les appels sortants;
- PrimaryRingPattern – indexation par rapport à une table de cadence MTA, de façon à choisir pour ce service un type de sonnerie;
- état administratif de ce service (suspendu, activé, modification du numéro, etc.);
- DisplayName – information d'affichage utilisée pour la fonctionnalité d'indication du nom du demandeur (CNAM);
- DisplayNumber – information d'affichage utilisée pour la fonctionnalité présentation du numéro du demandeur (CND);
- réglage d'annonce (mise en service, langue, fuseau horaire, etc.);
- codes porteuses (code porteuse longue distance, code porteuse dans les limites d'une zone d'accès et de transport local, code porteuse international);
- commande de portabilité du numéro local (statut de portabilité, indicateur de vérification STP, etc.);
- fonctionnalités d'appel – un service comporte une liste des objets fonctionnalité d'appel souscrits (décrite dans [12] et dans [13]);
- extensions – cet objet est extensible en deux emplacements: le corps principal de l'objet et la liste de fonctionnalités d'appel.

9.1.2 Objet PcpsMta

Un adaptateur de terminal de média regroupe un ou plusieurs points d'extrémité (contenus physiquement dans l'adaptateur). Il contient les informations génériques suivantes (des indications détaillées complètes figurent dans le schéma XML de PCSP):

- numéro FQDN de l'adaptateur MTA, identifiant ce dernier de façon univoque;
- port récepteur de signalisation d'appel par le réseau de l'adaptateur MTA (valeur par défaut: 2427);
- nom de domaine complet du système pilote de terminaison de câblo-modem;
- zone horaire où se trouve matériellement l'adaptateur MTA;

- désignation du protocole de signalisation – cette valeur du protocole est choisie par défaut pour tous les points d'extrémité contenus, à moins d'un choix prioritaire effectué par un point d'extrémité particulier;
- désignation de codec – choix par défaut du codec pour tous les points d'extrémité contenus, sauf désignation prioritaire par un point d'extrémité particulier;
- point unique pour extension.

9.1.3 Objet PcpEndpoint

Un point d'extrémité désigne un port matériel sur un adaptateur MTA/une passerelle. Il contient les informations génériques suivantes (des indications détaillées complètes figurent dans le schéma XML de PCSP):

- identificateur EndpointId – identificateur propre à ce point d'extrémité;
- choix du protocole de signalisation – priorité facultative par rapport au réglage de l'adaptateur MTA;
- statut administratif du point d'extrémité (déconnecté, service normal, mode essai, etc.);
- choix du codec – priorité facultative par rapport au réglage MTA;
- point unique pour extension.

9.1.4 Objet PcpCms

Cet objet maintient les associations entre le point d'extrémité/serveurs CMS et services/serveurs CMS). Il contient les informations génériques suivantes (des indications détaillées complètes figurent dans le schéma XML de PCSP):

- nom de domaine complet identifiant en propre ce serveur CMS;
- point d'extrémité unique pour extension.

9.1.5 Relations entre objets

Sur la Figure 3, les lignes reliant les différentes catégories représentent des relations d'objets (appelées parfois associations). Les relations représentées à la Figure 3 DOIVENT être prises en charge:

- service/serveur CMS – un serveur CMS type possèdera un bloc de numéros de téléphone;
- point d'extrémité/serveur CMS – un point d'extrémité a besoin d'un serveur CMS à des fins de signalisation;
- service/point d'extrémité – un numéro de téléphone doit être rattaché à un point d'extrémité physique;
- point d'extrémité/adaptateur MTA – les adaptateurs MTA contiennent physiquement les points d'extrémité.

9.2 Les relations sont codées au moyen de l'entité PcpRelation.Codage XML

Les objets du modèle de données seront codés au moyen du langage XML.

9.2.1 Schéma XML de PCSP

Le schéma XML de PCSP figure à l'Annexe A. Ce schéma définit la syntaxe de codage XML applicable aux entités suivantes (les entités DOIVENT se conformer au schéma):

- objets PcpService, PcpEndpoint, PcpMta et PcpCms – constituent les principaux objets modèle de données;
- PcpRelation – entité permettant d'établir ou de supprimer des relations entre objets;

- PcsImportExport – format de document à vocation générale, qui peut contenir un grand nombre d'objets ou de relations. Il sera généralement utilisé afin d'exporter des jeux de données complets, d'un serveur de mise en service vers un serveur de gestion d'appel.

Le code DOIT être utilisé par des analyseurs syntaxiques XML de validation afin de déterminer la validité syntaxique des entités codées.

9.2.2 Exemples de codages d'entité PCSP

Des exemples de codage XML de toutes les entités modèle de données PCSP figurent à l'Appendice I.

9.2.3 Extensions d'objet

Le schéma XML de PCSP autorise la réalisation d'extensions sur tous les objets (PcspService, PcspEndpoint, PscpMta et PcspCms) une fois que l'élément <Extension> y a été placé. La plupart des objets spécifient cet élément à la fin du corps principal. L'entité PcspService contient un élément <Extension> supplémentaire à la fin de la liste des fonctionnalités d'appel.

Ci-dessous figurent quelques règles simples concernant l'élément <Extension>.

- Tous les éléments <Extension> DOIVENT spécifier une définition de l'espace de nommage.
- Tous les sous-éléments de <Extension> DOIVENT remplir les conditions requises concernant l'espace de nommage.

Ces deux règles permettent au système d'analyse syntaxique XML de valider le contenu <Extension> par rapport à un fichier codé XML communiqué par un fournisseur. L'Appendice II contient un exemple d'extension.

10 Messagerie

10.1 Aperçu général

L'interface PCSP est conforme à un modèle de service Web. L'interface DOIT utiliser des messages SOAP 1.1 afin de transférer des entités codées XML (à partir du modèle de données PCSP) entre un client et un serveur. Les messages DOIVENT être acheminés entre un client et un serveur au moyen du protocole HTTP 1.1. Un examen complet des problèmes de transport figure à l'Appendice III.

L'interface est configurée suivant un modèle synchrone de demande/réponse (ou appel de procédure à distance (RPC, *remote procedure call*). Les modèles de message suivants sont pris en charge entre le client et le serveur.

- Message PUT – Le client rédige à destination du serveur un ou plusieurs objets ou relations codés XML. La création de nouveaux objets, comme la modification d'objets existants sont prises en charge.
- Message DELETE – Le client demande la suppression d'un ou plusieurs objets ou relations à partir du serveur.
- Message GET – Lecture d'un ou plusieurs objets codés XML à partir du serveur (objets uniquement ... les relations ne sont pas prises en charge).
- Message CMDSTATUS – Message permettant de transférer entre le client et le serveur des instructions "hors bande" et des statuts. Le client peut notifier au serveur différents états. Le client peut donner instruction au serveur d'exécuter différentes actions. Ce message est extensible au vendeur.

10.2 Prescriptions concernant le rôle d'émission de messages des serveurs CMS et PS

En règle générale, l'implémentation du serveur CMS comme celle du serveur PS PEUT prendre totalement en charge les rôles de messagerie du client et du serveur. Toutefois, dans le cadre de la mise en service par le serveur de gestion d'appels IPCablecom, les serveurs CMS et PS observent les prescriptions spécifiées dans le Tableau 1 quant à ce rôle.

Tableau 1/J.177 – Rôles de messagerie des serveurs CMS et PS

Message	CMS client	CMS serveur	PS client	PS serveur
GET	FACULTATIF	DOIT	DOIT	FACULTATIF
PUT	FACULTATIF	DOIT	DOIT	FACULTATIF
DELETE	FACULTATIF	DOIT	DOIT	FACULTATIF
CMDSTATUS	DOIT	DOIT	DOIT	DOIT

Les points suivants doivent être signalés:

- un serveur CMS DOIT prendre en charge le rôle de serveur pour les messages GET, PUT et DELETE;
- un serveur PS DOIT prendre en charge le rôle client pour les messages GET, PUT et DELETE;
- les serveurs CMS et PS DOIVENT prendre en charge les rôles de client et de serveur pour le message CMDSTATUS;
- tous les autres comportements sont FACULTATIFS.

Ces prescriptions assurent la mise en service des flux de données du serveur PS vers le serveur CMS et garantissent en outre que le serveur CMS n'est pas tenu de renvoyer vers le serveur PS des modifications de données dynamiques (modifications de fonctionnalité d'appel révisables par l'utilisateur, etc.).

Le serveur PS est en mesure de lire des objets spécifiques provenant du serveur CMS. Ce cas d'utilisation est pris en charge essentiellement pour permettre au serveur PS d'accéder au réglage de fonctionnalités d'appel de l'utilisateur ("données dynamiques") détenues par le serveur CMS, grâce à la lecture d'objets spécifiques PcpService à partir du serveur CMS.

La Figure 4 représente toutes les fonctions de messagerie requises.

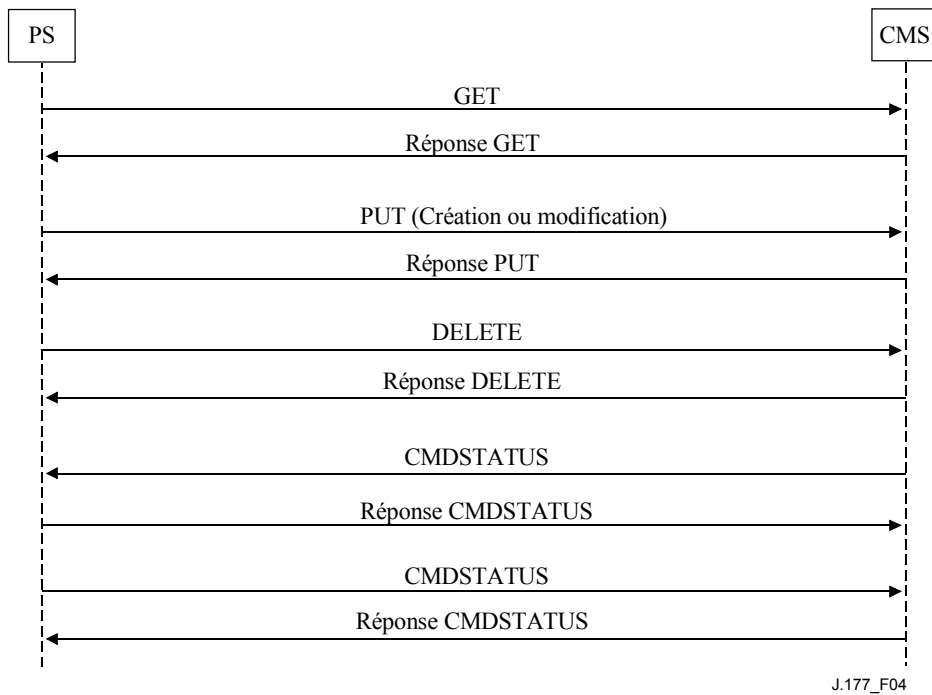


Figure 4/J.177 – Flux requis de messagerie

10.3 Spécification WSDL

La spécification de l'interface PCSP utilise le langage WSDL 1.1. A l'instar d'une spécification CORBA IDL, la définition d'interface WSDL spécifie les méthodes distantes utilisées à l'interface, les arguments acceptés par lesdites méthodes, les valeurs obtenues par l'application de ces méthodes, ainsi que tout type de données propres à l'interface dont la définition est obligatoire. De plus, la définition WSDL spécifie le format de codage du message (SOAP 1.1) et le lien de transport (HTTP 1.1).

Le langage WSDL est introduit dans différents outils Web disponibles pour la plupart des systèmes d'exploitation et des langages, afin de générer des talons d'interface client, des serveurs type et une prise en charge d'agencement de données SOAP.

Les serveurs PCSP DOIVENT s'appuyer sur la spécification WSDL 1.1 et DOIVENT observer le protocole de lien de transport HTTP 1.1 et enfin DOIVENT observer la définition WSDL présentée à l'Annexe B.

11 Sécurité

Le protocole IPsec ESP en mode transport assure la sécurisation de l'interface PCSP. La gestion de clé est implémentée par un échange de clés Internet où les clés sont partagées à l'avance. Cette infrastructure de sécurité est déjà utilisée au niveau du serveur CMS pour différentes interfaces. Des indications complètes figurent dans [3].

Annexe A

Schéma XML de PCSP

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- mis en forme sous XML Spy v4.0.1 U (http://www.xmlspy.com) - Paul Duffy (à
titre privé) -->
<!--
schéma PCSP de fourniture de service à l'abonné par le serveur de gestion
d'appels IPCablecom

Le protocole PCSP définit une interface de messagerie ainsi qu'un format de
codage XML applicable aux objets transmis par cette interface. Ce schéma
définit la syntaxe de codage XML pour les objets transmis par l'interface
PCSP.

Le codage des objets PcpsService, PcpsEndpoint, PcpsMta, et PcpsCms est
spécifié.
Un codage PcpsRelation décrit les associations entre objets.
Un codage PcpsImportExport permet d'obtenir des documents d'instance qui
contiennent de nombreux objets.
-->
<xs:schema targetNamespace="unique_fully_qualified_namespace"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="unique_fully_qualified_namespace" elementFormDefault="qualified">
<!-- NOTE: le paramètre <unique_fully_qualified_namespace> DOIT être remplacé
par des identificateurs uniques pour assurer l'implémentation effective du
schéma XML.
-->
===== DÉFINITIONS DE TYPES =====
-->
<!--
Chaîne non vide.
-->
<xs:simpleType name="nonEmptyString">
  <xs:restriction base="xs:string">
    <xs:minLength value="1"/>
  </xs:restriction>
</xs:simpleType>
<!--
Identificateur de service.
Chaîne non nulle contenant un attribut "format" (énumération), positionné
par défaut sur la valeur NSN.
-->
<xs:complexType name="ServiceIdType">
  <xs:simpleContent>
    <xs:extension base="nonEmptyString">
      <xs:attribute name="format" default="NSN">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="NSN"/>
            <xs:enumeration value="E164"/>
            <xs:enumeration value="ENUM"/>
            <xs:enumeration value="URL"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<!--
Type d'opération concernant une relation.
```



```

    Permet d'indiquer qu'une relation est "ajoutée" ou "supprimée"
-->
<xs:simpleType name="RelationOpType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="add"/>
    <xs:enumeration value="delete"/>
  </xs:restriction>
</xs:simpleType>
<!--
  Enumération de noms licites de "classes" d'objets.
-->
<xs:simpleType name="classType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="PcspService"/>
    <xs:enumeration value="PcspCms"/>
    <xs:enumeration value="PcspEndpoint"/>
    <xs:enumeration value="PcspMta"/>
  </xs:restriction>
</xs:simpleType>
<!--
  Liste de clés d'objet.
-->
<xs:complexType name="ListOfKeys">
  <xs:sequence>
    <xs:element name="Key" type="xs:string" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--
  Types de codec.
  Enumération qui correspond à l'objet PkttcCodecType
  de la spécification de codecs audio/vidéo IPCablecom™
  PKT-SP-CODEC-I03-011221.
  Cette énumération doit rester synchronisée avec la spécification
susmentionnée.

```

En ce qui concerne l'objet PcspEndpoint, une valeur de codec "inconnu" doit être interprétée comme équivalente à "utiliser la spécification de codec de l'adaptateur MTA".

```

-->
<xs:simpleType name="codecType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="other"/>
    <xs:enumeration value="unknown"/>
    <xs:enumeration value="G729"/>
    <xs:enumeration value="reserved"/>
    <xs:enumeration value="G729E"/>
    <xs:enumeration value="PCMU"/>
    <xs:enumeration value="G726-32"/>
    <xs:enumeration value="G728"/>
    <xs:enumeration value="PCMA"/>
    <xs:enumeration value="G726-16"/>
    <xs:enumeration value="G726-24"/>
    <xs:enumeration value="G726-40"/>
  </xs:restriction>
</xs:simpleType>
<!--
  Désignations de protocoles de signalisation.
  L'objet PcspEndpoint utilise "MtaDefault" pour forcer la valeur de réglage
par défaut du protocole de l'adaptateur MTA.
-->
<xs:simpleType name="protocolType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="MCGP 1.0 NCS 1.0"/>

```

```

        <xs:enumeration value="MtaDefault"/>
    </xs:restriction>
</xs:simpleType>
<!--
    Désignation numérique du fuseau horaire selon le document RFC 1123.
-->
<xs:simpleType name="NumTimezoneType">
    <xs:restriction base="xs:string">
        <xs:pattern value="[\+|-]\d{4}"/>
    </xs:restriction>
</xs:simpleType>
<!--
    Désignation ISO du fuseau horaire Continent/Ville ou Pays/Ville.
-->
<xs:simpleType name="IsoTimezoneType">
    <xs:element name="Country" type="xs:string"/>
    <xs:element name="City" type="xs:string"/>
</xs:simpleType>

<xs:complexType name="timezoneType">
    <xs:choice>
        <xs:element ref="NumTimezoneType">
        <xs:element ref="IsoTimezoneType">
    </xs:choice>
</xs:complexType>
<!--
    Statut de portage du numéro local

    0: non porté.
    1: importé (détenu par un autre fournisseur de services de
    télécommunication)
    2: exporté (prêté à un autre fournisseur de services de
    télécommunication)
-->
<xs:simpleType name="portingStatusType">
    <xs:restriction base="xs:integer">
        <xs:enumeration value="0"/>
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
    </xs:restriction>
</xs:simpleType>
<!--
    ===== DÉFINITIONS COMPLÉMENTAIRES D'ÉLÉMENTS =====
-->
<!--
    Restriction d'annonce réseau. Contient...

    Langage - selon type de langage du schéma XML.

    Fuseau horaire - voir définition ci-dessus.
-->
<xs:element name="Announcements">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="Language" type="xs:language"/>
            <xs:element name="Timezone" type="timezoneType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!--
    Codes de porteuse intercirconscription. Utilisés pour acheminer vers des
    porteuses particulières des appels régionaux et internationaux longue
    distance hors réseau.

```

PIC - porteuse prédésignée intercirconscription (longue distance).

LPIC - porteuse prédésignée d'accès et de transport local

IPIC - porteuse prédésignée internationale

-->

```
<xs:element name="InterExchange">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="PIC" type="xs:int"/>
      <xs:element name="LPIC" type="xs:int"/>
      <xs:element name="IPIC" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

<!--

Paramètres de portabilité du numéro local.

PortingStatus - voir type de statut de portabilité

LNPT - Le déclencheur de portabilité de numéro local détermine si le numéro est en transition

Faux/0: pas de vérification requise du point de transfert de signalisation.

Vrai/1: vérification requise du point de transfert de signalisation pour déterminer le numéro local d'acheminement du commutateur de destination.

-->

```
<xs:element name="LNP">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="PortingStatus" type="portingStatusType"/>
      <xs:element name="LNPT" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

<!--

Elément extension vendeur.

Utilisé dans les objets PcspService, PcspCms, PcspMta, et PcspEndpoint afin d'autoriser les extensions vendeur.

Egalement utilisé pour étendre la liste des fonctionnalités d'appel contenue dans l'objet PcspService.

-->

```
<xs:element name="Extension">
  <xs:complexType>
    <xs:sequence>
      <xs:any namespace="##any" processContents="strict" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

<!--

===== FONCTIONNALITÉS D'APPEL =====

Un service comporte une liste d'objets fonctionnalité d'appel, dont chacun code une des fonctionnalités d'appel décrites dans PKT-TR-VOIPBRF-R01-000608 et PKT-TR-VOIPERF-R01-000831.

Chaque fonctionnalité d'appel contient ses propres données d'état

"statiques" (détenues par le serveur PS):

Nom de la fonctionnalité (désigné implicitement par le nom de l'élément)

Etat abonné/non-abonné,

Etat administratif de la fonctionnalité.

Nombre de fonctionnalités d'appel contiennent uniquement ces informations
L'absence d'une fonctionnalité d'appel particulière implique le non-
abonnement à cette dernière.

L'état abonné sert à indiquer qu'une fonctionnalité d'appel figurant
explicitement dans la liste ne fait pas l'objet d'un abonnement (cas
atypique).

Plusieurs fonctionnalités étendent le jeu de paramètres "statiques" par des
données propres à la fonctionnalité considérée. Ces données sont généralement
configurées par l'utilisateur (au moyen du combiné téléphonique ou en appelant
un CSR).

La spécification PCSP classe les données modifiables par l'utilisateur dans
la catégorie des données "dynamiques", ce qui signifie qu'elles sont détenues
par le serveur CMS. Les modifications apportées aux données dynamiques contenues
dans le serveur CMS ne doivent être renvoyées vers le serveur PS.

```
-->
<!--
  Toujours -
    faux/0: L'abonné peut modifier le numéro de renvoi.
    vrai/1: (Seul) le fournisseur de service peut modifier le numéro de
    renvoi
-->
<xs:element name="Always" type="xs:boolean"/>
<!--
  ForwardTo - Identificateur du service vers lequel l'appel sera renvoyé.
  NOTE - Les chaînes vides sont admises.
-->
<xs:element name="ForwardTo" type="xs:string"/>
<!--
  ListOfServiceId - liste d'identificateurs de Service.
-->
<xs:element name="ListOfServiceId">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ServiceId" type="xs:string" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--
  ListOfSpeedDial - liste d'identificateurs de Service/de paires de
  numérotation abrégée
  Chaque paire contient un numéro de numérotation abrégée à un ou deux
  chiffres, avec son identificateur de service associé.
-->
<xs:element name="SdPair">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SdNum">
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:minInclusive value="0"/>
            <xs:maxInclusive value="99"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="ServiceId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ListOfSpeedDial">
  <xs:complexType>
    <xs:sequence>
```

```

        <xs:element ref="SdPair" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<!--
    Définition de chaque fonctionnalité d'appel prise en charge...

    Toutes les fonctionnalités d'appel peuvent être considérées comme étant
    constituées de deux parties:
    1. une partie commune contenant l'état administratif de la fonctionnalité
    (les données "statiques").
    2. une partie facultative contenant les paramètres propres à ladite
    fonctionnalité, généralement configurés par l'utilisateur final (les données
    "dynamiques").
-->
<!--
    L'"objet de base" pour toutes les fonctionnalités d'appel contient:

    Abonné -
        0/faux: fonctionnalité non souscrite
        1/vrai: fonctionnalité souscrite.

    AdminStatus -
        0: fonctionnalité suspendue par le fournisseur de service.
        1: fonctionnalité mise en service par le fournisseur de service.

    En général, la présence d'une fonctionnalité d'appel implique qu'elle a été
    souscrite.
    Le fanion "souscrite" est pris en charge dans le cas atypique de la
    non-souscription d'une fonctionnalité d'appel dont le nom figure
    explicitement sur la liste.
-->
<xs:complexType name="CfBase">
    <xs:sequence>
        <xs:element name="Subscribed" type="xs:boolean"/>
        <xs:element name="AdminStatus">
            <xs:simpleType>
                <xs:restriction base="xs:int">
                    <xs:enumeration value="0"/>
                    <xs:enumeration value="1"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<!--
    Présentation du numéro du demandeur ("CND" Calling Number Delivery)
-->
<xs:element name="CfCND">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="CfBase"/>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!--
    Présentation du nom du demandeur ("CNAM": Calling Name Delivery)
-->
<xs:element name="CfCNAM">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="CfBase"/>
        </xs:complexContent>
    </xs:complexType>

```

```

</xs:element>
<!--
  Présentation de l'identité du demandeur d'un appel en instance ("CIDCW":
Calling Identity Delivery on Call Waiting)
-->
<xs:element name="CfCIDCW">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
  Appel en instance ("CW": Call Waiting)
-->
<xs:element name="CfCW">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
  Annulation d'appel en instance ("CCW": Cancel Call Waiting (*70))
-->
<xs:element name="CfCCW">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
  Renvoi d'appel variable et fonction de l'utilisation ("CFV": Call
Forwarding Variable and Usage- Sensitive Call Forwarding (*72/*73)

  Etend CfBase par l'adjonction des éléments suivants:

  Activation -
  0/faux: l'utilisateur a désactivé la fonctionnalité (*73).
  1/vrai: l'utilisateur a activé la fonctionnalité (*72).
-->
<xs:element name="CfCFV">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
  Rappel automatique ("AR": Automatic Recall (*69))
-->
<xs:element name="CfAR">
  <xs:complexType>

```

```

        <xs:complexContent>
          <xs:extension base="CfBase"/>
        </xs:complexContent>
      </xs:complexType>
    </xs:element>
  <!--
    Rétroappel automatique ("AC": Automatic Callback (*66))
  -->
  <xs:element name="CfAC">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="CfBase"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <!--
    Indicateur de message visuel en instance ("VMWI": Visual Message Waiting
    Indicator)
  -->
  <xs:element name="CfVMWI">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="CfBase"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <!--
    Suivi demandé par un client ("COT": Customer Originated Trace (*57))
  -->
  <xs:element name="CfCOT">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="CfBase"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <!--
    Conversation à trois/sensible à l'utilisation ("TWC": Three-Way Calling /
    Usage-Sensitive Three-Way Calling (*71))
  -->
  <xs:element name="CfTWC">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="CfBase"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <!--
    Activation à distance du renvoi d'appel ("RACF": Remote Activation of Call
    Forwarding)
  -->
  <xs:element name="CfRACF">
    <xs:complexType>
      <xs:complexContent>
        <xs:extension base="CfBase"/>
      </xs:complexContent>
    </xs:complexType>
  </xs:element>
  <!--
    Avis d'appel externe ("OCAA": Outside Calling Area Alerting)
  -->
  <xs:element name="CfOCAA">
    <xs:complexType>
      <xs:complexContent>

```

```

        <xs:extension base="CfBase"/>
    </xs:complexContent>
</xs:complexType>
</xs:element>
<!--
    Identité du demandeur avec filtrage renforcé ("CIES": Calling Identity with
    Enhanced Screening)
-->
<xs:element name="CfCIES">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="CfBase"/>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!--
    Refus des appels anonymes ("ACR": Anonymous Call Rejection) (*77/*87)

    Extension de CfBase par les paramètres suivants:

    Activation -
    0/faux: l'utilisateur a désactivé cette fonctionnalité (*87).
    1/vrai: l'utilisateur a activé cette fonctionnalité (*77).
-->
<xs:element name="CfACR">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="CfBase">
                <xs:sequence>
                    <xs:element name="UserParams" minOccurs="0">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element name="Active" type="xs:boolean"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!--
    Rétroappel automatique restreint ("AC-R": Automatic Callback - Restrict)
-->
<xs:element name="CfAC-R">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="CfBase"/>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!--
    Blocage du rappel automatique ("ACB": Automatic Recall Blocking)
-->
<xs:element name="CfACB">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="CfBase"/>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
<!--
    Blocage de la présentation d'identité du demandeur ("CIDB" Calling Identity
    Delivery Blocking) (*67/*82).

```


Extension de CfBase par les paramètres suivants:

Fanion -

"PUBLIC": présenter les informations d'identification du demandeur

"ANONYMOUS": ne pas présenter les informations d'identification du demandeur.

-->

```
<xs:element name="CfCIDB">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Flag">
                  <xs:simpleType>
                    <xs:restriction base="xs:string">
                      <xs:enumeration value="PUBLIC"/>
                      <xs:enumeration value="ANONYMOUS"/>
                    </xs:restriction>
                  </xs:simpleType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

<!--

Renvoi en cas d'occupation ("CFBL" *Call Forwarding Busy Line*)
(*68/*40/*88).

Extension de CfBase par les paramètres dynamiques suivants, modifiables par l'utilisateur (détenus par le serveur CMS).

Activation -

0/faux: l'utilisateur a désactivé cette fonctionnalité (*88).

1/vrai: l'utilisateur a activé cette fonctionnalité (*68/*40).

Toujours - Voir définition précédente.

Numéro de renvoi - Voir définition précédente.

-->

```
<xs:element name="CfCFBL">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element ref="Always"/>
                <xs:element ref="ForwardTo"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:element>
```

```

    </xs:complexType>
</xs:element>
<!--
Renvoi d'appel sur non-réponse ("CFDA" Call Forwarding Don't Answer)
(*68/*42/*88)

Extension de CfBase par les paramètres dynamiques suivants, modifiables par
l'utilisateur (détenus par le serveur CMS):

Activation -
    0/faux: l'utilisateur a désactivé cette fonctionnalité (*88).
    1/vrai: l'utilisateur a activé cette fonctionnalité (*68/*42).

Toujours - Voir définition précédente.

Durée de la sonnerie - Nombre de cycles de sonnerie précédant l'activation
du renvoi.

Numéro de renvoi - Voir définition précédente.
-->
<xs:element name="CfCFDA">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element ref="Always"/>
                <xs:element name="RingPeriod" type="xs:int"/>
                <xs:element ref="ForwardTo"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
Association de renvoi d'appel ("CFC" Call Forwarding Combination)

Extension de CfBase par les paramètres dynamiques suivants, modifiables par
l'utilisateur (détenus par le serveur CMS):

Activation -
    0/faux: l'utilisateur a désactivé cette fonctionnalité (*88).
    1/vrai: l'utilisateur a activé cette fonctionnalité (*68).

Toujours - voir définition précédente.

Durée de la sonnerie - Nombre de cycles de sonnerie précédant l'activation
du renvoi.

Numéro de renvoi - Voir définition précédente.
-->
<xs:element name="CfCFC">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">

```

```

        <xs:complexType>
          <xs:sequence>
            <xs:element name="Active" type="xs:boolean"/>
            <xs:element ref="Always"/>
            <xs:element name="RingPeriod" type="xs:int"/>
            <xs:element ref="ForwardTo"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<!--

```

Renvoi d'appel sélectif ("SCF" *Selective Call Forwarding*) (*63/*83).

Extension de CfBase par les paramètres dynamiques suivants, modifiables par l'utilisateur (détenus par le serveur CMS) :

Activation -

0/faux: l'utilisateur a désactivé cette fonctionnalité (*83).
 1/vrai: l'utilisateur a activé cette fonctionnalité (*63).

ListOfServiceId - liste des identificateurs de service à réacheminer. Voir définition de l'élément précédent.

ForwardTo - service vers lequel le renvoi doit s'effectuer. Voir définition de l'élément précédent.

```

-->
<xs:element name="CfSCF">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element ref="ListOfServiceId"/>
                <xs:element ref="ForwardTo"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--

```

Refus d'appel sélectif ("SCR" *Selective Call Rejection*) (*60/*80).

Extension de CfBase par les paramètres dynamiques suivants, modifiables par l'utilisateur (détenus par le serveur CMS) :

Activation -

0/faux: l'utilisateur a désactivé cette fonctionnalité (*80).
 1/vrai: l'utilisateur a activé cette fonctionnalité (*60).

ListOfServiceIds - liste des identificateurs de service à refuser. Voir définition de l'élément précédent.

```

-->
<xs:element name="CfSCR">
  <xs:complexType>

```

```

<xs:complexContent>
  <xs:extension base="CfBase">
    <xs:sequence>
      <xs:element name="UserParams" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Active" type="xs:boolean"/>
            <xs:element ref="ListOfServiceId"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<!--
Sonnerie distinctive / Appel en instance ("DRCW" Distinctive Ringing/Call
Waiting) (*61/*81)

Extension de CfBase par les paramètres dynamiques suivants, modifiables par
l'utilisateur (détenus par le serveur CMS):

Activation -
0/faux: l'utilisateur a désactivé cette fonctionnalité (*81).
1/vrai: l'utilisateur a activé cette fonctionnalité (*61).

ListOfServiceIds - liste des identificateurs de service entrant qui
déclencheront une sonnerie distinctive (et non la sonnerie de tonalité normale
ou la tonalité d'attente). Voir la définition de l'élément précédent.
-->
<xs:element name="CfDRCW">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element ref="ListOfServiceId"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
Numérotation abrégée ("SPCALL" Speed Calling) (*74 / *75)

Extension de CfBase par les paramètres dynamiques suivants, modifiables par
l'utilisateur (détenus par le serveur CMS):

ListOfSpeedDial - Voir la définition de l'élément précédent.
-->
<xs:element name="CfSPCALL">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>

```

```

        <xs:sequence>
          <xs:element ref="ListOfSpeedDial"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<!--
  Service d'alerte distinctive de résidence ("RDA" Residence Distinctive
  Alerting Service.)
-->
<xs:element name="CfRDA">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase"/>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
  Restriction de service de la ligne ("LSR" Line Service Restriction)

  Extension de CfBase par les paramètres dynamiques suivants, modifiables par
  l'utilisateur (détenus par le serveur CMS):

  BlkDomLongDist - blocage des appels nationaux longue distance sortants.
    0/faux: pas de blocage.
    1/vrai: blocage.
  BlkIntlLongDist - blocage des appels internationaux longue distance
  sortants.
    0/faux: pas de blocage.
    1/vrai: blocage.
  BlkPayPerCall - blocage des appels facturés sortants (900/976).
    0/faux: pas de blocage.
    1/vrai: blocage.
  BlkOperatorAssist - blocage des appels sortants assistés par opérateur.
    0/faux: pas de blocage.
    1/vrai: blocage.
  BlkDirAssist - blocage des appels sortants d'assistance annuaire.
    0/faux: pas de blocage.
    1/vrai: blocage.
  BlkTollFree - blocage des appels gratuits sortants
    0/faux: pas de blocage.
    1/vrai: blocage.
  ServiceList - liste des identificateurs de service relatifs pour les appels
  nationaux longue distance qui restent autorisés.
-->
<xs:element name="CfLSR">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="BlkDomLongDist" type="xs:boolean"/>
                <xs:element name="BlkIntLongDist" type="xs:boolean"/>
                <xs:element name="BlkPayPerCall" type="xs:boolean"/>
                <xs:element name="BlkOperatorAssist" type="xs:boolean"/>
                <xs:element name="BlkDirAssist" type="xs:boolean"/>
                <xs:element name="BlkTollFree" type="xs:boolean"/>
                <xs:element ref="ListOfServiceId"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>

```

```

        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<!--
  Service "ne pas déranger" ("DND" Do Not Disturb)

  Extension de CfBase par les paramètres dynamiques suivants, modifiables par
  l'utilisateur (détenus par le serveur CMS).

  Activation -
    0/faux: l'utilisateur a désactivé cette fonctionnalité.
    1/vrai: l'utilisateur a activé cette fonctionnalité.

  WeekDayStartTod1 - heure de début du service DND pendant la semaine.
  WeekDayStopTod1 - heure d'arrêt du service DND pendant la semaine.
  WeekDayStartTod2 - heure de début du service DND pendant la semaine.
  WeekDayStopTod2 - heure d'arrêt du service DND pendant la semaine.
  WeekEndStartTod1 - heure de début du service DND pendant le week-end.
  WeekEndStopTod1 - heure d'arrêt du service DND pendant le week-end.
  WeekEndStartTod2 - heure de début du service DND pendant le week-end.
  WeekEndStopTod2 - heure d'arrêt du service DND pendant le week-end.
-->
<xs:element name="CfDND">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element name="WdStartTod1" type="xs:time"/>
                <xs:element name="WdStopTod1" type="xs:time"/>
                <xs:element name="WdStartTod2" type="xs:time"/>
                <xs:element name="WdStopTod2" type="xs:time"/>
                <xs:element name="WeStartTod1" type="xs:time"/>
                <xs:element name="WeStopTod1" type="xs:time"/>
                <xs:element name="WeStartTod2" type="xs:time"/>
                <xs:element name="WeStopTod2" type="xs:time"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
<!--
  Limitation du nombre d'appels ("COC" Curfew on Calls).

  Extension de CfBase par les paramètres dynamiques suivants, modifiables par
  l'utilisateur (détenus par le serveur CMS).

  Activation -
    0/faux: l'utilisateur a désactivé cette fonctionnalité.
    1/vrai: l'utilisateur a activé cette fonctionnalité.

  StartTod - heure de début du service COC.
  StopTod - heure d'arrêt du service COC.

```

ServiceList - liste des identificateurs de service pour les appels entrants et sortants autorisés à contourner l'interdiction d'annonce d'offre.

-->

```
<xs:element name="CfCOC">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element name="StartTod" type="xs:time"/>
                <xs:element name="StopTod" type="xs:time"/>
                <xs:element ref="ListOfServiceId"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

<!--

Interdiction d'annonce d'offre ("NSA" *No Solicitation Announcement*)

Extension de CfBase par les paramètres dynamiques suivants, modifiables par l'utilisateur (détenus par le serveur CMS).

Activation -

0/faux: l'utilisateur a désactivé cette fonctionnalité.

1/vrai: l'utilisateur a activé cette fonctionnalité.

StartTod - heure de début du service COC.

StopTod - heure d'arrêt du service COC.

ServiceList - liste des identificateurs de service identifiants pour les services entrants et sortants habilités à contourner l'interdiction d'annonce d'offre.

-->

```
<xs:element name="CfNSA">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="CfBase">
        <xs:sequence>
          <xs:element name="UserParams" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="Active" type="xs:boolean"/>
                <xs:element name="StartTod" type="xs:time"/>
                <xs:element name="StopTod" type="xs:time"/>
                <xs:element ref="ListOfServiceId"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

<!--

liste des fonctionnalités d'appel. La liste peut contenir une au plus de chacune des fonctionnalités d'appel définies ci-dessus, ainsi que toute fonctionnalité d'appel des extensions vendeur.

-->

```

<xs:element name="ListOfCallFeatures">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="CfCND" minOccurs="0"/>
      <xs:element ref="CfCNAM" minOccurs="0"/>
      <xs:element ref="CfCIDCW" minOccurs="0"/>
      <xs:element ref="CfCW" minOccurs="0"/>
      <xs:element ref="CfCCW" minOccurs="0"/>
      <xs:element ref="CfCFV" minOccurs="0"/>
      <xs:element ref="CfAR" minOccurs="0"/>
      <xs:element ref="CfAC" minOccurs="0"/>
      <xs:element ref="CfVMWI" minOccurs="0"/>
      <xs:element ref="CfCOT" minOccurs="0"/>
      <xs:element ref="CfTWC" minOccurs="0"/>
      <xs:element ref="CfRACF" minOccurs="0"/>
      <xs:element ref="CfOCAA" minOccurs="0"/>
      <xs:element ref="CfCIES" minOccurs="0"/>
      <xs:element ref="CfACR" minOccurs="0"/>
      <xs:element ref="CfAC-R" minOccurs="0"/>
      <xs:element ref="CfACB" minOccurs="0"/>
      <xs:element ref="CfCIDB" minOccurs="0"/>
      <xs:element ref="CfCFBL" minOccurs="0"/>
      <xs:element ref="CfCFDA" minOccurs="0"/>
      <xs:element ref="CfCFC" minOccurs="0"/>
      <xs:element ref="CfSCF" minOccurs="0"/>
      <xs:element ref="CfSCR" minOccurs="0"/>
      <xs:element ref="CfDRCW" minOccurs="0"/>
      <xs:element ref="CfSPCALL" minOccurs="0"/>
      <xs:element ref="CfRDA" minOccurs="0"/>
      <xs:element ref="CfLSR" minOccurs="0"/>
      <xs:element ref="CfDND" minOccurs="0"/>
      <xs:element ref="CfCOC" minOccurs="0"/>
      <xs:element ref="CfNSA" minOccurs="0"/>
      <xs:element ref="Extension" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```
<!--
```

```
===== DÉFINITIONS DES PRINCIPAUX OBJETS =====
```

6 codages sont définis dans le schéma PCSP.

Les 4 principaux codages d'objets:

PcspCms - serveur CMS. Ensemble de services et de points d'extrémité.

PcspService - correspond à un numéro de téléphone, à sa configuration et aux fonctionnalités d'appel.

PcspMta - correspond à un adaptateur MTA matériel et à sa configuration. Ensemble de points d'extrémité.

PcspEndpoint - correspond à un point d'extrémité sur un adaptateur MTA.

Objet PcspRelation. Cet objet code les associations entre objets.

Objet PcspImportExport. Sert à créer un fichier de chargement en bloc pour le serveur CMS.

```
-->
<!--
```

```
PcspRelation.
```

L'objet relation spécifie les associations entre les objets PcspCms, PcspService, PcspEndpoint, et PcspMta.

L'attribut "relOp" a spécifié si la relation est ajoutée ou supprimée.

```
-->
```



```

<xs:element name="PcspRelation">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Class1" type="classType"/>
      <xs:element name="Key" type="xs:string"/>
      <xs:element name="Class2" type="classType"/>
      <xs:element name="ListOfKeys" type="ListOfKeys"/>
    </xs:sequence>
    <xs:attribute name="relOp" type="RelationOpType" use="required"/>
  </xs:complexType>
</xs:element>
<!--
  Objet PcspCms.

```

Cet objet actualise les associations entre les points d'extrémité, les services et les serveurs gestionnaires correspondants.

Contenu...

CmsFqdn - Nom de domaine complet identifiant de façon exclusive ce serveur CMS.

```

-->
<xs:element name="PcspCms">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="CmsFqdn" type="nonEmptyString"/>
      <xs:element ref="Extension" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--
  Objet PcspEndpoint.

```

Un point d'extrémité est un port physique sur une passerelle/adaptateur MTA.

Contenu...

EndpointId - Identificateur exclusif de ce point d'extrémité. Format défini par la spécification

"IPCablecom Network Based Call Signalling Protocol Specification".

Exemple: "aaln/1@mta01.cablelabs.com"

AdminStatus -

0: point d'extrémité déconnecté

1: normal - point d'extrémité en service

2: mode test - point d'extrémité testé.

Protocole - neutralisation facultative du réglage de protocole MTA.

Codec - neutralisation facultative du réglage de codec MTA.

```

-->
<xs:element name="PcspEndpoint">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="EndpointId" type="nonEmptyString"/>
      <xs:element name="AdminStatus">
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:enumeration value="0"/>
            <xs:enumeration value="1"/>
            <xs:enumeration value="2"/>
          </xs:restriction>
        </xs:simpleType>

```

```

    </xs:element>
    <xs:element name="Protocol" type="protocolType"/>
    <xs:element name="Codec" type="codecType"/>
    <xs:element ref="Extension" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<!--

```

Objet PcspMta.

Un adaptateur de terminal de média comprend un ou plusieurs points d'extrémité (situés physiquement dans l'adaptateur MTA).

Contenu...

MtaFqdn - nom de domaine complet identifiant de façon exclusive cet adaptateur MTA.

MtaPort - port de réception de signalisation d'appel par le réseau de l'adaptateur MTA (valeur par défaut: 2427)

CmtsFqdn - nom de domaine complet du système CMTS de commande. Ce nom est indispensable au serveur CMS pour établir la qualité de fonctionnement dynamique de l'adaptateur MTA avec le système de terminaison CMTS approprié.

Timezone - fuseau horaire dans lequel cet adaptateur MTA se trouve physiquement. Format numérique de fuseau (RFC 1123) conformément au document.

Protocol - paramètre à positionner sur la valeur par défaut "MGCP 1.0 NCS 1.0", applicable à tous les points d'extrémité contenus dans l'adaptateur.

Codec - valeur par défaut pour tous les points d'extrémité contenus.

```

-->
<xs:element name="PcspMta">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="MtaFqdn" type="nonEmptyString"/>
      <xs:element name="ListenPort" type="xs:int"/>
      <xs:element name="CmtsFqdn" type="xs:string"/>
      <xs:element name="Timezone" type="timezoneType"/>
      <xs:element name="Protocol" type="protocolType"/>
      <xs:element name="Codec" type="codecType"/>
      <xs:element ref="Extension" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--

```

Objet PcspService.

Contenu...

ServiceId - identificateur exclusif du service.

AdminStatus -

0: suspendu (par ex. En cas de non-paiement des factures).

1: activé (état normal).

2: changement de numéro

3: hors service.

4: non attribué.

BillingId - Numéro de téléphone identifiant un autre service à facturer au lieu de ce service.

ExternalId - chaîne arbitraire utilisée pour transporter des données telles que l'identificateur d'abonné, etc.

IsPrimary - Lorsque plusieurs services sont fournis en un point d'extrémité, un service DOIT positionner ce fanion de façon à indiquer le service par défaut à utiliser pour les appels sortants.

faux/0: ce service n'est pas un service primaire.

vrai/1: ce service est un service primaire

PrimaryRing - Identificateur de type de sonnerie primaire. Index qui se réfère à la table de cadence de l'adaptateur MTA, de façon à choisir le type de sonnerie attribué à ce service.

DisplayName - Utilisé pour la fonctionnalité de présentation du nom du demandeur (CNAM)

DisplayNumber - Utilisé pour la fonctionnalité de présentation du numéro d'appel du demandeur (CND)

Password - différentes fonctionnalités d'appel exigent l'introduction d'un mot de passe avant d'autoriser toute modification.

Contrôle des annonces d'offre sur le réseau. Voir définition ci-dessus.

Réglage des codes interurbains et du paramètre portabilité du numéro local. Voir définitions précédentes.

Fonctionnalités d'appel. Voir définitions précédentes.

```
-->
<xs:element name="PcspService">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ServiceId" type="ServiceIdType"/>
      <xs:element name="AdminStatus">
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:enumeration value="0"/>
            <xs:enumeration value="1"/>
            <xs:enumeration value="2"/>
            <xs:enumeration value="3"/>
            <xs:enumeration value="4"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="BillingId" type="ServiceIdType"/>
      <xs:element name="ExternalId" type="xs:string"/>
      <xs:element name="IsPrimary" type="xs:boolean"/>
      <xs:element name="PrimaryRing" type="xs:string"/>
      <xs:element name="DisplayName" type="xs:string"/>
      <xs:element name="DisplayNumber" type="xs:string"/>
      <xs:element name="Password" type="xs:string"/>
      <xs:element ref="Announcements"/>
      <xs:element ref="InterExchange"/>
      <xs:element ref="LNP"/>
      <xs:element ref="ListOfCallFeatures"/>
      <xs:element ref="Extension" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!--
```

Format de fichier Import/Export.

Pour le transfert d'un ou plusieurs objets et relations entre serveurs PS/CMS.

NOTE - PcspCms n'est pas inclus. Rien ne justifie actuellement l'obtention par un serveur CMS de son propre objet CMS en provenance du serveur PS.

```
-->
<xs:element name="PcspImportExport">
  <xs:complexType>
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="PcspService"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

```

        <xs:element ref="PcspEndpoint"/>
        <xs:element ref="PcspMta"/>
        <xs:element ref="PcspRelation"/>
    </xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>

```

Annexe B

Spécification WSDL pour messagerie PCSP

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
    Interface de fourniture de service à l'abonné par le serveur de gestion
    d'appels IPCablecom spécifiée en langage WSDL 1.1.
-->
<definitions name="PcspI01Service"
targetNamespace="unique_fully_qualified_namespace"
xmlns:tns="unique_fully_qualified_namespace"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
xmlns="http://schemas.xmlsoap.org/wsdl/">
<!-- NOTE - L'implémentation effective DOIT comporter le remplacement du
paramètre <unique_fully_qualified_namespace> par les identificateurs uniques
complets.
-->
<!--
    La section <types> définit les types de données particuliers requis par
l'interface.

```

PCSPI01 prescrit deux types de données particuliers:

PcspArg (et tableaux)
PcspObj (et tableaux).

```

// PcspArg (pseudo-code)
//
classe PcspArg
{
    // EntityName et clé d'un objet spécifique.
    // Les caractères génériques ne sont actuellement pas autorisés.
    // Il n'est pas tenu compte de la clé pour l'entité PcspRelation.
    //
    Chaîne entityName;
    Clé de chaîne;

    // Réserve pour utilisation future. Mis à zéro pour l'instant.
    //
    fanions int;
}

// PcspObj (pseudo-code).
//
classe PcspObj
{
    // EntityName et clé de l'objet spécifique.
    // Il n'est pas tenu compte de la clé pour l'entité PcspRelation.
    //

```

```

    Chaîne entityName;
    Clé de chaîne;

    // cmdStatus:
    // PcspObj en tant que résultat/conclusion de la méthode - DOIT être
positionné sur l'un des codes de statut spécifiés ci-dessous.
    // PcspObj en argument de Put() - DOIT être positionné sur l'une des
valeurs suivantes:
    //      1: création d'un nouvel objet
    //      2: modification d'un objet existant.
    // Il n'est pas tenu compte de ce champ pour l'entité PcspRelation.
    //
    int cmdStatus;

    // codage XML suivant le système PCSP Data Model Schema ou 0 (nul)
    //
    Chaîne xmlEncoding;
}

```

EntityNames; DOIT figure parmi les noms suivants:

```

"PcspService"
"PcspMta"
"PcspEndpoint"
"PcspCms"
"PcspRelation"

```

Codes de statut: Codage du résultat de la méthode ou valeur contenue dans le champ cmdStatus d'un résultat PcspObj.

```

0, Opération aboutie
1, Objet non trouvé
2, Spécification de mode Put() non valide.
3, Création d'objet non aboutie; l'objet existe déjà
4, Opération de lecture non aboutie
5, Opération de création non aboutie
6, Opération de modification non aboutie
7, Opération de suppression non aboutie
8, Problème interne.
9, Serveur occupé
10, Opération non prise en charge.
11, Extension vendeur.

```

... extension en fonction des besoins ...

```

-->
<types>
  <schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.IPCablecom.com/pcsp/i01">
    <complexType name="PcspObj">
      <sequence>
        <element name="entityName" type="string"/>
        <element name="key" type="string"/>
        <element name="cmdStatus" type="int"/>
        <element name="xmlEncoding" type="string"/>
      </sequence>
    </complexType>
    <complexType name="ArrayOfPcspObj">
      <complexContent>
        <restriction base="soapenc:Array">
          <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns:PcspObj []"/>
        </restriction>
      </complexContent>
    </complexType>

```

```

    <complexType name="PcspArg">
      <sequence>
        <element name="entityName" type="string"/>
        <element name="key" type="string"/>
        <element name="flags" type="int"/>
      </sequence>
    </complexType>
    <complexType name="ArrayOfPcspArg">
      <complexContent>
        <restriction base="soapenc:Array">
          <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns:PcspArg[]" />
        </restriction>
      </complexContent>
    </complexType>
  </schema>
</types>
<!--
  Section messages.

```

L'invocation d'une méthode à l'interface implique deux messages ... un message entrée et un message sortie.

Le premier contient la série des arguments d'entrée pour l'appel de la méthode.

Le second contient les valeurs renvoyées.

```

-->
<message name="Get0In">
  <part name="args" type="tns:ArrayOfPcspArg"/>
</message>
<message name="Get0Out">
  <part name="Result" type="tns:ArrayOfPcspObj"/>
</message>
<message name="Put1In">
  <part name="objs" type="tns:ArrayOfPcspObj"/>
</message>
<message name="Put1Out">
  <part name="Result" type="tns:ArrayOfPcspObj"/>
</message>
<message name="Delete2In">
  <part name="args" type="tns:ArrayOfPcspArg"/>
</message>
<message name="Delete2Out">
  <part name="Result" type="tns:ArrayOfPcspObj"/>
</message>
<message name="CmdStatus3In">
  <part name="isCmd" type="xsd:boolean"/>
  <part name="code" type="xsd:int"/>
  <part name="subCode" type="xsd:int"/>
  <part name="vendorExtension" type="xsd:string"/>
</message>
<message name="CmdStatus3Out">
  <part name="Result" type="xsd:int"/>
</message>
<!--

```

Le type de port définit l'interface.

Chaque "opération" correspond à l'application d'une méthode à l'interface, avec les messages connexes d'entrée et de sortie (arguments et valeurs renvoyées).

```

// Interface de service PCSP (en pseudo-code).
//
interface IPcspI01Service
{

```

```

// lecture (Get) d'un ou plusieurs objets provenant du serveur.
// EntityName de 'PcspRelation' n'est pas autorisé (objets uniquement)
//
PcspObj[] Get(PcspArg[] args);

// Ecriture (Put) objets et relations vers le serveur.
//
PcspObj[] Put(PcspObj[] objs);

// Suppression (Delete) d'objets et de relations provenant du serveur.
//
PcspObj[] Delete(PcspArg[] args);

// Notification de commande et de statut hors bande.
//
// Codes de commande prédéfinis:
// 0 - commande d'extension
//
// Codes de statut prédéfinis :
// 0 - Statut d'extension
//
int CmdStatust(booléen cmd, // vrai pour CMD, faux pour STATUS.
               int code, // code CMD ou STATUS (voir ci-dessus).
               int subCode // SubCode. Précise le code.
               Extension de chaîne);
}
-->
<portType name="PcspI01Service">
  <operation name="Get" parameterOrder="args">
    <input name="Get0In" message="tns:Get0In"/>
    <output name="Get0Out" message="tns:Get0Out"/>
  </operation>
  <operation name="Put" parameterOrder="objs">
    <input name="Put1In" message="tns:Put1In"/>
    <output name="Put1Out" message="tns:Put1Out"/>
  </operation>
  <operation name="Delete" parameterOrder="args">
    <input name="Delete2In" message="tns:Delete2In"/>
    <output name="Delete2Out" message="tns:Delete2Out"/>
  </operation>
  <operation name="CmdStatus" parameterOrder="isCmd code subCode
vendorExtension">
    <input name="CmdStatus3In" message="tns:CmdStatus3In"/>
    <output name="CmdStatus3Out" message="tns:CmdStatus3Out"/>
  </operation>
</portType>
<!--
Lier l'interface ("portType") aux caractéristiques de transport.
En principe les flux d'entrée et de sortie de chaque méthode sont liés en
tant qu'invocation de procédure distante selon le protocole SOAP 1.1.
-->
<binding name="PcspI01Service" type="tns:PcspI01Service">
  <soap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="Get">
    <soap:operation soapAction="Get" style="rpc"/>
    <input name="Get0In">
      <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding"/>
    </input>
    <output name="Get0Out">

```

```

        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </output>
</operation>
<operation name="Put">
    <soap:operation soapAction="Put" style="rpc" />
    <input name="Put1In">
        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </input>
    <output name="Put1Out">
        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </output>
    </operation>
<operation name="Delete">
    <soap:operation soapAction="Delete" style="rpc" />
    <input name="Delete2In">
        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </input>
    <output name="Delete2Out">
        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </output>
    </operation>
<operation name="CmdStatus">
    <soap:operation soapAction="CmdStatus" style="rpc" />
    <input name="CmdStatus3In">
        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </input>
    <output name="CmdStatus3Out">
        <soap:body use="encoded"
namespace="http://www.IPCablecom.com/pcsp/i01"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </output>
    </operation>
</binding>
<!--

```

Définition de niveau supérieur du service PCSP I01.

Il y a lieu de noter que l'élément <service> ne contient pas d'adresse. On suppose que l'adresse effective du service sera définie explicitement entre le client et le serveur.

```

-->
<service name="PcspI01Service">
    <documentation>IPCablecom CMS Subscriber Provisioning
Service I01</documentation>
    <port name="PcspI01Service" binding="tns:PcspI01Service">
        <soap:address location="" />
    </port>
</service>
</definitions>

```


Appendice I

Exemples de codage d'entités

Pour les exemples suivants on a supposé que le paramètre `<unique_qualified_namespace>` défini à l'Annexe A était positionné sur la valeur `"http://www.cablelabs.com/Pcsp/I01/schema"`.

I.1 Exemple d'objet PcspService

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.0.1 U (http://www.xmlspy.com) by Paul Duffy
(private) -->
<!--
  Exemple de codage d'objet Service.

  Valeur par défaut et espace de nommage "pcsp" mis à la valeur PcspI01.
  L'espace de nommage "pcsp" est une commodité, qui permet aux extensions
  vendeur de référencer les éléments du schéma PCSP principal..*

-->
<PcspService xmlns="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:pcsp="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="PcspI01.xsd">
  <!--
    Exemple d'objet Service.
  -->
  <ServiceId format="NSN">9785551212</ServiceId>
  <AdminStatus>1</AdminStatus>
  <BillingId>9785550000</BillingId>
  <ExternalId>0123456789</ExternalId>
  <IsPrimary>true</IsPrimary>
  <PrimaryRing>IndexIntoCadenceTable</PrimaryRing>
  <DisplayName>John Q Public</DisplayName>
  <DisplayNumber>(978) -555-1212</DisplayNumber>
  <Password>45hjg3j6gkg6h54j6gkj3g6</Password>
  <Announcements>
    <Language>EN</Language>
    <Timezone>+0500</Timezone>
  </Announcements>
  <InterExchange>
    <PIC>0123</PIC>
    <LPIC>0123</LPIC>
    <IPIC>0123</IPIC>
  </InterExchange>
  <LNP>
    <PortingStatus>0</PortingStatus>
    <LNPT>0</LNPT>
  </LNP>
  <ListOfCallFeatures>
    <CfCND>
      <Subscribed>true</Subscribed>
      <AdminStatus>1</AdminStatus>
    </CfCND>
    <CfCIDB>
      <Subscribed>0</Subscribed>
      <AdminStatus>1</AdminStatus>
      <UserParams>
        <Flag>PUBLIC</Flag>
      </UserParams>
    </CfCIDB>
  </ListOfCallFeatures>
</PcspService>
```

```

<CfCFBL>
  <Subscribed>true</Subscribed>
  <AdminStatus>1</AdminStatus>
  <UserParams>
    <Active>true</Active>
    <Always>0</Always>
    <ForwardTo>9785551212</ForwardTo>
  </UserParams>
</CfCFBL>
<CfSPCALL>
  <Subscribed>0</Subscribed>
  <AdminStatus>1</AdminStatus>
  <UserParams>
    <ListOfSpeedDial>
      <SdPair>
        <SdNum>1</SdNum>
        <ServiceId>9785551212</ServiceId>
      </SdPair>
      <SdPair>
        <SdNum>3</SdNum>
        <ServiceId>9785551000</ServiceId>
      </SdPair>
    </ListOfSpeedDial>
  </UserParams>
</CfSPCALL>
<CfRDA>
  <Subscribed>1</Subscribed>
  <AdminStatus>1</AdminStatus>
</CfRDA>
<CfLSR>
  <Subscribed>1</Subscribed>
  <AdminStatus>1</AdminStatus>
  <UserParams>
    <BlkDomLongDist>1</BlkDomLongDist>
    <BlkIntLongDist>1</BlkIntLongDist>
    <BlkPayPerCall>1</BlkPayPerCall>
    <BlkOperatorAssist>1</BlkOperatorAssist>
    <BlkDirAssist>1</BlkDirAssist>
    <BlkTollFree>1</BlkTollFree>
    <ListOfServiceId>
      <ServiceId>9895551001</ServiceId>
      <ServiceId>9895551002</ServiceId>
      <ServiceId>9895551003</ServiceId>
    </ListOfServiceId>
  </UserParams>
</CfLSR>
<CfDND>
  <Subscribed>1</Subscribed>
  <AdminStatus>1</AdminStatus>
  <UserParams>
    <Active>true</Active>
    <WdStartTod1>00:00:00+05:00</WdStartTod1>
    <WdStopTod1>06:00:00+05:00</WdStopTod1>
    <WdStartTod2>18:00:00+05:00</WdStartTod2>
    <WdStopTod2>20:00:00+05:00</WdStopTod2>
    <WeStartTod1>00:00:00+05:00</WeStartTod1>
    <WeStopTod1>09:00:00+05:00</WeStopTod1>
    <WeStartTod2>18:00:00+05:00</WeStartTod2>
    <WeStopTod2>20:00:00+05:00</WeStopTod2>
  </UserParams>
</CfDND>
<CfCOC>
  <Subscribed>1</Subscribed>
  <AdminStatus>1</AdminStatus>

```

```

    <UserParams>
      <Active>true</Active>
      <StartTod>00:00:00+05:00</StartTod>
      <StopTod>06:00:00+05:00</StopTod>
      <ListOfServiceId>
        <ServiceId>9895551001</ServiceId>
        <ServiceId>9895551002</ServiceId>
        <ServiceId>9895551003</ServiceId>
      </ListOfServiceId>
    </UserParams>
  </CfCOC>
  <CfNSA>
    <Subscribed>1</Subscribed>
    <AdminStatus>1</AdminStatus>
    <UserParams>
      <Active>true</Active>
      <StartTod>00:00:00+05:00</StartTod>
      <StopTod>06:00:00+05:00</StopTod>
      <ListOfServiceId>
        <ServiceId>9895551001</ServiceId>
        <ServiceId>9895551002</ServiceId>
        <ServiceId>9895551003</ServiceId>
      </ListOfServiceId>
    </UserParams>
  </CfNSA>
</ListOfCallFeatures>
</PcspService>

```

I.2 Exemple d'objet PcspEndpoint

```

<?xml version="1.0" encoding="UTF-8"?>
<PcspEndpoint xmlns="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:pcsp="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <!--
    Exemple d'objet Endpoint.
  -->
  <EndpointId>aaln/1@mta01.cablelabs.com</EndpointId>
  <AdminStatus>2</AdminStatus>
  <Protocol>MtaDefault</Protocol>
  <Codec>2</Codec>
</PcspEndpoint>

```

I.3 Exemple d'objet PcspMta

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.0.1 U (http://www.xmlspy.com) by Paul Duffy
(private) -->
<PcspMta xmlns="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:pcsp="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="PcspI01.xsd">
  <!--
    Exemple d'objet adaptateur MTA.
  -->
  <MtaFqdn>mta01.cablelabs.com</MtaFqdn>
  <ListenPort>2427</ListenPort>
  <CmtsFqdn>cmta01.cablelabs.com</CmtsFqdn>
  <Timezone>-0500</Timezone>
  <Protocol>MCGP 1.0 NCS 1.0</Protocol>
  <Codec>5</Codec>
</PcspMta>

```

I.4 Exemple d'objet PcspCms

```
<?xml version="1.0" encoding="UTF-8"?>
<PcspCms xmlns="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:pcsp="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <!--
    Objet CMS.

    Encore incomplètement défini ... Simplement sa clé.
    Constitue un ensemble de services et de points d'extrémité.
  -->
  <CmsFqdn>cma01.cablelabs.com</CmsFqdn>
</PcspCms>
```

I.5 Exemple de relation PcspRelation

```
<?xml version="1.0" encoding="UTF-8"?>
<PcspRelation xmlns="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.cablelabs.com/Pcsp/I01/schema PcspI01.xsd"
relOp="add">
  <!--
    relation PcspRelation.

    Cette relation associe plusieurs points d'extrémité au service
    "9785551212".
  -->
  <Class1>PcspService</Class1>
  <Key>9785551212</Key>
  <Class2>PcspEndpoint</Class2>
  <ListOfKeys>
    <Key>aaln/1@mta01.cablelabs.com</Key>
    <Key>aaln/1@mta02.cablelabs.com</Key>
    <Key>aaln/1@mta03.cablelabs.com</Key>
    <Key>aaln/1@mta04.cablelabs.com</Key>
  </ListOfKeys>
</PcspRelation>
```

Appendice II

Exemple d'extension d'objet

Pour les exemples suivants on a supposé que le paramètre `<unique_qualified_namespace>` défini à l'Annexe A était positionné sur la valeur "http://www.cablelabs.com/Pcsp/I01/schema".

II.1 Exemple d'objet étendu PcspService Object

L'exemple suivant illustre les capacités d'extension offertes par le schéma PCSP. L'exemple en question étend un objet PcspService au moyen d'une nouvelle fonctionnalité d'appel et de plusieurs nouveaux éléments ajoutés au corps principal de l'objet.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Exemple illustrant les modalités d'extension d'un objet Pcsp.
  Il étend l'objet PcspService par des champs et des fonctionnalités d'appel
  supplémentaires.
-->
```

Les indications détaillées figurent ci-dessous.

```
-->
<PcspService xmlns="http://www.cablelabs.com/Pcsp/I01/schema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:pcsp="http://www.cablelabs.com/Pcsp/I01/schema">
<!--
  Le corps principal de l'objet Service est rempli par des spécimens de
  données qui permettront de valider l'objet.
-->
<ServiceId>5551212</ServiceId>
<AdminStatus>0</AdminStatus>
<BillingId>5551212</BillingId>
<ExternalId>5551212</ExternalId>
<IsPrimary>>true</IsPrimary>
<PrimaryRing/>
<DisplayName/>
<DisplayNumber/>
<Password/>
<Announcements>
  <Language>EN</Language>
  <Timezone>+0500</Timezone>
</Announcements>
<InterExchange>
  <PIC>0</PIC>
  <LPIC>0</LPIC>
  <IPIC>0</IPIC>
</InterExchange>
<LNP>
  <PortingStatus>1</PortingStatus>
  <LNPT>>true</LNPT>
</LNP>
<!--
```

Il est possible d'étendre un objet Service en deux emplacements:

1. corps principal de l'objet.
2. liste des fonctionnalités d'appel.

Dans le cas présent nous étendons l'ensemble des fonctionnalités d'appel en ajoutant la fonctionnalité CfXYZ.

1. L'élément VendorExt doit spécifier un espace de nommage valide pour le schéma de l'extension. L'analyseur syntaxique peut alors repérer l'emplacement du fichier du schéma de l'extension.

2. Tout contenu de l'élément VendorExt doit être complet par rapport à l'espace de nommage, en vue de la validation par rapport au schéma.

```
-->
<ListOfCallFeatures>
  <Extension xmlns:ext="http://www.cablelabs.com/SampleExtension">
    <ext:CfXYZ>
      <ext:Subscribed>>true</ext:Subscribed>
      <ext:Enabled>>true</ext:Enabled>
    </ext:CfXYZ>
  </Extension>
</ListOfCallFeatures>
<!--
```

Ce cas consiste à étendre le contenu de données du corps principal de l'objet Service.

```
-->
<Extension xmlns:ext="http://www.cablelabs.com/SampleExtension">
  <ext:A>Sample extension A</ext:A>
  <ext:B>Sample extension B</ext:B>
  <ext:C>Sample extension C</ext:C>
</Extension>
</PcspService>
```

II.2 Schéma d'extension

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Schéma de l'exemple d'extension PcspService.

  Ce schéma définit plusieurs extensions:

  A, B, et C pour le corps principal de l'objet Service.
  Fonctionnalité d'appel CfXYZ pour la liste de fonctionnalités d'appel du
  service.
-->
<xs:schema targetNamespace="http://www.cablelabs.com/SampleExtension"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://www.cablelabs.com/SampleExtension" elementFormDefault="qualified">
  <xs:element name="A" type="xs:string"/>
  <xs:element name="B" type="xs:string"/>
  <xs:element name="C" type="xs:string"/>
  <xs:element name="CfXYZ">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Subscribed" type="xs:boolean"/>
        <xs:element name="Enabled" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Appendice III

Evaluation du codage des données

Options envisagées pour le codage des objets données et de la messagerie.

III.1 XML

XML est un métalangage standard qui permet aux entreprises d'élaborer leur propre langage de balisage pour la publication de documents et les échanges de données. Ces balisages figurent en mode texte: ils sont censés être évidents tant aux personnes qu'aux processus. Le langage XML offre les possibilités suivantes:

- échange de données ouvert, fondé sur des normes, et indépendant des plates-formes;
- analyseurs syntaxiques normalisés pour la mise de données en mémoire;
- interfaces normalisées (arborescentes et en continu) pour le traitement des données;
- modalités normalisées de présentation des données;
- modalités normalisées d'interrogation;
- modalités normalisées d'établissement de liaisons entre les données;
- formation normalisée du personnel aux tâches de publication et de traitement des données.

Coût: format de codage légèrement accru et frais d'analyse syntaxique plus importants.

La spécification XML est élaborée sous le contrôle du Groupe de travail XML du Consortium *www* (W3C, *world wide web consortium*). Des Groupes d'experts spéciaux de différentes disciplines y participent. Il s'agit d'une norme publique – aucune entreprise ne détient de droits exclusifs sur sa conception. La spécification v1.0 a été approuvée le 10 février 1998 par le

Consortium W3C en tant que Recommandation. Le texte de la spécification figure sur le site <http://www.w3.org/TR/REC-xml>.

III.2 ASN.1/Règles de codage fondamentales (BER)

La notation ASN.1 correspond à une syntaxe en mode texte indépendante des plates-formes, servant à représenter des structures de données arbitraires. Elle est couramment employée pour les représentations de bases de données de gestion suivant un protocole simple de gestion de réseau. Les règles de codage fondamentales sont constituées par un algorithme simplement récursif établissant un codage d'octets compact à partir d'une description ASN.1. Les règles de codage fondamentales codent chaque élément par une étiquette, indiquant de quel type il s'agit, une donnée de longueur indiquant la taille de l'objet et enfin une valeur qui renferme le contenu réel de l'objet.

Voir le Tableau III.1.

III.3 ASCII propriétaire

Les codages propriétaires ne relèvent pas de la présente Recommandation.

III.4 Protocole de description de session (SDP)

Ce protocole n'est pas souple en termes de contenu. Il sert essentiellement à la description des capacités de lecture du support en continu.

III.5 RADIUS

Le système RADIUS de codage de données (TLV) (type-longueur-valeur) est trop primitif et ne peut appliquer une mise en séquence. Ce langage est difficile et son application est limitée au codage des structures.

III.6 SQL

Langage associé à un système/une implémentation spécifique de base de données relationnelles. Certains vendeurs peuvent déjà avoir déployé des bases de données utilisant des schémas non compatibles.

III.7 Récapitulation des options

Tableau III.1/J.177 – Options de codage des données

Option	Avantages	Inconvénients
XML	Souple, utilisation de balises ASCII. Permet une vérification de la syntaxe grâce à l'utilisation d'un schéma/d'une fonction de définition de contenu de document. Facilité d'extension sans réalisation d'un transport. Autorise les extensions vendeur. Indépendance de la plate-forme. Indépendance du langage.	Absence de sécurisation, exige une couche Transport sécurisée. Utilisera une plus forte capacité d'unité centrale et de réseau par comparaison à un codage binaire. Temps d'analyse syntaxique, etc.

Tableau III.1/J.177 – Options de codage des données

Option	Avantages	Inconvénients
ASN.1/Règle de codage fondamental (BER)	<p>Structure hiérarchique de formatage des données. Définition d'un langage de description du format des données (ou "schéma").</p> <p>Possibilité d'imbrication des structures de données.</p> <p>Formatage des données indépendant d'une plate-forme.</p> <p>Possibilité d'extension du format SI la conception prévoit une possibilité de différentes versions du format de façon à ce que l'application puisse déterminer le format nécessaire à utiliser pour analyser le contenu des données.</p>	<p>La notation ASN.1 n'est pas facilement extensible.</p> <p>La compatibilité ascendante des versions de format risque d'être difficile à intégrer dans la conception et à implémenter.</p> <p>La plupart des implémentations utilisent pour chaque système des analyseurs compilés de niveau binaire, ce qui complique considérablement la définition d'applications souples.</p> <p>La mise au point des applications ainsi que leur interopérabilité risquent de s'avérer problématiques dans la mesure où une petite erreur de formatage peut empêcher l'analyse/la lecture d'un "paquet" de données.</p>
ASCII propriétaire	Ne relève pas de la présente Recommandation.	Ne relève pas de la présente Recommandation.
SDP (Description de session)		<p>Absence de souplesse en termes de contenu.</p> <p>Sert essentiellement à la description des capacités de lecture en continu des médias.</p>
RADIUS		<p>Le codage des données RADIUS (TLV) est trop rudimentaire.</p> <p>Impossibilité d'imposer une mise en séquence.</p> <p>Difficile et limité aux structures de codage.</p>
SQL		Associé à l'implémentation de systèmes de bases de données relationnelles – que certains vendeurs n'utilisent pas nécessairement.

III.8 Recommandation: XML

Le langage XML offre une forme de structuration des messages et de mise en paquets des données indépendante des plates-formes et technologiquement neutre. Il s'agit d'un choix excellent pour envoyer des données entre applications hétérogènes, sans que chaque application soit tenue de déterminer le format propriétaire de l'autre. Puisqu'il s'agit d'un langage structuré, le langage XML est bien adapté aux messages de type hiérarchique. Le mappage des données vers les éléments étant facile, un document XML (en tant que structure arborescente) assure la maintenance de la hiérarchie. Coûts: taille accrue des données utiles câblées et temps accru d'agencement des données (analyse syntaxique) pour les objets.

Appendice IV

Evaluation du protocole de transport

IV.1 Protocole TFTP offrant une sécurité Internet

L'infrastructure IPCablecom utilise d'ores et déjà le protocole TFTP (DOCSIS). Ce protocole est censé être utilisé en tant que protocole léger de transfert de fichier.

IV.2 Enregistrement RADIUS multiple par lots en demande simple via messages d'événement

RADIUS est une norme de l'IETF pour traiter les opérations d'authentification, d'autorisation et de comptabilisation de liaisons établies par Internet. RADIUS est actuellement la norme de facto employée par la plupart des constructeurs de routeurs pour les activités de ce type. Plusieurs vendeurs de passerelles téléphoniques IP utilisent d'ores et déjà RADIUS pour les extensions vendeur permettant d'acheminer l'information nécessaire à la facturation.

RADIUS définit conjointement un protocole de transport et une spécification de formats de message. En tant que protocole de transport RADIUS s'appuie sur le protocole datagramme d'utilisateur (UDP, *user datagram protocol*) pour l'envoi de messages et se réfère à des ports d'accès.

En ce qui concerne le format des messages, les données sont formatées en type-longueur-valeur (formatage appelé également attribut-longueur-valeur). Les étiquettes standard d'authentification, d'autorisation et de comptabilisation sont prédéfinies et sont rarement nécessaires. Toutefois, il est possible d'ajouter de nouveaux attributs sans modifier les implémentations existantes du protocole. Le langage RADIUS comporte une longueur totale de message minimale de 20 caractères et une longueur maximale de 4096 caractères. Les champs de données individuels prennent en charge 247 octets, par exemple un URL ou un nom de fichier de 247 caractères.

RADIUS comporte de très mauvaises caractéristiques de fiabilité et une capacité pratiquement inexistante de reprise en cas d'erreur, en particulier dans les nouvelles étiquettes (possibilité de définition limitée à 255 au total) (par comparaison aux 600 fonctionnalités existantes dans certains commutateurs RTPC de classe 5)

IV.3 Diameter

Diameter désigne une activité du groupe de travail de l'IETF, conçu pour présenter une compatibilité ascendante avec RADIUS. Diameter présente une capacité d'extension nettement plus importante, comporte des avantages accrus en termes de sécurité, tandis que sa conception vise à réduire au minimum les tâches de configuration. De plus, il prend très bien en charge les tâches d'authentification, d'autorisation et de comptabilisation interdomaines en accueillant divers types de systèmes de sécurité, par exemple des clés publiques, etc. Diameter prend en charge la reprise sur incident vers un serveur auxiliaire (il est conçu pour des environnements caractérisés par des exigences de faible niveau de défaillance (99,99+)). Voir le protocole de base Diameter IETF RFC [18].

RADIUS/DIAMETER n'assure pas les communications bidirectionnelles (uniquement les accusés de réception) et s'avère par conséquent non conforme aux prescriptions.

IV.4 Systèmes d'objets répartis

IV.4.1 CORBA/IIOP

Object Management Group ou le groupe Object Management Group se fait le champion de la technologie dite des objets répartis. Cette technologie est soutenue par les membres de l'OMG qui sont plus de 800.

L'architecture de courtier commun de requêtes d'objets (CORBA, *common object request broker architecture*) permet aux applications de communiquer entre elles au moyen d'un courtier de requêtes d'objets (ORB, *object request broker*). L'ORB est un logiciel médiateur qui établit des liens client-serveur entre objets. L'ORB enregistre les clients et gère divers types de droits notamment de publication, d'abonnement et de réception. Grâce à un courtier ORB, un client peut invoquer de façon transparente une méthode vis-à-vis d'un objet serveur, laquelle peut se trouver soit sur la même machine soit dans un réseau. Le courtier ORB intercepte l'appel local et se charge de trouver un objet capable d'implémenter la requête, de transmettre ses paramètres, d'invoquer sa méthode et de renvoyer les résultats.

Le langage de définition d'interface (IDL, *interface definition language*) sert à établir le contrat de protocole ORB entre les objets client et serveur. Le courtier ORB a essentiellement pour effet de masquer au programmeur les caractéristiques détaillées de transport. Le langage IDL est compilé en implémentations C++, Java, etc., de talons clients et serveurs, traitant toutes les tâches de codage/décodage de données requises par le protocole de transport IIOP utilisé entre clients et serveurs.

L'architecture CORBA traitera les aspects détaillés de la recherche du serveur pour une demande de méthode, du transport des arguments de la machine client à la machine serveur, et enfin du transport de tout code restitué en retour jusqu'à la machine client.

Nombre de vendeurs peuvent actuellement fournir des courtiers ORB pour plus de trois douzaines de plates-formes logicielles et de systèmes d'exploitation. L'architecture CORBA est particulièrement répandue sur les plates-formes Unix. Toutefois, la concurrence entre les vendeurs ORB porte sur les fonctionnalités. Des problèmes persistants d'interopérabilité apparaissent dès lors que l'on va au-delà des aspects élémentaires (sécurité, etc.). Ainsi deux courtiers ORB choisis au hasard ont une faible probabilité de pouvoir communiquer efficacement entre eux. Au niveau du développement, l'architecture CORBA tend à présenter une grande complexité. De plus, il s'agit d'une option relativement coûteuse (temps d'exécution et licences de développement).

IV.4.2 DCOM

Modèle d'objets composants répartis de Microsoft. Ce modèle a en commun les caractéristiques suivantes avec l'architecture CORBA:

- sépare l'interface objet de l'implémentation. Il utilise à cet effet le langage MIDL (variante Microsoft du langage IDL);
- transparence d'emplacement. Les serveurs client invoquent des méthodes disponibles sur des objets distants sans connaître la machine sur laquelle l'objet distant fonctionne;
- système uniforme de traitement des exceptions (la méthode DCOM intitulé un renvoi un simple statut de renvoi HRESULT).

Toutefois:

- le modèle DCOM s'appuie sur le protocole de transport DCE ORPC, qui est incompatible avec le transport IOP utilisé par l'architecture CORBA;
- le modèle DCOM est essentiellement une technologie strictement Microsoft. Il utilise de façon moins normalisée les plates-formes Win95, Win98 et Windows NT.

IV.5 HTTP

Le protocole de transfert hypertexte (HTTP, *hypertext transfer protocol*) est le protocole de base pour la toile mondiale. Il a été conçu pour relier des sources de données hétérogènes de façon à créer un système d'information réparti. Il a également été conçu dans une optique d'extensibilité. Une transaction type HTTP:

- 1) connexion au serveur établie par le client;
- 2) envoi par le client d'une requête au serveur (avec les paramètres URL);
- 3) envoi par le serveur d'une réponse contenant le statut et l'URL demandé;
- 4) possibilité de se déconnecter pour chaque extrémité.

Le protocole HTTP fournit des en-têtes de transaction pour les requêtes client et les réponses serveur. L'en-tête de transaction client peut contenir les paramètres utilisés pour faciliter l'acheminement des informations requises jusqu'au client (par exemple type de format de données, langage, etc.). La transaction serveur peut contenir des paramètres indiquant des informations concernant la réponse (par exemple statut de la requête (par exemple code retour), longueur des données envoyées, type de contenu, langage du contenu, etc.).

Compte tenu de l'état actuel de la toile, le protocole HTTP est omniprésent. Par ailleurs, il s'apprête bien à l'installation de pare-feu.

IV.6 Récapitulation des options

Tableau IV.1/J.177 – Options de transport

Option	Avantage	Inconvénient
Protocole TFTP avec sécurité Internet	Simplicité. Déjà implémentée pour DOCSIS.	Ne permet pas les communications bidirectionnelles.
RADIUS	Souple: comporte des champs propres au vendeur et propres à l'abonné. IP Cablecom a sans doute la possibilité de définir des champs dans cet espace. Utilisé par de nombreux systèmes de comptabilisation de téléphonie Internet. Déjà mis en place à grande échelle sur des composants Internet tels que les routeurs.	Tous les produits RADIUS ne prennent pas en charge les tâches d'authentification, d'autorisation et de comptabilisation. La couche Application doit traiter les problèmes de fiabilité. Sécurité intégrée inadéquate; il faut un protocole de confiance indépendant ou des clés de secret communes au routeurs périphériques.
Protocole propriétaire à points d'interliaison avec SSL		Propriétaires. Si nous poursuivons dans un sens, nous finirons sans doute par rédiger la plupart des programmes actuellement disponibles selon les protocoles SOAP ou XMLP.

Tableau IV.1/J./177 – Options de transport

Option	Avantage	Inconvénient
CORBA/IIOP	Facilité d'implémentation; l'infrastructure CORBA gère en détail les problèmes de résolution de noms, les paramètres de mise en paquet codés dans les messages et les caractéristiques de transport.	Constitue généralement une solution plus coûteuse. L'infrastructure CORBA doit soit être mise au point soit être achetée; dans un cas comme dans l'autre, sa mise en œuvre doit coïncider avec celle de l'application. Plusieurs questions restent à résoudre: en ce qui concerne notamment l'interopérabilité des différents produits CORBA/ORB.
DCOM	DCOM et CORBA/IIOP sont des technologies similaires.	Il s'agit essentiellement d'une option propre à Windows de Microsoft. Pas d'interfonctionnement avec CORBA-IIOP.
HTTP	Le protocole HTTP est déjà largement déployé en tant que protocole de transport sous-jacent pour les échanges entre différentes sources de données (serveurs Web) et les consommateurs de données (navigateurs client). La transmission de données s'appuie sur de simples transactions. Protocole simple, sans états, rédigé en texte ASCII. Simplifie les échanges de données en faisant appel aux infrastructures de réseau les plus couramment déployées (passerelles, etc.). Possibilité pour le client d'utiliser des appels simples de méthode pour les requêtes, par exemple GET, POST, HEAD, PUT, DELETE, LINK, et UNLINK.	Etant sans états, le protocole n'a pas de mémoire de la transaction une fois celle-ci menée à bien. Permet-il d'assurer le rythme requis de transaction du serveur CMS? Relativement coûteux en termes de largeur de bande et de besoins de traitement.

IV.7 Recommandation: HTTP 1.1

Appendice V

Bibliographie

- [12] *Basic Residential Feature Descriptions for IPCablecom-Based VOIP Services*, PKT-TR-VoIPBRF-R01-000608, Cable Television Laboratories, Inc., 8 juin 2000.
- [13] *Enhanced Residential Feature Descriptions for IPCablecom-Based VOIP Services*, PKT-TR-VoIPERF-R01-000831, Cable Television Laboratories, Inc., 31 août 2000.
- [14] *IPCablecom Electronic Surveillance Call Flows Technical Report*, PKT_TR-ESCF-V01-991229, Cable Television Laboratories, Inc., 29 décembre 1999.

- [15] IETF RFC 2139 (1997), *RADIUS Accounting*.
- [16] COMMON LANGUAGE® General Codes, *Telecommunications Service Providers IAC Codes, Exchange Carrier Names, Company Codes – Telcordia and Region Number: BR-751-100-112*. Issue 4: avril 2002.
- [17] *Telcordia Local Exchange Routing Guide (LERG)*, Telcordia Technologies, Inc.
- [18] IETF Internet Draft, *Diameter Base Protocol*, <http://www.ietf.org/internet-drafts/draft-ietf-aaa-diameter-17.txt>.

SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série B	Moyens d'expression: définitions, symboles, classification
Série C	Statistiques générales des télécommunications
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	Systèmes audiovisuels et multimédias
Série I	Réseau numérique à intégration de services
Série J	Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	RGT et maintenance des réseaux: systèmes de transmission, circuits téléphoniques, télégraphie, télécopie et circuits loués internationaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	Réseaux de données et communication entre systèmes ouverts
Série Y	Infrastructure mondiale de l'information et protocole Internet
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication

