

Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

H.248.9

(01/2005)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET
MULTIMÉDIAS

Infrastructure des services audiovisuels – Procédures de
communication

**Protocole de commande de passerelle:
paquetages de serveur média évolué**

Recommandation UIT-T H.248.9

RECOMMANDATIONS UIT-T DE LA SÉRIE H
SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

CARACTÉRISTIQUES DES SYSTÈMES VISIOPHONIQUES	H.100–H.199
INFRASTRUCTURE DES SERVICES AUDIOVISUELS	
Généralités	H.200–H.219
Multiplexage et synchronisation en transmission	H.220–H.229
Aspects système	H.230–H.239
Procédures de communication	H.240–H.259
Codage des images vidéo animées	H.260–H.279
Aspects liés aux systèmes	H.280–H.299
Systèmes et équipements terminaux pour les services audiovisuels	H.300–H.349
Architecture des services d'annuaire pour les services audiovisuels et multimédias	H.350–H.359
Architecture de la qualité de service pour les services audiovisuels et multimédias	H.360–H.369
Services complémentaires en multimédia	H.450–H.499
PROCÉDURES DE MOBILITÉ ET DE COLLABORATION	
Aperçu général de la mobilité et de la collaboration, définitions, protocoles et procédures	H.500–H.509
Mobilité pour les systèmes et services multimédias de la série H	H.510–H.519
Applications et services de collaboration multimédia mobile	H.520–H.529
Sécurité pour les systèmes et services multimédias mobiles	H.530–H.539
Sécurité pour les applications et services de collaboration multimédia mobile	H.540–H.549
Procédures d'interfonctionnement de la mobilité	H.550–H.559
Procédures d'interfonctionnement de collaboration multimédia mobile	H.560–H.569
SERVICES À LARGE BANDE ET MULTIMÉDIAS TRI-SERVICES	
Services multimédias à large bande sur VDSL	H.610–H.619

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

Recommandation UIT-T H.248.9

Protocole de commande de passerelle: paquetages de serveur média évolué

Résumé

La présente Recommandation contient un ensemble de paquetages syntaxiques et un ensemble de paquetages fonctionnels. Les paquetages syntaxiques permettent de spécifier des annonces avec un contenu variable, le degré de souplesse n'étant limité que par la mise à disposition de la passerelle média (MG, *media gateway*) et du contrôleur de passerelle média (MGC, *media gateway controller*). Cette syntaxe peut en principe servir à spécifier des annonces multimédias, même si l'application qui est mentionnée dans la présente Recommandation concerne un contenu audio. Les paquetages fonctionnels prennent en charge la commande perfectionnée d'une fonction de ressource audio sur la base du protocole H.248.1. Les paquetages permettent de passer des annonces enregistrées avec un contenu variable ainsi que de collecter des chiffres ou des séquences audio enregistrées sur invite. Un autre paquetage permet à la passerelle média de gérer des segments médias enregistrés.

Les modifications dont a fait l'objet la Révision 1 de la présente Recommandation H.248.9 sont les suivantes:

- 6.3.6.11 Introduction d'un nouveau type de variable "tonalité" permettant de spécifier dynamiquement des segments audio.
- 6.4 Extension des ensembles dans la syntaxe de base: introduction d'un nouveau sélecteur pour les attributs de texte.
- 6.5.5.1 Type de variable "phrase": introduction de sous-types.
- 9.3.1 Signal PlayCollect: amélioration de la fonctionnalité, introduction de nouveaux paramètres.

NOTE – La présente Recommandation a été renumérotée en 2002. Il s'agit de l'ancienne Annexe M.1 de la Rec. UIT-T H.248.

Source

La Recommandation UIT-T H.248.9 a été approuvée le 8 janvier 2005 par la Commission d'études 16 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8.

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 2005

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
2	Références.....	2
	2.1 Références normatives.....	2
	2.2 Références informatives	2
3	Définitions	2
4	Abréviations et acronymes	3
5	Philosophie de conception	4
6	Syntaxe de spécification d'annonce	4
	6.1 Concepts syntaxiques: segments audio, variables et variables imbriquées....	4
	6.2 Paquetage de syntaxe d'annonce de base.....	9
	6.3 Paquetage de syntaxe de variable vocale.....	12
	6.4 Ensemble: extension de la syntaxe de base	18
	6.5 Type de variable texte générique: extension de la syntaxe de base	20
	6.6 Exemples	21
7	Nouveaux codes d'erreur H.248.1.....	23
	7.1 Syntaxe illégale dans une spécification d'annonce.....	23
	7.2 Type de variable non pris en charge.....	24
	7.3 Valeur de variable hors intervalle.....	24
	7.4 Catégorie non prise en charge	24
	7.5 Type de sélecteur non pris en charge	24
	7.6 Valeur de sélecteur non prise en charge	25
	7.7 Identificateur de segment inconnu.....	25
	7.8 Discordance entre la spécification de lecture et les données mises à disposition.....	25
	7.9 Erreur de mise à disposition	25
	7.10 Décalage non valable.....	26
	7.11 Aucun identificateur de segment libre.....	26
	7.12 Segment temporaire non trouvé.....	26
	7.13 Segment en cours d'utilisation	26
8	Paquetage de base de serveur audio évolué.....	27
	8.1 Propriétés.....	27
	8.2 Événements.....	27
	8.3 Signaux	28
	8.4 Statistiques.....	30
	8.5 Procédure.....	30
9	Paquetage de collecte de chiffres de serveur AAS	30
	9.1 Propriétés.....	30
	9.2 Événements.....	31

	Page
9.3 Signaux	32
9.4 Statistiques.....	41
9.5 Procédures	41
10 Paquetage d'enregistrement de serveur AAS.....	44
10.1 Propriétés.....	44
10.2 Evénements.....	45
10.3 Signaux	47
10.4 Statistiques.....	53
10.5 Procédures	53
11 Paquetage de gestion de segments de serveur audio évolué.....	54
11.1 Propriétés.....	54
11.2 Evénements.....	55
11.3 Signaux	55
11.4 Statistiques.....	57
11.5 Procédures	57

Recommandation UIT-T H.248.9

Protocole de commande de passerelle: paquetages de serveur média évolué

1 Domaine d'application

La présente Recommandation utilise le mécanisme de paquetage pour définir une syntaxe avec paramètres permettant de se référer à des annonces mises à disposition et de passer un contenu variable dans ces annonces. Comme indiqué dans la description des paquetages concernés, cette syntaxe contient des fonctionnalités facultatives, dont la prise en charge est indiquée par la présence de paquetages additionnels au niveau de la terminaison. La syntaxe est potentiellement utilisable pour un contenu multimédia, mais pour la plus grande partie, ce sujet nécessite un complément d'étude. Par ailleurs, on ajoute dans la présente Recommandation une série de paquetages fonctionnels au protocole Megaco/H.248.1 afin de commander une fonction de ressource audio qui peut résider dans une passerelle média ou dans un serveur audio spécialisé.

La syntaxe de spécification d'annonce est décrite dans une série de paquetages:

- paquetage de syntaxe de base: prend en charge la syntaxe permettant de se référer aux segments médias mis à disposition, avec une capacité générale d'extension. On trouvera une introduction au § 6.1 et une définition détaillée au § 6.2;
- paquetage de variables vocales: extension facultative de la syntaxe de base prenant en charge des variables autonomes et des variables imbriquées, avec un ensemble initial de types de variable vocale. On trouvera une introduction au § 6.1.4 et une définition détaillée au § 6.3;
- paquetage de syntaxe d'ensemble: extension facultative de la syntaxe de base prenant en charge un nombre arbitraire de qualificatifs définis par l'utilisateur à utiliser pour la résolution de structures audio complexes. Par exemple, l'utilisateur peut définir des qualificatifs pour tout ou partie des caractéristiques suivantes: langue, accent, format de fichier audio, sexe, locuteur, client. On trouvera une introduction au § 6.1.5 et une définition détaillée au § 6.4;
- paquetage de syntaxe de texte générique: extension facultative de la syntaxe de base prenant en charge le type de variable vocale suivant: texte générique. On trouvera une introduction au § 6.1.6 et une définition détaillée au § 6.5.

Les paquetages fonctionnels décrits dans la présente Recommandation sont les suivants:

- paquetage de base de serveur audio évolué (AAS, *advanced audio server*): prend en charge un signal permettant de passer une annonce et des événements permettant de contrôler le résultat de la demande de lecture. Voir le § 8;
- paquetage de collecte de chiffres de serveur AAS: étend le paquetage de base de serveur AAS en prenant en charge un signal et des événements permettant de coordonner la collecte de chiffres avec la lecture d'invités. Voir le § 9;
- paquetage d'enregistrement de serveur AAS: étend le paquetage de base de serveur AAS en prenant en charge une propriété, des signaux et des événements permettant de coordonner la collecte de séquences vocales enregistrées avec la lecture d'invités. Voir le § 10;
- paquetage de gestion de segments de serveur AAS: permet au contrôleur MGC de spécifier un autre segment audio qui est passé à la place d'un segment donné chaque fois que ce segment est invoqué, jusqu'à ce que le contrôleur MGC mette fin à ce remplacement. Il permet aussi de supprimer des segments persistants. A la différence des autres paquetages, ce paquetage est défini au niveau d'une terminaison spéciale de commande de segment logique et utilise uniquement la syntaxe de spécification d'annonce de base. Voir le § 11.

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 sont 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.

- Recommandation UIT-T H.248.1 (2002), *Protocole de commande de passerelle: version 2*, tel qu'amendée par le Corrigendum 1 (03/2004).
- Recommandation UIT-T Q.1218 (1995), *Recommandation relative à l'interface pour l'ensemble de capacités 1 du réseau intelligent*.
- IETF RFC 1738 (1994), *Uniform Resource Locators (URL)*.
- IETF RFC 2396 (1998), *Uniform Resource Identifiers (URI): Generic Syntax*.
- IETF RFC 2616 (1999), *Hypertext Transfer Protocol – HTTP/1.1*.
- IETF RFC 3066 (2001), *Tags for the Identification of Languages*.
- ISO 639-1:2002, *Codes pour la représentation des noms de langue – Partie 1: Code alpha 2*.
- ISO 639-2:1988, *Codes pour la représentation des noms de langue – Partie 2: Code alpha 3*.
- ISO 3166-1:1997, *Codes pour la représentation des noms de pays et de leurs subdivisions – Partie 1: Codes pays*.
- ISO 3166-2:1998, *Codes pour la représentation des noms de pays et de leurs subdivisions – Partie 2: Code pour les subdivisions de pays*.
- ISO 4217:2001, *Codes pour la représentation des monnaies et types de fonds*.
- ISO 8601:2000, *Éléments de données et formats d'échange – Échange d'information – Représentation de la date et de l'heure*.
- ISO/CEI 10646-1:2000 et Amendements, *Technologies de l'information – Jeu universel de caractères codés sur plusieurs octets (JUC) – Partie 1: Architecture et plan multilingue de base*.

2.2 Références informatives

- IETF RFC 2279 (1998), *UTF-8, a transformation format of ISO 10646*.
- IETF RFC 2326 (1998), *Real Time Streaming Protocol (RTSP)*.
- IETF RFC 2805 (2000), *Media Gateway Control Protocol Architecture and Requirements*.

3 Définitions

La présente Recommandation définit les termes suivants:

3.1 segment audio: unité de contenu audio pouvant être spécifiée séparément. Ce concept peut être généralisé par le concept de *segment média*, avec contenu multimédia général.

3.2 spécification de segment: ensemble des informations que le contrôleur doit fournir pour invoquer la lecture d'un segment audio. Les spécifications de segment de la forme définie dans la présente Recommandation sont potentiellement utilisables aussi pour invoquer la lecture de contenu multimédia, mais les détails nécessitent un complément d'étude.

3.3 variable vocale: unité de contenu audio présentant l'un des types et éventuellement un sous-type tels que définis dans la présente Recommandation, son contenu réel étant donné dans le cadre de la spécification de segment. Comme la valeur d'une variable vocale est spécifiée par un texte, une variable vocale peut aussi être assimilée à une variable textuelle si le moyen d'expression est le texte.

3.4 variable autonome: segment audio dont la spécification décrit une seule instance de variable vocale.

3.5 segment mis à disposition: segment audio pouvant être extrait grâce à un simple identificateur ou à un identificateur URI, qui doit faire partie de la spécification de segment. Un segment mis à disposition peut inclure des variables vocales. Le contenu de ces variables doit aussi figurer dans la spécification de segment, mais pas leur type/sous-type.

3.6 ensemble de segments: ensemble de diverses formes d'expression (par exemple langues différentes, locuteurs différents) du même contenu sémantique d'un segment audio. Pour indiquer, dans la spécification de segment, le choix de la forme d'expression à utiliser dans une instance donnée de segment audio, on donne une valeur au sélecteur associé à l'ensemble. Un segment audio donné pouvant être associé à plusieurs ensembles, plusieurs sélecteurs doivent figurer dans la spécification de segment pour définir une instance unique.

3.7 sélecteur: paramètre associé à un ensemble, ayant une plage de valeurs prédéfinies, correspondant aux membres de l'ensemble. Les ensembles, les sélecteurs et les plages possibles de valeurs de sélecteur (et les valeurs par défaut) sont définis par la mise à disposition dans la passerelle média et dans les dispositifs supports.

3.8 annonce: résultat audible de la lecture d'une séquence de segments audio. La production d'annonces multimédias nécessite un complément d'étude.

4 Abréviations et acronymes

La présente Recommandation utilise les abréviations suivantes:

AAS	serveur audio évolué (<i>advanced audio server</i>)
ABNF	formalisme de Backus-Naur étendu (<i>augmented Backus-Naur form</i>)
ASN.1	notation de syntaxe abstraite numéro un (<i>abstract syntax notation one</i>)
BER	règles de codage de base (<i>basic encoding rules</i>)
BR	bref (<i>brief</i>) (type de signal utilisé dans la Rec. UIT-T H.248.1)
FTP	protocole de transfert de fichiers (<i>file transfer protocol</i>)
HTTP	protocole de transfert hypertexte (<i>hypertext transfer protocol</i>)
INAP	partie d'application du réseau intelligent (<i>intelligent network application part</i>)
MG	passerelle média (<i>media gateway</i>)
MGC	contrôleur de passerelle média (<i>media gateway controller</i>)
OO	commuté (<i>on/off</i>) (type de signal utilisé dans la Rec. UIT-T H.248.1)
RTSP	protocole d'écoulement en temps réel (<i>real time streaming protocol</i>)
TO	temporisé (<i>timeout</i>) (type de signal utilisé dans la Rec. UIT-T H.248.1)

UCS	jeu de caractères universel (<i>universal character set</i>)
URI	identificateur de ressource universel (<i>universal resource identifier</i>)
URL	localisateur universel de ressources (<i>universal resource locator</i>)
UTF	format de transformation UCS (<i>UCS transformation format</i>)

5 Philosophie de conception

Les paquetages syntaxiques décrits dans la présente Recommandation constituent un dispositif formel permettant au contrôleur MGC de déterminer le niveau de capacité de la passerelle média à traiter certaines constructions de la syntaxe de spécification d'annonce au niveau d'une terminaison donnée. Le contrôleur MGC acquiert les informations voulues par audit. La présence d'un paquetage syntaxique donné indique la capacité à traiter la syntaxe décrite dans la section procédures de ce paquetage. Les paquetages syntaxiques contiennent uniquement ces procédures.

Les paquetages fonctionnels décrits dans la présente Recommandation prennent en charge des capacités importantes, la plupart étant commandées via des paramètres de protocole. La majorité des paramètres sont facultatifs et peuvent généralement être omis, au profit de leurs valeurs par défaut. Une application audio qui invoque des références à des structures audio complexes mises à disposition peut spécifier des événements audio avec un minimum de syntaxe en tirant parti du caractère facultatif de certains paramètres et des valeurs par défaut des paramètres.

Les opérations dont il est question dans la présente Recommandation sont invoquées sous forme de signaux au niveau d'une terminaison associée à l'utilisateur (à l'exception des opérations de remplacement d'annonce, qui correspondent à des signaux invoqués au niveau d'une terminaison spéciale de commande de segment logique). Ce mécanisme de base présente des aspects auxquels il faut faire attention lors de l'utilisation de l'opération de lecture et de collecte de chiffres et de l'opération de lecture et d'enregistrement. Le principal souci est d'éviter toute interruption involontaire de ces opérations due à la reconnaissance d'événements au niveau de la terminaison. Le contrôleur MGC doit s'assurer que le drapeau KeepActive (maintien à l'état actif) est levé pour les événements qu'il active et qui ne visent pas à arrêter ces opérations.

Les règles usuelles pour le remplacement d'un descripteur de signal s'appliquent aux signaux associés aux paquetages fonctionnels et décrits dans la présente Recommandation. Autrement dit, si le descripteur de signal qui a invoqué une opération donnée est remplacé, l'opération continuera sans interruption uniquement si elle est invoquée de manière identique dans le nouveau descripteur de signaux, le drapeau KeepActive du signal étant levé.

L'opération de lecture et de collecte de chiffres utilise le descripteur de script de numérotation H.248.1 pour indiquer la séquence attendue de chiffres à collecter. Toutefois, l'interaction requise avec l'utilisateur en cas d'échec de la collecte de la séquence attendue lors de la première tentative empêche d'utiliser le mécanisme complet de script de numérotation intégré à la Rec. UIT-T H.248.1. En revanche, il faut spécifier le comportement modifié et fournir les résultats dans un événement d'achèvement d'opération et non dans un événement d'achèvement de script de numérotation.

6 Syntaxe de spécification d'annonce

6.1 Concepts syntaxiques: segments audio, variables et variables imbriquées

Tous les paquetages de la présente Recommandation reposent sur l'utilisation d'une syntaxe spéciale avec paramètres pour la description des annonces à passer. Cette syntaxe permet de décrire les annonces sous la forme d'une série de segments audio, chacun ayant été mis à disposition en un certain endroit physique ou étant spécifié dynamiquement par la description d'annonce proprement dite (sous la forme d'une variable vocale autonome).

Le paquetage de syntaxe d'annonce de base prend en charge aussi bien les structures audio simples que les structures audio complexes. Une structure audio simple pourrait être une annonce unique telle que "Bienvenu dans le service d'assistance automatique à l'annuaire". Une structure audio plus complexe pourrait consister en une annonce suivie par une variable vocale suivie par une autre annonce, par exemple "Il reste trente sept minutes sur votre carte d'appel à prépaiement", où "Il reste" est une annonce, le nombre de minutes est une variable vocale et "minutes sur votre carte d'appel à prépaiement" est une autre annonce.

Il existe deux méthodes de spécification d'une structure audio complexe. La première consiste à faire référence directement aux différentes composantes. Pour cela, il faut spécifier la description complète de chaque composante via le protocole. La seconde méthode consiste à mettre à disposition les composantes sur le serveur audio sous forme d'une seule entité à exporter une référence à cette entité vers l'agent d'appel. Dans ce cas, seule la référence (plus les éventuelles données dynamiques requises, comme les données relatives aux variables) est transmise via le protocole et il n'est pas nécessaire de spécifier les différentes composantes. La syntaxe de spécification de segment audio prend en charge les deux méthodes.

La syntaxe décrite dans la présente Recommandation comprend trois composantes: la syntaxe de base qui doit être prise en charge par toutes les implémentations des paquetages de la présente Recommandation, la syntaxe prenant en charge l'utilisation "d'ensembles" pour qualifier la lecture d'annonces et une syntaxe prenant en charge des variables de type texte arbitraire. Les capacités au-delà de la syntaxe de base sont facultatives; leur prise en charge est indiquée par la présence des paquetages correspondants au niveau de la terminaison où la lecture est invoquée.

6.1.1 Segments audio mis à disposition

Il est possible qu'une référence unique à un segment audio mis à disposition permette d'invoquer une structure audio complexe, y compris des variables dont les valeurs sont à spécifier au moment de l'invocation. La syntaxe permet au contrôleur MGC de spécifier les valeurs de ces variables imbriquées. A cette exception près, la différence entre les segments audio simples mis à disposition et les segments audio complexes mis à disposition est invisible pour le contrôleur MGC et ne concerne pas le protocole.

La syntaxe utilise des identificateurs (URI, *universal resource identifier*) pour désigner les segments mis à disposition, qui peuvent alors être situés physiquement soit au niveau de la passerelle média soit au niveau d'un autre dispositif, sans affecter le flux de messages entre le contrôleur MGC et la passerelle média. A chaque segment mis à disposition, on attribue un identificateur URI unique, qui, entre autres, peut être un nom hiérarchique ou simplement un nom ou un numéro.

6.1.2 Segments audio spécifiés dynamiquement

Un segment audio spécifié dynamiquement est un segment audio spécifié par une variable vocale autonome. Pour plus d'informations sur les variables, on se reportera au § 6.1.4.

6.1.3 Identificateurs de segment

Les segments mis à disposition et les segments enregistrés au moment de l'exécution sont identifiés par des identificateurs URI tels que définis dans l'IETF RFC 2396.

Un identificateur URI peut être un nom simple ou une adresse URL. Trois types d'adresses URL sont autorisées: file:, ftp: et http:. Les adresses file: sont utilisées pour les segments audio locaux au niveau du serveur audio. Les adresses ftp: sont utilisées pour les segments audio distants par rapport au serveur audio. Les adresses http: peuvent être utilisées pour les segments audio locaux au niveau du serveur audio avec la convention http://localhost ou pour les segments audio distants par rapport au serveur audio. Toutes les références audio qui nécessitent des paramètres codés dans

l'adresse URL doivent utiliser une adresse de type http:. Les exemples qui suivent montrent quelques possibilités. On trouvera d'autres exemples au § 6.6.

NOTE – Concernant la lecture de médias plus généraux sur IP, il convient d'envisager aussi l'utilisation du type d'adresse rtsp:. Les spécifications d'annonce multimédia nécessitent un complément d'étude.

Référence à des segments audio locaux (nom simple): 12354
 Référence à des segments audio locaux (fichier non hiérarchique): file://welcome
 Référence à des segments audio locaux: file://audio/xyztel/welcome
 Référence à des segments audio distants: http://audio/xyztel/welcome

6.1.4 Variables

Une variable vocale représente un seul concept sémantique (par exemple une date ou un numéro) et produit dynamiquement la parole appropriée sur la base des informations fournies au moment de l'exécution. Par exemple, si une application doit passer une date, plutôt que de dire au serveur audio de passer les différentes composantes de la date (par exemple "vingt" "deux" "mars" "mille" "neuf cent" "quatre-vingt-dix" "neuf"), le contrôleur MGC peut spécifier une variable vocale de type date avec la valeur "19990322". Ensuite, le serveur audio assemble et passe les composantes audio nécessaires pour donner la date.

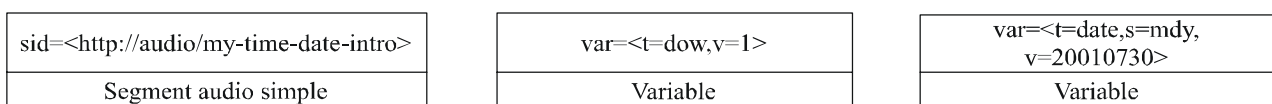
Les variables sont spécifiées par les paramètres suivants: type, sous-type et valeur. Les types de variable peuvent notamment être les suivants: date, argent, numéro, heure, etc. Le sous-type permet de préciser le type. Par exemple, le type de variable argent peut avoir comme sous-types associés: dollar, roupie, dinar, etc. Les variables ne nécessitent pas toutes un sous-type, auquel cas le paramètre de sous-type doit être mis à néant.

Comme décrit plus haut, la syntaxe d'annonce de serveur AAS prend en charge deux types de variables: autonomes et imbriquées. Les variables autonomes sont des variables qui ne font pas partie d'un segment audio mis à disposition. Leur type, sous-type et valeur doivent être spécifiés complètement par le contrôleur MGC. Cette spécification constitue un segment audio spécifié dynamiquement comme décrit plus haut.

Les variables imbriquées sont des variables qui sont mises à disposition dans le cadre d'un segment audio mis à disposition. Au moment de l'exécution, le contrôleur MGC fait référence au segment et spécifie une valeur pour chaque variable imbriquée dans ce segment. Si un segment comprend plusieurs variables imbriquées, les valeurs doivent être données dans l'ordre dans lequel les variables sont rencontrées lors de la lecture du segment.

6.1.4.1 Exemple d'utilisation de variables dans une séquence

Dans l'exemple qui suit, la séquence à passer est la suivante: "Aujourd'hui, nous sommes <jour de la semaine> <date>". Cette séquence est constituée de trois segments: un segment audio simple, une variable de type jour de la semaine et une autre variable de type date. La séquence peut être implémentée de deux façons: sous la forme d'une séquence spécifiée explicitement par le contrôleur MGC ou sous la forme d'un seul segment audio mis à disposition avec deux variables imbriquées. Ces deux possibilités sont illustrées.



H.248.9_F1

Figure 1/H.248.9 – Séquence explicite avec trois segments audio

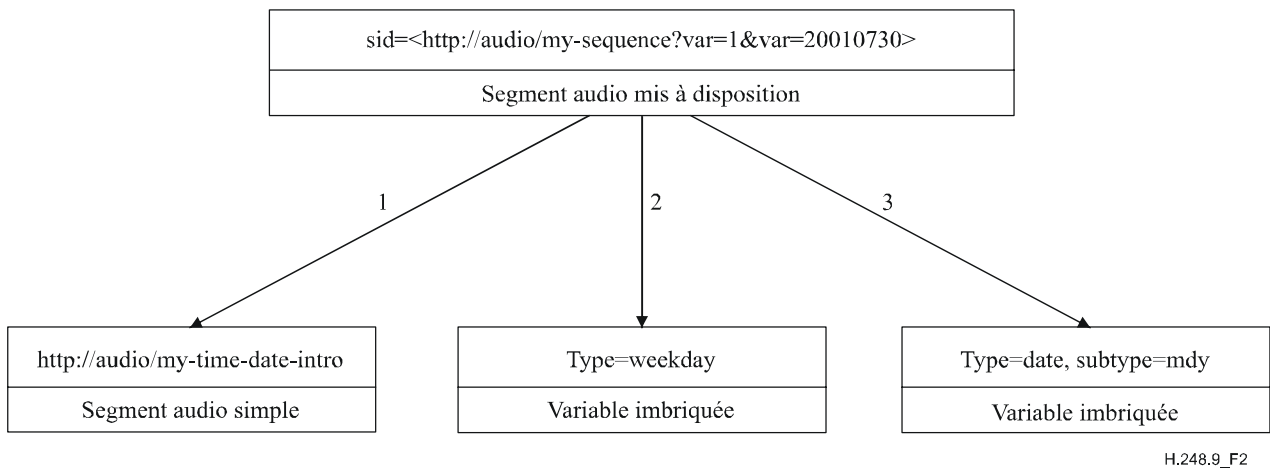


Figure 2/H.248.9 – Segment audio mis à disposition avec deux variables imbriquées

Dans les deux cas, le fournisseur a installé un segment audio simple désigné par `http://audio/my-time-date-intro`. Dans le premier cas, le contrôleur MGC peut voir ce segment. Dans le second cas, il a uniquement connaissance du segment mis à disposition `http://audio/my-sequence`, qui contient une variable imbriquée de jour de la semaine et une variable imbriquée de date, apparaissant dans cet ordre. Le fait que `http://audio/my-sequence` fasse référence à `http://audio/my-time-date-intro` n'est connu que du dispositif qui résout `http://audio/my-sequence`.

6.1.5 Ensembles de segments

Les ensembles constituent une caractéristique facultative améliorée de la syntaxe de spécification d'annonce. Un ensemble regroupe différents segments audio mis à disposition interchangeables et un sélecteur associé. A chaque ensemble est attribué un identificateur URI unique. Au moment de l'exécution, la valeur du sélecteur sert à déterminer l'élément de l'ensemble à passer. Dans une spécification d'annonce, un ensemble apparaît comme un seul segment audio mis à disposition avec sa ou ses valeurs de sélecteur.

Les différents types de sélecteur ne sont pas définis dans la syntaxe (sauf pour le sélecteur de langue prédéfini) mais sont définis en revanche par le fournisseur. Un fournisseur peut par exemple définir un ou plusieurs des types de sélecteur suivants: langue, accent, sexe, accent, client et/ou jour de la semaine. Pour chaque type de sélecteur, le fournisseur doit définir une plage de valeurs valables. Il peut aussi choisir de définir une valeur par défaut. Si une valeur de sélecteur n'est pas fournie au moment de l'exécution, c'est la valeur par défaut qui est utilisée.

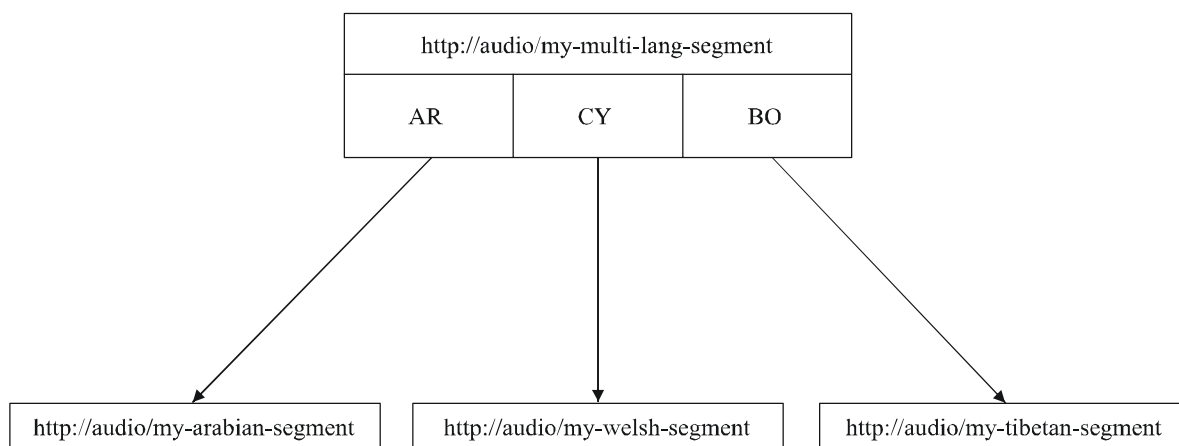
Les ensembles multidimensionnels sont autorisés. Ils prennent en charge un vecteur de types de sélecteur. Une valeur doit être spécifiée pour chaque type de sélecteur afin de déterminer une instance particulière du segment audio concerné.

Un ensemble peut contenir des variables imbriquées. Le type et l'ordre de ces variables doivent être identiques pour tous les membres de l'ensemble. La lecture d'une variable imbriquée doit être cohérente avec la valeur du sélecteur utilisé pour invoquer le segment audio dans lequel elle est imbriquée. Ainsi, par exemple, l'invocation d'un segment audio mis à disposition associé à un sélecteur de langue et contenant une variable de date imbriquée doit se traduire par la lecture de la valeur de date fournie dans l'invocation, dans la langue indiquée par la valeur du sélecteur. Comme pour les autres segments contenant des variables, si un ensemble comprend plusieurs variables imbriquées, les valeurs des variables doivent être spécifiées dans l'ordre dans lequel les variables sont rencontrées lors de la lecture du segment. Les ensembles pour lesquels les variables doivent être passées dans différents ordres suivant la valeur du sélecteur ne sont pas pris en charge.

6.1.5.1 Exemple d'ensemble

La Figure 3 illustre un exemple d'ensemble. Pour prendre en charge une application passant une séquence audio particulière en arabe, en gallois ou en tibétain, un fournisseur pourrait définir un ensemble avec le sélecteur prédéfini "lang" et définir trois valeurs possibles pour ce sélecteur: "ar", "cy" et "bo". Il mettrait à disposition trois segments audio, un dans chaque langue, et associerait le segment arabe à la valeur de sélecteur "ar", etc. Il pourrait aussi définir une valeur par défaut pour le sélecteur à utiliser lorsque aucune valeur n'est fournie, "ar" par exemple. On attribuerait à l'ensemble tout entier un identificateur URI unique, qui serait le seul identificateur URI que le contrôleur MGC pourrait voir.

Au moment de l'exécution, une référence à l'ensemble avec la valeur de sélecteur "cy" se traduirait par la lecture de la version galloise de l'annonce. Une référence à l'ensemble sans valeur de sélecteur se traduirait par la lecture de la version arabe de l'annonce étant donné que l'arabe a été défini comme valeur par défaut du sélecteur.



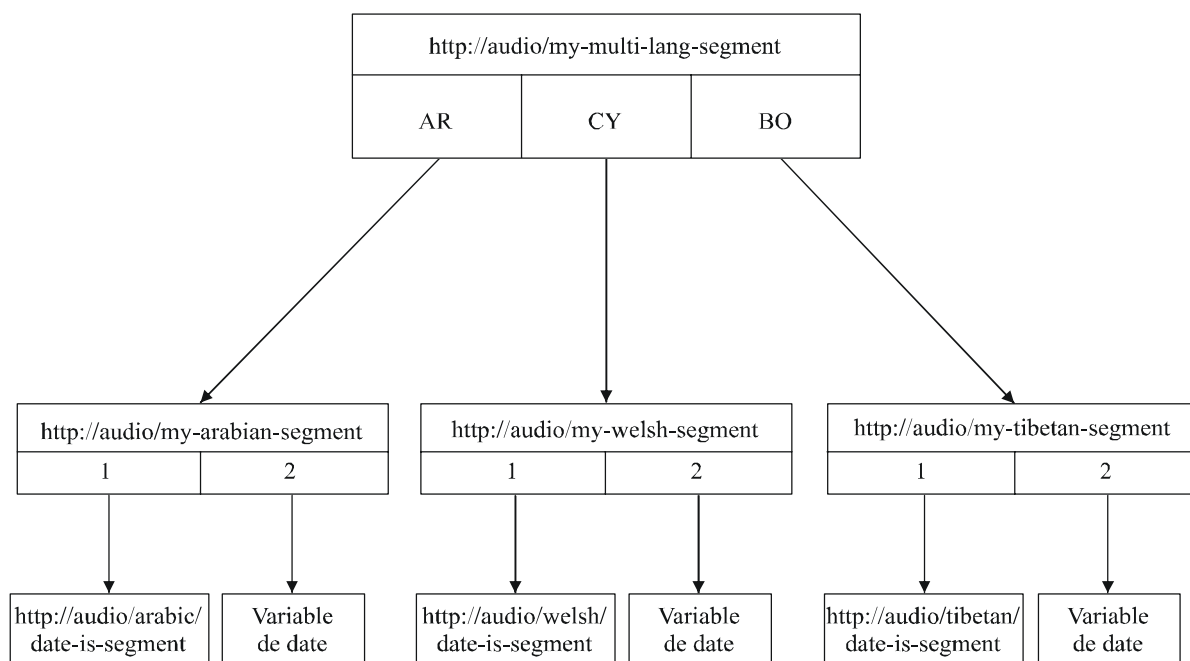
H.248.9_F3

Figure 3/H.248.9 – Exemple d'ensemble

6.1.5.2 Exemple d'ensemble avec variable imbriquée

Dans cet exemple, le fournisseur a mis à disposition trois séquences, une en arabe, une en gallois et une en tibétain, chacune étant constituée d'un segment audio simple suivi par une variable de date. Voir l'illustration sur la Figure 4. Le fournisseur les a regroupées dans un ensemble comprenant les trois séquences avec la langue comme sélecteur. A nouveau, le contrôleur MGC ne peut voir que l'identificateur URI se rapportant à l'ensemble entier, le sélecteur de langue et la variable de date imbriquée.

Au moment de l'exécution, une référence à l'ensemble avec la valeur de sélecteur "ar" et une valeur de variable "20001015" se traduirait par la lecture en arabe de: "Aujourd'hui, nous sommes le 15 octobre 2000".



H.248.9_F4

Figure 4/H.248.9 – Exemple d'ensemble avec variable imbriquée

6.1.6 Variables de type texte générique

La syntaxe prend en charge une capacité facultative permettant de lire une variable de type expression arbitraire. L'expression est représentée dans la spécification de segment au moyen d'un codage UTF-8 (IETF RFC 2279) du système d'écriture par défaut mis à disposition pour la passerelle média. Suivant les capacités de la passerelle média, la langue dans laquelle l'expression est dite peut être mise à disposition ou peut être indiquée par l'utilisation du sélecteur de langue. La capacité est fournie sous la forme d'un type de variable vocale additionnel.

6.2 Paquetage de syntaxe d'annonce de base

Nom du paquetage: Basic Announcement Syntax

Identificateur du paquetage: bannsyx (0x0047)

Description: ce paquetage existe uniquement pour indiquer que la passerelle média est capable de traiter la syntaxe décrite ici. Un contrôleur MGC apprend que la capacité est prise en charge en auditant les paquetages pris en charge par la terminaison au niveau de laquelle la lecture doit avoir lieu et en vérifiant que ce paquetage en fait partie.

La syntaxe définie dans le présent paragraphe sert aux divers signaux de serveur audio évolués définis dans la présente Recommandation à désigner les annonces devant être passées. Elle peut aussi être utilisée pour désigner un contenu multimédia, même s'il peut être souhaitable de définir des extensions (par exemple des types URL additionnels) à cette fin.

Version: 1

Extension de: néant.

6.2.1 Propriétés

Néant.

6.2.2 Événements

Néant.

6.2.3 Signaux

Néant.

6.2.4 Statistiques

Néant.

6.2.5 Procédures

6.2.5.1 Structure générale

Une spécification d'annonce comprend une ou plusieurs spécifications de segment. Chaque spécification de segment décrit soit un segment audio mis à disposition (avec éventuellement des variables imbriquées) soit une variable vocale autonome.

NOTE – La structure générale qui vient d'être décrite peut facilement être généralisée au contenu multimédia, mais l'utilisation de variables est l'une des questions nécessitant un complément d'étude pour ce cas. Pour que la syntaxe de base soit immédiatement applicable au contenu multimédia, les variables vocales sont décrites dans un paquetage distinct.

6.2.5.1.1 Codage ASN.1

Dans le codage ASN.1, la syntaxe générale de paramètre de signal est limitée, en fin de compte, par la production Value de l'Annexe A/H.248.1. Les valeurs de paramètre sont doublement enveloppées, d'abord avec un codage BER interne appliqué pour faciliter l'interprétation du paramètre, suivi par un codage BER externe sous forme de OCTET STRING. La structure générale de la syntaxe de spécification d'annonce de base de serveur AAS aux fins du codage interne est exprimée de la façon suivante:

```
AnnouncementSpec ::= IA5String
```

Les détails concernant la structure de la chaîne sont tels que spécifiés dans le reste du § 6: ils s'appliquent aussi bien au codage textuel qu'au codage ASN.1.

6.2.5.1.2 Codage textuel

Dans le codage textuel, la syntaxe détaillée de paramètre de signal est limitée, en fin de compte, par la production VALUE de l'Annexe B/H.248.1. La description ABNF de la structure grossière d'une spécification d'annonce est la suivante:

```
announcementSpec = DQUOTE segSpec *( COMMA segSpec ) DQUOTE
                    ; DQUOTE et COMMA sont tels que définis dans l'Annexe B/H.248.1.

segSpec = keyword "=" "<" spec ">" ; les crochets angulaires servent de
                    délimiteurs

keyword = "sid"          ; identificateur de segment mis à disposition
        / "var"         ; variable autonome

spec     = provSegSpec   ; identificateur de segment mis à disposition
        / varSegSpec    ; variable autonome

varSegSpec = varSpec    ; niveau général add. facilitant l'extension de
                    ; sélecteur
```

La forme quotedString de VALUE est requise pour la spécification announcementSpec parce qu'une spécification segSpec peut contenir des caractères réservés (par exemple =, <, > comme

indiqué ci-dessus) et parce que des spécifications segSpec successives sont séparées par une virgule. Toutefois, la production VALUE nécessite des échappements pour:

- tous les caractères de commande (%x00-%x1F et %x7F) à l'exception de TAB (%x09);
- le caractère DQUOTE (%x22).

En dehors des identificateurs URI, la question de l'échappement ne se pose que pour les séquences de caractères généraux, qui sont possibles avec les types de variable caractères et expression. (Voir le § 6.5 pour ce dernier.) Cette spécification représente les caractères généraux UTF-8 sous la forme U+xxxx afin d'éviter de devoir recourir à un échappement pour les diverses valeurs d'octet.

L'échappement dans le cadre des identificateurs URI doit être exécuté comme décrit dans l'IETF RFC 2396. L'échappement dans le cadre de spécifications de variables vocales autonomes utilise le même mécanisme que celui décrit dans l'IETF RFC 2396, mais s'applique uniquement aux caractères énumérés ci-dessus, le symbole de pourcentage "%" (qui est utilisé comme caractère d'échappement) et le crochet angulaire de fermeture ">" (utilisé à la fin d'une valeur de variable). (Il s'agit actuellement d'un faux problème, car ni "%" comme caractère utilisé non pas pour l'échappement mais à d'autres fins ni ">" ne figureront dans une quelconque valeur de variable définie dans la présente Recommandation.)

Pour les mots clés de la syntaxe de codage de texte, aucune distinction n'est faite entre majuscules et minuscules. La distinction ou non entre minuscules et majuscules pour les identificateurs URI est définie par les normes applicables. Une distinction est faite entre minuscules et majuscules pour les valeurs de variable uniquement lorsque c'est explicitement spécifié.

6.2.5.2 Spécification de segment mis à disposition

Une spécification de segment mis à disposition est constituée soit d'un nom simple soit d'un identificateur URI formé selon les règles de l'IETF RFC 2396. La syntaxe d'un nom simple est légèrement plus souple que la construction NAME de l'Annexe B/H.248.1; en effet, un nom simple ne doit pas obligatoirement commencer par un caractère alphabétique. La présente Recommandation prend en charge trois types d'identificateur URI:

- file: utilisé pour les segments mis à disposition situés au niveau de la passerelle média;
- ftp: utilisé pour les segments situés au niveau d'un dispositif distant de la passerelle média;
- http: utilisé pour les segments situés soit au niveau soit à distance de la passerelle média. Les segments situés au niveau de la passerelle média doivent utiliser "localhost", comme partie <host> de l'identificateur URI.

Le contrôleur MGC doit utiliser le type http: si le segment audio mis à disposition contient des variables imbriquées. Il en est de même si le segment prend en charge des sélecteurs (voir le § 6.4). Cette restriction est nécessaire car la syntaxe de spécification d'annonce utilise la partie query de l'identificateur http: pour acheminer les valeurs de variable imbriquée (et de sélecteur).

Conformément à l'IETF RFC 2396, les caractères suivants doivent faire l'objet d'un échappement dans tous les identificateurs URI:

- les caractères réservés dans les différents identificateurs URI. L'IETF RFC 1738 contient la description la plus récente des identificateurs file:, ftp: et http:. Conformément à ce document, "/" est réservé pour la séparation des composantes d'une hiérarchie de chemin, ";" est réservé dans les identificateurs ftp: et http: et "?" est réservé dans les identificateurs http:;
- le caractère d'espace;
- les caractères utilisés comme délimiteurs ou pour l'échappement: "<", ">", "#", "%" et "<>";
- les caractères pouvant conduire à des transformations non souhaitées ou à de mauvaises interprétations: "{", "}", "|", "\", "^", "[", "]" et "".

6.2.5.2.1 Codage textuel

Le présent paragraphe contient une description détaillée de la production provSegSpec dont il est question au § 6.2.5.1.2.

```
provSegSpec = simple / ftpurl / httpurl / fileurl

simple = 1* ( ALPHA / DIGIT / "_" )
; ALPHA et DIGIT sont tels que définis dans l'Annexe B/H.248.1

fileurl = "file://" host path
; Voir RFC 1738 pour plus de détails. Pour "file://" une distinction
est faite entre minuscules et majuscules.

ftpurl = "ftp://" [user [":" password ] "@" ] host [":" port ]
[ "/" *(cwd "/" ) name [";type=" type] ]
; Voir RFC 1738 pour plus de détails. Pour "ftp://" une
distinction est faite entre minuscules et majuscules.

httpurl = "http://" host [ ":" port ] [ abs_path [ "?" query ] ]
; Omettre "?" si query est vide.
; Voir RFC 1738 et 2616 pour plus de détails. Pour "http://" une
distinction est faite entre minuscules et majuscules
```

Lorsque la forme httpurl est utilisée, la partie query doit être présente si le segment audio mis à disposition contient des variables imbriquées (ou prend en charge des sélecteurs, voir le § 6.4). La forme générale de la partie query est la suivante:

```
query = category "=" catVal *( "&" category "=" catVal)
category = 1*ALPHA
; aucune distinction n'est faite entre minuscules et majuscules.
```

Outre les règles d'échappement de caractère déjà décrites, le caractère "&" doit faire l'objet d'un échappement (doit être remplacé par "%26") s'il est présent dans un catVal.

6.3 Paquetage de syntaxe de variable vocale

Nom du paquetage: Voice Variable Syntax

Identificateur du paquetage: vvsyx (0x0048)

Description: ce paquetage existe uniquement pour indiquer que la passerelle média est capable de traiter la syntaxe décrite ici. Un contrôleur MGC apprend que la capacité est prise en charge en auditant les paquetages pris en charge par la terminaison au niveau de laquelle la lecture doit avoir lieu et en vérifiant que ce paquetage en fait partie.

La syntaxe définie dans le présent paragraphe sert à désigner des variables vocales, soit sous forme de variables imbriquées dans des segments d'annonce, soit sous forme de variables autonomes. Les valeurs de variable étant spécifiées sous forme de texte, les variables vocales peuvent aussi être utilisées comme variables textuelles lorsque les annonces sont exprimées sous forme de texte.

Version: 2

Extension de: bannsyx version 1

6.3.1 Propriétés

Néant.

6.3.2 Événements

Néant.

6.3.3 Signaux

Néant.

6.3.4 Statistiques

Néant.

6.3.5 Procédures

6.3.5.1 Variables imbriquées

Lorsque des variables imbriquées sont présentes, leurs valeurs sont fournies sous forme de composantes successives séparées par le caractère "&" dans la partie query définie au § 6.2.5.2.1. Une seule valeur est fournie pour chaque variable imbriquée, dans l'ordre de l'imbrication. Sur le plan formel, la syntaxe d'une variable imbriquée est représentée par l'extension suivante à la syntaxe du § 6.2.5.2.1:

```
category =/ "var"  
catVal =/ varVal  
varVal = genval / default / empty  
  
genval = 1* (SafeChar / RestChar / WSP)  
; SafeChar, RestChar et WSP tels que définis dans l'Annexe B/H.248.1.  
; Echappement requis comme indiqué dans le présent paragraphe et dans  
; le § 6.2.5.2.  
; Le codage textuel est donné par la partie de la production  
; correspondant au type applicable du § 6.3.6 (et du § 6.5)  
; qui suit l'étiquette "v=".  
  
default = "-"  
; Le caractère unique "-" suivi par le délimiteur "&" ou ">" indique  
; que l'hôte exécutant devrait utiliser la valeur par défaut mise à  
; disposition, s'il en existe une, de la variable imbriquée.  
  
empty = ""  
; Chaîne vide (c-à-d qu'un délimiteur suit immédiatement "var=").  
; Indique que la variable imbriquée ne doit pas être passée.
```

Si le contrôleur MGC demande la lecture d'une valeur par défaut alors qu'aucune n'est mise à disposition, cela constitue une erreur. Le code d'erreur 607 est applicable, que l'erreur soit signalée dans la réponse à la transaction ou au moyen de l'événement d'échec d'opération audio défini dans le paquetage de base de serveur AAS.

6.3.6 Spécification de segment audio dynamique (variable vocale autonome)

Une spécification audio dynamique est constituée d'un type, d'un éventuel sous-type et d'une valeur pour une seule variable. La plage possible de sous-types varie en fonction du type de variable. La syntaxe de base prend en charge les types et les sous-types énumérés ci-dessous.

Les descriptions en codage textuel figurant dans le présent paragraphe étendent la production varSpec dont il est question au § 6.2.5.1.2 afin d'inclure des variables vocales.

Une spécification de variable autonome inclut le type, l'éventuel sous-type et la valeur. Chacune de ces composantes est introduite par une étiquette: "t=", "s=" et "v=" respectivement. Les composantes successives sont séparées par des virgules. La valeur correspond à l'ensemble de caractères qui suivent "v=" et qui précèdent le crochet angulaire de fermeture ">" de la production

segSpec. Les règles d'échappement du § 6.2.5.1.2 doivent être appliquées aux valeurs de variable en fonction des besoins.

```
varSpec =/ vvarSpec ; Voice variable specification
```

La production varSpec est étendue au § 6.4.5.3.2 afin d'inclure des sélecteurs. La production vvarSpec est étendue dans les paragraphes qui suivent afin d'inclure des spécifications détaillées par type de variable.

6.3.6.1 Type de variable: heure

Définition: heure du jour.

Sous-types: les sous-types associés à cette variable spécifient le format dans lequel l'heure est exprimée (format à 12 heures ou format à 24 heures). Toutefois, dans de nombreuses langues, seule l'expression des éléments de l'heure dans un format unique a un sens. Si, dans une langue, il existe plusieurs modes d'expression de l'heure, on peut utiliser un sous-type pour remplacer le mode par défaut. Si, dans une langue, il n'existe qu'un seul mode d'expression de l'heure, le sous-type peut être omis; si un sous-type est spécifié, il sera ignoré.

Valeur: une chaîne de quatre chiffres spécifiant une heure sous la forme HHMM (selon l'ISO 8601), dans un format à vingt-quatre heures.

Exemple: "1700" correspond à "cinq heures de l'après-midi" dans le format à douze heures ou à "dix-sept heures " dans le format à vingt-quatre heures.

Codage textuel:

```
vvarSpec =/ todSpec  
  
; heure du jour  
todSpec = "t=tod" [ ",s=" ( "t12" / "t24" ) ] ",v=" 4DIGIT  
; Le sous-type permet de choisir le format à 12 ou à 24 heures.  
; La valeur est spécifiée sous la forme HHMM (ISO 8601).
```

6.3.6.2 Type de variable: jour de la semaine

Définition: nom d'un jour de la semaine.

Sous-types: non applicable.

Valeur: un seul chiffre, compris entre "1" (dimanche) et "7" (samedi).

Exemple: "2" correspond à "lundi".

Codage textuel:

```
vvarSpec =/ dowSpec  
  
; jour de la semaine  
dowSpec = "t=dow" ",v=" %d1-7  
; "1" correspond à dimanche ... "7" correspond à samedi
```

6.3.6.3 Type de variable: date

Définition: date constituée de trois composantes: jour du mois, nom du mois et année.

Sous-types: les sous-types associés à cette variable spécifient l'ordre dans lequel les éléments de la date (jour, mois et année) sont exprimés. Toutefois, dans de nombreuses langues, seule l'expression des éléments de la date dans un certain ordre a un sens. Si, dans une langue, il existe plusieurs modes d'expression de la date, on peut utiliser un sous-type pour remplacer le mode par défaut. Si,

dans une langue, il existe un seul mode d'expression de la date, le sous-type peut être omis; si un sous-type est spécifié, il sera ignoré.

Valeur: une chaîne de huit chiffres spécifiant une date sous la forme AAAAMMJJ (selon l'ISO 8601).

Exemple: la valeur "20001015" correspond au "quinze octobre deux mille".

Codage textuel:

```
vvarSpec = / dateSpec

; Date
dateSpec = "t=date" [ ",s=" dateorder ] ",v=" 8DIGIT
; Le sous-type détermine l'ordre dans lequel les composantes
  sont exprimées.
; La valeur est spécifiée sous la forme AAAAMMJJ (ISO 8601).

; Le sous-type Ordre est séparé afin qu'il puisse être étendu si
  on le souhaite.
dateorder = "mdy"           ; mois-jour-année
           / "dmy"         ; jour-mois-année
```

6.3.6.4 Type de variable: mois

Définition: nom du mois.

Sous-types: non applicable.

Valeur: chaîne à deux chiffres spécifiant un mois sous la forme MM, "01" correspondant à janvier, "02" à février, etc.

Exemple: "10" correspond à "octobre".

Codage textuel:

```
vvarSpec = / monthSpec

; mois
monthSpec = "t=month" ",v=" 2DIGIT
; "01" correspond à janvier ... "12" correspond à décembre
```

6.3.6.5 Type de variable: durée

Définition: période de temps exprimée dans une ou plusieurs unités de temps selon qu'il convient.

Sous-types: non applicable.

Valeur: entier donnant un nombre de secondes.

Exemple: la valeur "3661" correspond à "une heure, une minute et une seconde."

Codage textuel:

```
vvarSpec = / durSpec

; Durée
durSpec = "t=dur" ",v=" 1*DIGIT
```

6.3.6.6 Type de variable: chiffres

Définition: séquence de chiffres lus un par un.

Sous-types: non applicable.

Valeur: chaîne de chiffres de longueur arbitraire, donnés dans l'ordre dans lequel ils doivent être lus.

Exemple: la valeur "61360961" correspond à "six un trois six zéro neuf six un".

Codage textuel:

```
vvarSpec = / digitSpec
; séquence de chiffres
digitSpec = "t=digits" ",v=" 1*DIGIT
```

6.3.6.7 Type de variable: caractères

Définition: séquence pouvant contenir des caractères alphabétiques en majuscules ou en minuscules (s'il existe des majuscules et des minuscules dans le système d'écriture considéré), des chiffres et les caractères spéciaux # et *. Une distinction est faite entre majuscules et minuscules pour les caractères alphabétiques (à nouveau s'il existe des minuscules et des majuscules).

Sous-types: non applicable.

Valeur: les caractères valables du jeu de caractères ASCII sont a-z, A-Z, 0-9, # et *. Il est à noter qu'un échappement est nécessaire concernant le caractère # lorsque celui-ci est présent dans la valeur d'une variable imbriquée. Les restrictions relatives aux caractères des autres systèmes d'écriture nécessitent un complément d'étude, mais elles doivent respecter le but qui est d'utiliser ce type de variable pour lire des invitations à numéroté, des numéros de téléphone ou des noms et des adresses.

Codage textuel:

```
vvarSpec = / charSpec
; séquence de caractères
charSpec = "t=chars" ",v="
( 1*( LOWALPHA / UPALPHA / DIGIT / ( "#" / "%23" ) / "*" )
; Chaîne ASCII, restreinte à (a-z, A-Z, 0-9, #, *)
; Note - nécessité d'échappement pour "#" dans la valeur d'une
variable imbriquée
/ ( "U+" 2*12HEX ) *( "." 2*12HEX ) ) )
; Chaîne UTF-8 générale sous forme d'une séquence de valeurs
hexadécimales
; séparées par des points commençant par "U+", représentant de
1 à 6 octets.
; LOWALPHA, UPALPHA, DIGIT, HEX tels que définis dans RFC 2396
```

6.3.6.8 Type de variable: argent

Définition: quantité d'argent dans une monnaie donnée, exprimée dans les unités de cette monnaie et sous forme d'une quantité positive ou d'une quantité négative comme indiqué par le signe de la valeur.

Sous-types: les différents types de monnaie spécifiés dans l'ISO 4217. On donne ci-après un petit extrait de cette norme:

Code alphabétique	Code numérique	Monnaie	Entité
VEB	862	Bolivar	Venezuela
VND	704	Dông	Viet Nam
USD	840	Dollar des Etats-Unis	Iles Vierges (britanniques)

Valeur: entier facultativement signé, donnant une quantité d'argent spécifiée dans la plus petite unité d'une monnaie donnée.

Exemple: "110" en dollars des Etats-Unis correspond à "un dollar et dix cents."

Codage textuel:

```
vvarSpec = / moneySpec
; quantité d'argent (positif ou négatif)
moneySpec = "t=money" [",s=" 3ALPHA] ",v=" [ "-" ] 1*DIGIT
; le sous-type est un code alphabétique de l'ISO 4217
```

6.3.6.9 Type de variable: entier

Définition: entier.

Sous-types: indiquent si le nombre est exprimé sous forme de nombre cardinal ou de nombre ordinal.

Valeur: entier facultativement signé. Les entiers négatifs ne sont autorisés qu'avec le sous-type cardinal.

Exemple: "100" correspond à "cent" (cardinal) ou à "centième" (ordinal).

Codage textuel:

```
vvarSpec = / intSpec
; entier (ordinal ou cardinal positif ou négatif)
intSpec = "t=int" [",s=" ( "card" / "ord" ) ] ",v=" [ "-" ] 1*DIGIT
; les valeurs négatives ne sont autorisées que pour les nombres
cardinaux
```

6.3.6.10 Type de variable: silence

Définition: période de silence d'une certaine durée.

Sous-types: non applicable.

Valeur: entier non signé donnant la durée de la période de silence en centaines de millisecondes.

Exemple: "10" correspond à une seconde de silence.

Codage textuel:

```
vvarSpec = / hushSpec
; intervalle de silence
hushSpec = "t=sil" ",v=" %d1-600
; durée du silence, par pas de 100 ms, 1 minute max.
```

6.3.6.11 Type de variable: tonalité

Définition: lecture d'une tonalité générée par un algorithme. Il s'agit de l'homologue du paramètre de la partie d'application du réseau intelligent (INAP, *intelligent network application part*) "Tone".

Tonalité

- toneId indique la tonalité à envoyer
- duration indique la durée en centaines de millisecondes de la tonalité à envoyer. ZERO indique une durée illimitée.

Sous-types: non applicable.

Valeur: entier ≥ 0 (pour les deux composantes du paramètre "Tone")

Codage textuel:

```
vvarSpec =/ toneSpec
toneSpec = "t=tone" ",tid=" UINT32
          ["dur=" UINT32]
          ; spécifiée en centaines de millisecondes
```

6.4 Ensemble: extension de la syntaxe de base

Nom du paquetage: Announcement Set Syntax

Identificateur du paquetage: setsyx (0x0049)

Description: ce paquetage existe uniquement pour indiquer que la passerelle média est capable de traiter la syntaxe décrite ici. Un contrôleur MGC apprend que la capacité est prise en charge en auditant les paquetages pris en charge par la terminaison au niveau de laquelle la lecture doit avoir lieu et en vérifiant que ce paquetage en fait partie.

La syntaxe définie dans le présent paragraphe sert à choisir différents membres d'ensembles de segments médias qui acheminent un contenu sémantique équivalent.

Version: 2

Extension de: bannsyx version 1

6.4.1 Propriétés

Néant.

6.4.2 Evénements

Néant.

6.4.3 Signaux

Néant.

6.4.4 Statistiques

Néant.

6.4.5 Procédures

Les ensembles de segments sont décrits au § 6.1.5. Ils constituent une extension facultative à la syntaxe de base pour la spécification d'un segment média. La présente Recommandation définit deux étiquettes de sélecteur:

6.4.5.1 Sélecteur de langue "lang"

Les valeurs associées à ce sélecteur sont les étiquettes définies dans l'IETF RFC 3066. Ces étiquettes sont la combinaison d'une langue et d'autres informations facultatives telles qu'une région ou un pays (par exemple l'étiquette "en-us" correspond à l'anglais tel qu'il est parlé aux Etats-Unis et "cy" correspond au gallois (aucun qualificatif de lieu n'est nécessaire). Le concept de sélecteur est applicable au contenu multimédia, même si les exemples fournis dans la présente Recommandation illustrent uniquement son utilisation avec des segments audio.

6.4.5.2 Sélecteur "tatb" pour les attributs textuels

Les valeurs associées à ce sélecteur sont des entiers sans signe à 16 bits.

Les valeurs employées doivent avoir été fixées d'un commun accord par la partie commande des services (partie INAP) et la partie équipement, où sont situées les applications texte vers parole. Les

variables de type texte prononcé ou texte affiché (voir le § 6.5.5.1) peuvent être spécifiées dans ce sélecteur.

6.4.5.3 Codage textuel pour les deux sélecteurs

Dans une spécification de segment audio, tous les sélecteurs doivent être spécifiés dans une seule liste de la forme

```
selList = "sel=" selSpec *( "&" selSpec )
```

Chaque construction selSpec donne le nom d'un type de sélecteur et lui attribue l'une de ses valeurs possibles.

```
selSpec = seltype "=" selval  
  
seltype = "lang" / "tatb" / otherSel  
  
otherSel = NAME  
; Comme défini dans l'Annexe B/H.248.1.  
; Pour les types de sélecteur, aucune distinction n'est faite  
; entre minuscules et majuscules.  
  
selval = Language-Tag / UINT16 / otherSelVal  
; Restriction: si seltype = "lang" then selval = Language-Tag,  
; si seltype = "tatb" then selval = UINT16
```

La définition de Language-Tag est tirée de l'IETF RFC 3066:

```
Language-Tag = Primary-subtag *( "-" Subtag )  
; Aucune distinction n'est faite entre minuscules et majuscules.  
  
Primary-subtag = 1*8ALPHA  
; Tiré généralement de l'ISO 639, mais voir RFC 3066  
  
Subtag = 1*8(ALPHA / DIGIT)  
; Tiré généralement de l'ISO 3166, mais voir RFC 3066
```

D'une manière générale, les valeurs de sélecteur peuvent se composer de n'importe quelle combinaison de caractères satisfaisant la production safeChar de l'Annexe B/H.248.1, sous réserve des règles d'échappement applicables dans le contexte de la spécification de segment.

```
otherSelVal = safeChar
```

6.4.5.3.1 Codage textuel pour les segments mis à disposition

Tout comme la spécification de valeurs de variable imbriquée, la spécification de valeurs de sélecteur est faite dans la partie query d'une adresse URL http:. Pour simplifier l'analyse, les sélecteurs doivent être spécifiés après toutes les valeurs de variable imbriquée requises. Si des valeurs de variable imbriquée sont présentes, la dernière valeur de variable est séparée de la première valeur de sélecteur par le caractère &. La production query telle qu'elle figure au § 6.2.5.2.1 est donc étendue comme suit:

```
query =/ ( ( "var=" varVal *( "&var=" varVal ) ) "&" selList  
; valeur(s) de variable imbriquée suivie(s) par une (des)  
spécification(s) de sélecteur  
/ selList  
; spécification(s) de sélecteur uniquement
```

6.4.5.3.2 Codage textuel pour les variables autonomes

La définition de la production varSegSpec donnée au § 6.3.6 est étendue comme suit:

```
varSegSpec =/ varSpec "&" selList
```

6.5 Type de variable texte générique: extension de la syntaxe de base

Nom du paquetage: Phrase Variable Syntax

Identificateur du paquetage: phrsyx (0x004a)

Description: ce paquetage existe uniquement pour indiquer que la passerelle média est capable de traiter la syntaxe décrite ici. Un contrôleur MGC apprend que la capacité est prise en charge en auditant les paquetages pris en charge par la terminaison au niveau de laquelle la lecture doit avoir lieu et en vérifiant que ce paquetage en fait partie.

La syntaxe définie dans le présent paragraphe sert à désigner des variables vocales expression, qui permettent de passer d'un texte arbitraire à la voix.

Version: 2

Extension de: vvsyx version 1

6.5.1 Propriétés

Néant.

6.5.2 Evénements

Néant.

6.5.3 Signaux

Néant.

6.5.4 Statistiques

Néant.

6.5.5 Procédures

Le présent paragraphe définit le type de variable texte générique comme étant une extension facultative à l'ensemble de base des types de variable définis au § 6.3.6. Voir également le § 6.1.6.

6.5.5.1 Type de variable: expression

Définition: expression, prononcée ou affichée, qui est spécifiée sous la forme d'une séquence de caractères ASCII ou UTF-8 conformément à l'orthographe de la langue considérée. Cette variable peut être spécifiée avec le sélecteur pour les "attributs textuels".

Sous-types:

- **spk** (texte prononcé): valeur par défaut.
- **dsp** (texte affiché): avec ce sous-type, "phrase" a les mêmes attributs de syntaxe, mais correspond à un texte à afficher sur le terminal de l'utilisateur.

Valeur: les caractères valables du jeu de caractères ASCII sont a-z, A-Z et blanc (" "). Il est à noter que les blancs doivent faire l'objet d'un échappement sous la forme "%20" lorsqu'ils sont présents dans la valeur d'une variable imbriquée. Concernant les caractères valables de tout autre jeu de caractères, un complément d'étude peut être nécessaire.

Codage textuel:

le présent paragraphe étend la définition de la production `vvarSpec` donnée au § 6.3.6 afin d'inclure le type de variable expression.

```
vvarSpec =/ phraseSpec

; Expression
phraseSpec = "t=phrase" ",v=" [" ,s=spk" ] [" ,s=dsp" ]
  ( 1*( ALPHA / ( %x20 / "%20" ) )
    ; Chaîne ASCII, limitée à (a-z, A-Z, blanc)
    ; Note - nécessité d'échappement pour les blancs
    ; compris dans la valeur d'une variable imbriquée
  / ( "U+" 2*12HEX ) *( "." 2*12HEX ) ) )
  ; Chaîne UTF-8 générale sous la forme d'une séquence
  ; de valeurs à codage hexadécimal séparées par des
  ; points, commençant par "U+".
  ; HEX comme défini dans RFC 2396
; Une distinction est faite entre minuscules et majuscules car
; la lecture des minuscules et des majuscules peut être différente
; dans certaines langues.
```

6.6 Exemples

Ces exemples utilisent le signal de lecture du paquetage de base de serveur audio amélioré, le signal de lecture et de collecte du paquetage de collecte de chiffres de serveur AAS et le signal de lecture et d'enregistrement du paquetage d'enregistrement de serveur AAS.

Lecture d'une annonce constituée d'un seul segment résidant, au niveau du serveur audio, dans un fichier non hiérarchique:

```
Signals { aasb/play { an = "sid=<file://1947>" } }
```

Lecture d'une annonce constituée d'un seul segment résidant, au niveau du serveur audio, dans un fichier non hiérarchique utilisant la convention `http://localhost`. Cet exemple est strictement équivalent au premier exemple:

```
Signals { aasb/play { an = "sid=<http://localhost/1947>" } }
```

Lecture d'une annonce constituée d'un seul segment résidant, au niveau du serveur audio, dans un système de fichiers hiérarchiques:

```
Signals { aasb/play { an = "sid=<file://audio/current/1947>" } }
```

Lecture d'une annonce constituée d'un seul segment résidant dans une machine appelée "darkstar" qui est extérieure au serveur audio:

```
Signals { aasb/play { an = "sid=<http://darkstar/welcome>" } }
```

Lecture d'une annonce constituée de plusieurs segments. On a ajouté des retours à la ligne pour plus de clarté dans la présentation.

```
Signals { aasb/play { an = "sid=<file://audio/voice/brenda/123>,
                          sid=<file://audio/voice/althea/098>,
                          sid=<file://audio/voice/delia/086>" } }
```

Lecture d'une annonce constituée d'un enregistrement suivi par une variable vocale directe:

```
Signals { aasb/play {  
    an = "sid=<file://gdtrfb>,var=<t=dat,s=mdy,v=19550809>" } }
```

Lecture d'une annonce qui exprime le numéro de téléphone 0800 321 589 sous la forme "zéro huit cents ... trois deux un ... cinq huit neuf":

```
Signals { aasb/play { an= "var=<t=dig,v=0>,var=<t=int,s=car,v=800>,  
    var=<t=sil,v=5>,var=<t=dig,v=321>,  
    var=<t=sil,v=5>,var=<t=dig,v=589>" } }
```

Lecture d'une annonce avec deux variables imbriquées. Les valeurs de variable sont données dans l'ordre dans lequel elles apparaissent dans l'annonce:

```
Signals { aasb/play {  
    an = "sid=<http://localhost/113?var=3999&var=20001015>" } }
```

Lecture d'une annonce en anglais avec l'accent de Glasgow, en supposant que <http://localhost/1947> désigne un ensemble et que la syntaxe d'ensemble est prise en charge:

```
Signals { aasb/play { an ="sid=< http://localhost/1947?sel=lang=en-gb-  
glg>" } }
```

Lecture d'une annonce en danois avec une voix féminine. On suppose que l'annonce a été mise à disposition conjointement avec un sélecteur de type "sexe" avec "féminin" comme l'une des valeurs possibles et avec le sélecteur de type "lang".

```
Signals { aasb/play {  
    an ="sid=<http://localhost/jackstraw/ann45?sel=lang=da&  
gender=female>" } }
```

Lecture de la première partie d'une annonce en anglais, de la deuxième partie dans la langue par défaut et de la troisième partie en français. Les deux premiers segments résident dans le serveur audio et le troisième dans une machine distante. On a ajouté des retours à la ligne pour plus de clarté dans la présentation.

```
Signals { aasb/play { an ="sid=<http://localhost/ann1?sel=lang=eng>,  
    sid=<http://localhost/audio/myannouncements/ann2>,  
    sid=<http://darkstar/audio/ann3?sel=lang=fra>" } }
```

Lecture d'une annonce avec une variable date autonome en anglais:

```
Signals { aasb/play { an = "sid=<http://darkstar/audio/ann7?sel=lang=en>,  
    var=<t=date,s=mdy,v=20001015&sel=lang=en>" } }
```

Lecture d'une invite et collecte d'un mot de passe à huit chiffres. Si besoin est, lecture d'une nouvelle invite, d'une invite en cas d'absence de chiffre et d'une annonce de succès ou d'échec. Il est donné à l'utilisateur trois fois la possibilité de saisir le mot de passe. Par défaut, la saisie du mot de passe interrompt le mécanisme d'invite.

```
Signals { aasdc/playcol { ip = "sid=<file://enterpassword>",&br/>    rp = "sid=<file://tryagain>",&br/>    nd = "sid=<file://nodigits>",&br/>    sa = "sid=<file://goodpassword>",&br/>    fa = "sid=<file://badpassword>",&br/>}
```

```

mxatt = 3 ,
dm = passwmap }

```

Lecture d'une invite et enregistrement de la voix. Si l'utilisateur ne parle pas, lecture d'une invite en cas d'absence de parole. Par défaut, il n'y a pas d'annonce de succès ou d'échec et les temporisations avant et après parole sont chacune de 5 secondes. Il est donné à l'utilisateur deux fois la possibilité de faire un enregistrement. L'enregistrement ne doit pas dépasser 5 minutes et il doit avoir un identificateur de segment choisi par la passerelle média.

```

Signals { aasrec/playrec { ip = "sid=<file://sayname>",
                          ns = "sid=<file://nospeech>",
                          mxatt = 2,
                          rlt = 3000,
                          rid = "$"
                        } }

```

Lecture d'une annonce à une vitesse plus rapide que la vitesse normale de 10% et avec un volume plus faible que le volume normal de cinq décibels. Triple lecture de l'annonce avec deux secondes de silence entre chaque lecture.

```

Signals { aasb/play { an = "sid=<file://brenda>",
                    sp = +10 , vl = -5 , it = 3 , iv = 20 } }

```

Il est donné à l'utilisateur trois fois la possibilité de saisir un numéro à 11 chiffres commençant par 0 ou 1. Si l'utilisateur fait une erreur en saisissant les chiffres, il peut appuyer sur la touche *, ce qui permet d'ignorer les chiffres déjà collectés, de repasser l'invite et de reprendre la collecte.

```

Signals { aasdc/playcol { ip = "sid=<file://enterdigits>",
                        mxatt = 3, dm = elevendig, rsk = "*" } },
DigitMap = elevendig { [0-1]xxxxxxxxxxx }

```

7 Nouveaux codes d'erreur H.248.1

Les paquetages de la présente Recommandation sont fondés sur une syntaxe qui vient d'être décrite et reposent sur des ressources de segment audio qui sont identifiées au moyen de cette syntaxe. Les erreurs survenant lors de l'exécution de demandes de transaction peuvent apparaître avant que la réponse à la transaction ne soit retournée ou beaucoup plus tard. Afin d'améliorer la signalisation des erreurs dans la réponse à la transaction, le présent paragraphe définit un certain nombre de codes d'erreur propres à l'application, qui sont valables pour les paquetages de la Rec. UIT-T H.248.9.

NOTE – La description des paquetages donne parfois des indications sur le moment où les erreurs devraient être détectées pour des signaux particuliers. Toutefois, le moment où les erreurs sont détectées dépend, dans une certaine mesure, de l'implémentation voire de la ressource particulière: une implémentation peut assembler les segments audio "juste à temps" et découvrir qu'il manque un segment au milieu de la lecture, tandis qu'une autre implémentation s'assure que toutes les ressources sont présentes avant de commencer la lecture. Un événement d'échec est défini dans le paquetage de base de serveur AAS afin de pouvoir signaler de manière autonome les erreurs dans le premier cas. Dans le second cas, il semble raisonnable de signaler toute erreur dans la réponse à la transaction.

7.1 Syntaxe illégale dans une spécification d'annonce

Code d'erreur n°: 600

Nom: Illegal syntax within an announcement specification

Définition: un certain aspect d'une spécification d'annonce n'est pas conforme à la syntaxe requise.

Texte de l'erreur dans le descripteur d'erreur: la partie incriminée de la spécification.

Observation: indique une éventuelle erreur logicielle au niveau de la passerelle média ou du contrôleur MGC.

7.2 Type de variable non pris en charge

Code d'erreur n°: 601

Nom: Variable type not supported

Définition: alors que la syntaxe d'une spécification de variable autonome est apparemment correcte, la passerelle média ne prend pas en charge le type de variable spécifié.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: le contrôleur MGC peut faire un audit pour déterminer les types de variable autres que de base pris en charge par la passerelle média.

7.3 Valeur de variable hors intervalle

Code d'erreur n°: 602

Nom: Variable value out of range

Définition: la valeur est syntaxiquement correcte mais inacceptable. Ce code s'applique à la fois aux variables imbriquées et aux variables autonomes. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: indique une éventuelle erreur de mise à disposition au niveau de la passerelle média ou du contrôleur MGC.

7.4 Catégorie non prise en charge

Code d'erreur n°: 603

Nom: Category not supported

Définition: l'entité responsable de l'exécution de la partie query d'un segment audio mis à disposition a rencontré une catégorie de composante (par exemple "sel") qu'elle ne prend pas en charge. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: le contrôleur MGC peut faire un audit pour déterminer les catégories autres que de base prises en charge par la passerelle média.

7.5 Type de sélecteur non pris en charge

Code d'erreur n°: 604

Nom: Selector type not supported

Définition: l'étiquette qui suit le mot clé "sel=" n'est pas mise à disposition sous forme de type de sélecteur au niveau de l'entité responsable de l'exécution de la partie

query d'un segment audio mis à disposition. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: indique une probable erreur de mise à disposition au niveau de la passerelle média ou du contrôleur MGC.

7.6 Valeur de sélecteur non prise en charge

Code d'erreur n°: 605

Nom: Selector value not supported

Définition: la valeur donnée n'est pas mise à disposition au niveau de l'entité responsable de l'exécution de la partie query d'un segment audio mis à disposition. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: indique une probable erreur de mise à disposition au niveau de la passerelle média ou du contrôleur MGC.

7.7 Identificateur de segment inconnu

Code d'erreur n°: 606

Nom: Unknown segment ID

Définition: un segment identifié par une référence de segment mis à disposition ne peut pas être localisé. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio. Voir le code d'erreur n° 611 pour le cas particulier d'un échec de l'opération de transformation en segment persistant.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: indique une probable erreur de mise à disposition au niveau de la passerelle média ou du contrôleur MGC.

7.8 Discordance entre la spécification de lecture et les données mises à disposition

Code d'erreur n°: 607

Nom: Mismatch between play specification and provisioned data

Définition: cette erreur indique une discordance entre le contenu de la partie query d'une spécification de segment mis à disposition et ce qui a été mis à disposition pour ce segment. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: indique une probable erreur de mise à disposition au niveau de la passerelle média ou du contrôleur MGC.

7.9 Erreur de mise à disposition

Code d'erreur n°: 608

Nom: Provisioning error

Définition: par exemple, un identificateur de segment mis à disposition pointe bien sur une séquence de segments physiques, mais il en manque un. Suivant l'implémentation, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: la spécification de segment incriminée.

Observation: indique une probable erreur de mise à disposition au niveau de la passerelle média ou du contrôleur MGC.

7.10 Décalage non valable

Code d'erreur n°: 609

Nom: Invalid offset

Définition: la valeur du décalage dans un signal de lecture et de collecte dépasse la durée de l'invite initiale. Comme il est possible que la passerelle média ne détecte pas cette erreur avant que la réponse à la transaction ne soit envoyée, cette erreur peut être signalée par l'événement d'échec d'opération audio.

Texte de l'erreur dans le descripteur d'erreur: –

Observation: –

7.11 Aucun identificateur de segment libre

Code d'erreur n°: 610

Nom: No free segments ids

Définition: l'espace local des identificateurs de segment est épuisé et le paramètre d'identificateur d'enregistrement de la commande de lecture et d'enregistrement avait la valeur "\$".

Texte de l'erreur dans le descripteur d'erreur: –

Observation: –

7.12 Segment temporaire non trouvé

Code d'erreur n°: 611

Nom: Temporary segment not found

Définition: la transformation en segment persistant a échoué car le segment temporaire cible n'était pas associé à la terminaison considérée.

Texte de l'erreur dans le descripteur d'erreur: l'identificateur URI du segment manquant.

Observation: il se peut que le segment n'existe plus en raison d'une expiration de temporisation ou il se peut qu'il n'ait pas été enregistré au niveau de la terminaison en question.

7.13 Segment en cours d'utilisation

Code d'erreur n°: 612

Nom: Segment in use

Définition: une demande de suppression d'un segment persistant a échoué car ce segment est utilisé par une autre opération.

Texte de l'erreur dans le descripteur d'erreur: l'identificateur URI du segment en cours d'utilisation.

Observation: –

8 Paquetage de base de serveur audio évolué

Nom du paquetage: Advanced audio server base package

Identificateur du paquetage: aasb (0x0033)

Description: Le paquetage de base de serveur audio évolué (AAS, *advanced audio server*) prend en charge un signal permettant de passer une annonce et un événement permettant d'indiquer un échec de la demande de lecture. En ce qui concerne l'événement, le paquetage définit un code de retour et certaines valeurs possibles de ce code. Le paquetage aasb/play ne peut être appliqué à une terminaison que si celle-ci prend au moins en charge le paquetage de syntaxe d'annonce de base.

Version: 1

Extension de: néant

8.1 Propriétés

Néant.

8.2 Evénements

8.2.1 Echec d'opération audio

Nom de l'événement: Audio operation failure

Identificateur de l'événement: audfail (0x0001)

Description: indique l'échec d'une opération de serveur audio évolué après le retour de la réponse à la transaction qui a invoqué cette opération.

Paramètres du descripteur d'événement:

Néant.

Paramètres du descripteur d'événement observé:

Nom du paramètre: Return Code

Identificateur du paramètre: rc (0x0001)

Description: ce paramètre indique pourquoi une opération de serveur audio évolué a échoué.

Type: entier

En option: non

Valeurs possibles: elles sont comprises entre 600 et 699. Les codes d'échec 600 à 612 signalent les mêmes erreurs que les codes d'erreur correspondants du § 7, sauf que, dans le cas présent, les conditions d'erreur sont détectées après que la réponse à la transaction a été retournée. (Le fait que ces erreurs puissent être détectées après la réponse dépend de l'implémentation et des segments audio invoqués.)

600 Syntaxe illégale dans une spécification d'annonce

601 Type de variable non pris en charge

602 Valeur de variable hors intervalle

- 603 Catégorie non prise en charge
- 604 Type de sélecteur non pris en charge
- 605 Valeur de sélecteur non prise en charge
- 606 Identificateur de segment inconnu
- 607 Discordance entre la spécification de lecture et les données mises à disposition
- 608 Erreur de mise à disposition
- 609 Décalage non valable
- 610 Aucun identificateur de segment libre
- 611 Segment temporaire non trouvé
- 612 Segment en cours d'utilisation

En plus de ces codes d'erreur et d'échec communs, les valeurs suivantes de code d'échec sont définies dans le paquetage de base. D'autres valeurs de code peuvent être ajoutées par d'autres paquetages.

- 615 Défaillance matérielle de serveur AAS
- 616 Défaillance non spécifiée de serveur AAS

Valeurs par défaut: néant

8.3 Signaux

8.3.1 Lecture

Nom du signal: Play

Identificateur du signal: play (0x0001)

Description: lecture d'un ou de plusieurs segments audio.

Type de signal: la valeur par défaut est le signal BR (la lecture se poursuit jusqu'à ce que le nombre spécifié ou le nombre par défaut d'itérations soit terminé).

Durée: non applicable aux signaux BR.

8.3.1.1 Paramètres additionnels

8.3.1.1.1 Nom du paramètre: Annonce

Identificateur du paramètre: an (0x0001)

Description: une annonce à passer. Elle est constituée d'un ou de plusieurs segments audio. Il s'agit du seul paramètre non facultatif pour le signal de lecture.

Type: chaîne

En option: non

Valeurs possibles: une séquence de spécifications de segment conformes à la syntaxe décrite au § 6. La prise en charge d'éléments facultatifs de cette syntaxe est indiquée par la présence des paquetages correspondants au niveau de la terminaison.

Valeurs par défaut: néant

8.3.1.1.2 Nom du paramètre: Iterations

Identificateur du paramètre: it (0x0002)

Description: le nombre maximal de fois qu'une annonce doit être passée.

Type: entier

En option: oui

Valeurs possibles: comme décrit plus loin, la lecture peut se terminer avant que le nombre spécifié d'itérations soit terminé si le type de signal est mis à TO et si la limite fixée par le paramètre de durée est atteinte en premier. La valeur 0 (zéro) indique que l'annonce doit être répétée jusqu'à ce qu'elle soit interrompue par d'autres moyens, indépendamment du nombre d'itérations.

Valeur par défaut: 1

8.3.1.1.3 Nom du paramètre: Interval

Identificateur du paramètre: iv (0x0003)

Description: l'intervalle de silence à insérer entre lectures successives, exprimé en dizaines de millisecondes.

Type: entier

En option: oui

Valeurs possibles: valeurs supérieures ou égales à 0.

Valeurs par défaut: néant

8.3.1.1.4 Nom du paramètre: Speed

Identificateur du paramètre: sp (0x0004)

Description: la vitesse relative de lecture d'annonce exprimée sous la forme d'une différence positive (plus rapide) ou négative (plus lente) en pourcentage par rapport à la vitesse normale de lecture. La vitesse réelle de lecture exprimée en pourcentage de la vitesse normale est égale à la valeur de ce paramètre plus 100.

Type: entier

En option: oui

Valeurs possibles: valeurs supérieures ou égales à -99.

Valeur par défaut: 0

8.3.1.1.5 Nom du paramètre: Volume

Identificateur du paramètre: vl (0x0005)

Description: le volume relatif de lecture d'annonce exprimé sous la forme d'une différence positive (plus fort) ou négative (plus faible) en décibels par rapport au volume normal de lecture.

Type: entier

En option: oui

Valeurs possibles: dépendent de l'implémentation.

Valeur par défaut: 0

8.4 Statistiques

Néant.

8.5 Procédure

Le contrôleur MGC invoque le signal aasb/play avec au moins le paramètre d'annonce positionné sur une certaine annonce. La lecture d'annonce peut être interrompue normalement par des événements ou par de nouveaux positionnements du descripteur de signal. Si le paramètre de type de signal est mis sur OO, il s'agit du seul moyen de mettre fin à l'annonce: les paramètres de durée et d'itérations sont tous deux ignorés. Si le paramètre de type de signal est mis à sa valeur par défaut (BR), la durée est ignorée mais l'annonce se terminera lorsque le nombre spécifié d'itérations sera terminé. Si le paramètre de type de signal est mis à TO, l'annonce se terminera dès que la durée donnée par le paramètre de durée (qui doit être spécifié) est écoulée ou dès que la lecture du nombre d'itérations spécifié par le paramètre d'itérations et les pauses associées sont terminées, si elles se terminent avant.

Le contrôleur MGC peut utiliser la capacité standard de notification d'achèvement de signal pour déterminer le moment et le motif de l'interruption de la lecture. Afin d'avoir plus de détails sur les échecs, le contrôleur MGC devrait activer l'événement d'échec de lecture.

Le signal aasb/play peut être utilisé dans le cadre d'une opération de collecte de chiffres avec invite. Le contrôleur MGC doit activer un événement pour chaque chiffre DTMF ou un script de numérotation H.248.1 standard et doit invoquer le signal aasb/play. En cas d'activation d'un événement pour chaque chiffre DTMF, le contrôleur MGC peut, si nécessaire, lever le drapeau KeepActive de l'événement de sorte que le mécanisme d'invite soit exécuté jusqu'à la fin même si l'abonné commence à saisir des chiffres plus tôt. Si le contrôleur MGC détermine que l'abonné a fait une erreur ou qu'il n'a rien saisi, il peut invoquer à nouveau le signal aasb/play avec de nouvelles invites selon qu'il est nécessaire.

9 Paquetage de collecte de chiffres de serveur AAS

Nom du paquetage: AAS Digit Collection Package

Identificateur du paquetage: aasdc (0x0034)

Description: le paquetage de collecte de chiffres de serveur AAS étend le paquetage de base de serveur AAS en prenant en charge un signal et un événement permettant de coordonner la collecte de chiffres avec la lecture d'invites. Il permet d'optimiser la collecte de chiffres par rapport à l'utilisation du signal aasb/play, décrite dans le § 8. L'utilisation du signal aasdc/playcol permet d'éviter de recourir à l'échange de messages qui, autrement, serait nécessaire pour invoquer de nouvelles invites et pour signaler les chiffres qui ne sont pas conformes à une séquence attendue.

Version: 2

Extension de: aasb (0x0033) version 1

9.1 Propriétés

Néant.

9.2 Événements

9.2.1 Echec d'opération audio

Nom de l'événement: Audio Operation Failure

Identificateur de l'événement: audfail (0x0001)

Description: ce paquetage définit les autres valeurs suivantes pour le code de retour renvoyé par l'événement d'échec d'opération audio défini au § 8.2.1:

- 617 fin d'opération prématurée. L'opération audio a été interrompue avant son achèvement normal, par la reconnaissance d'un événement avec le drapeau KeepActive non levé, par le remplacement du descripteur de signal sans poursuite du signal ou par l'expiration de la temporisation liée à la durée du signal.
- 618 Détection d'une séquence de commande non valable.
- 619 Nombre maximal de tentatives dépassé. Les chiffres collectés lors de la dernière tentative ne correspondent à aucune séquence du script de numérotation.
- 620 Absence de chiffre. Le nombre maximal de tentatives a été atteint et aucun chiffre n'a été saisi lors de la dernière tentative.

9.2.1.1 Paramètres de descripteur d'événement

Néant.

9.2.1.2 Paramètres de descripteur d'événement observé

Voir le § 8.2.1.

9.2.2 Succès de la lecture et de la collecte

Nom de l'événement: Play Collect Success

Identificateur de l'événement: pcolsucc (0x0002)

Description: cet événement indique que l'exécution d'un signal playcol a abouti.

9.2.2.1 Paramètres de descripteur d'événement

Néant.

9.2.2.2 Paramètres de descripteur d'événement observé

9.2.2.2.1 Chiffres collectés

Nom du paramètre: Digits collected

Identificateur du paramètre: dc (0x0003)

Description: les chiffres DTMF qui ont été collectés pendant un signal de lecture.

Type: chaîne

En option: non

Valeurs possibles: n'importe quelle séquence de chiffres DTMF valables 0-9, A-D ou a-d, * ou #. Un chiffre peut être précédé du modificateur de longue durée "Z" ou "z" si la détection d'une tonalité de longue durée à cette position a été activée par le script de numérotation désigné dans le signal playcol.

Valeurs par défaut: néant

9.2.2.2.2 Nombre de tentatives

Nom du paramètre: Number of Attempts

Identificateur du paramètre: na (0x0002)

Description: le nombre de tentatives faites par la passerelle média pour collecter une séquence de chiffres valable.

Type: entier

En option: non

Valeurs possibles: valeurs supérieures ou égales à 1.

Valeurs par défaut: néant

9.2.2.2.3 Durée passée

Nom du paramètre: Amount played

Identificateur du paramètre: ap (0x0003)

Description: la durée de l'invite initiale qui a été passée, si cette invite a été interrompue (c'est-à-dire si des chiffres ont été saisis et que le paramètre de lecture ininterrompible était à FALSE), exprimée en dizaines de millisecondes.

Type: entier

En option: oui

Valeurs possibles: valeurs supérieures ou égales à 0.

Valeurs par défaut: néant

9.3 Signaux

9.3.1 Lecture et collecte

Nom du signal: PlayCollect

Identificateur du signal: playcol (0x0002)

Description: lecture d'une annonce ou d'une tonalité (facultative) et collecte des chiffres dtmf ou des paroles introduits par l'utilisateur. Le modèle le plus complet pris en charge par le signal playcol comprend une invite initiale, une nouvelle invite si des chiffres non valables sont saisis, une nouvelle invite différente si l'utilisateur n'a saisi aucun chiffre, une annonce de succès lorsqu'une séquence de chiffres valable a été collectée et une annonce d'échec en cas d'échec de la tentative de collecte de chiffres. Des valeurs par défaut sont attribuées si aucune annonce particulière n'est spécifiée dans ce modèle, comme indiqué dans la description des différents paramètres.

Type de signal: la valeur par défaut est TO

Durée: la valeur par défaut est la valeur mise à disposition pour la terminaison. La durée est utilisée uniquement comme protection contre toute durée excessive de l'ensemble de l'opération de collecte.

9.3.1.1 Paramètres additionnels

9.3.1.1.1 Invite initiale

Nom du paramètre: InitialPrompt

Identificateur du paramètre: ip (0x0001)

Description: l'annonce initiale invitant l'utilisateur à saisir des chiffres DTMF. Elle peut être constituée d'un ou de plusieurs segments audio. Si elle n'est pas spécifiée, la collecte de chiffres commence immédiatement.

Type: chaîne

En option: oui

Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6. La prise en charge d'aspects facultatifs de cette syntaxe, pour ce paramètre d'annonce et les autres, est indiquée par la présence des paquetages associés.

Valeurs par défaut: néant

9.3.1.1.2 Nouvelle invite

Nom du paramètre: Reprompt

Identificateur du paramètre: rp (0x0002)

Description: elle est passée si l'utilisateur a fait une erreur, par exemple s'il a saisi une séquence de chiffres non valable. Cette invite est constituée d'un ou de plusieurs segments audio. La valeur par défaut est l'invite initiale.

Type: chaîne

En option: oui

Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6.

Valeurs par défaut: néant

9.3.1.1.3 Invite en cas d'absence de chiffre

Nom du paramètre: NoDigitsPrompt

Identificateur du paramètre: nd (0x0003)

Description: elle est passée si l'utilisateur n'a saisi aucun chiffre après une invite. Cette invite est constituée d'un ou de plusieurs segments audio. La valeur par défaut est la nouvelle invite.

Type: chaîne

En option: oui

Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6.

Valeurs par défaut: néant

9.3.1.1.4 Annonce de succès

Nom du paramètre: SuccessAnnouncement

Identificateur du paramètre: sa (0x0004)
Description: elle est passée lorsque la collecte de données a abouti. Cette annonce est constituée d'un ou de plusieurs segments audio. Il n'y a pas de valeur par défaut (autrement dit, aucune séquence audio n'est passée si ce paramètre n'est pas spécifié).
Type: chaîne
En option: oui
Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6.
Valeurs par défaut: néant

9.3.1.1.5 Annonce d'échec

Nom du paramètre: FailureAnnouncement
Identificateur du paramètre: fa (0x0005)
Description: elle est passée lorsque toutes les tentatives de saisie de données ont échoué. Cette annonce est constituée d'un ou de plusieurs segments audio. Il n'y a pas de valeur par défaut (autrement dit, aucune séquence audio n'est passée si ce paramètre n'est pas spécifié).
Type: chaîne
En option: oui
Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6.
Valeurs par défaut: néant

9.3.1.1.6 Lecture ininterrompible

Nom du paramètre: NonInterruptiblePlay
Identificateur du paramètre: ni (0x0006)
Description: spécifie si les invites peuvent être interrompues ou non par la saisie de chiffres.
Type: booléen
En option: non
Valeurs possibles: TRUE (les invites ne peuvent pas être interrompues) ou FALSE (les invites peuvent être interrompues par la saisie de chiffres).
Valeur par défaut: FALSE

9.3.1.1.7 Conservation des chiffres

Nom du paramètre: KeepDigits
Identificateur du paramètre: kdg (0x0007)
Description: spécifie le traitement à appliquer aux chiffres détectés pendant la lecture d'une invite ininterrompible. Comme décrit au § 9.5.1, les chiffres saisis pendant une invite ininterrompible seront accumulés si ce paramètre est mis à TRUE.
Type: booléen
En option: oui

Valeurs possibles: TRUE ou FALSE. La valeur par défaut est FALSE (les chiffres détectés pendant une invite ininterrompue sont ignorés).

Valeur par défaut: FALSE

9.3.1.1.8 Effacement de la mémoire tampon de chiffres

Nom du paramètre: ClearDigitBuffer

Identificateur du paramètre: cb (0x0008)

Description: si ce paramètre est mis à TRUE, la passerelle média efface la mémoire tampon de collecte de chiffres avant de passer une invite, quelle qu'elle soit.

Type: booléen

En option: non

Valeurs possibles: TRUE ou FALSE.

Valeur par défaut: FALSE

9.3.1.1.9 Nombre maximal de tentatives

Nom du paramètre: MaxAttempts

Identificateur du paramètre: mxatt (0x0009)

Description: le nombre maximal de tentatives offert à l'utilisateur pour saisir une séquence de chiffres valable.

Type: entier

En option: non

Valeurs possibles: valeurs supérieures ou égales à 1.

Valeur par défaut: 1

9.3.1.1.10 Script de numérotation

Nom du paramètre: DigitMap

Identificateur du paramètre: dm (0x000a)

Description: le nom du script de numérotation qui est actif au niveau de la terminaison.

Type: chaîne

En option: non

Valeurs possibles: pour le codage textuel, toute chaîne correspondant à la production NAME. L'équivalent pour le codage binaire serait la production Name de l'Annexe A/H.248.1, mais un Name est un ensemble arbitraire de 16 bits et ne constitue pas nécessairement un caractère UTF-8 admis. Le nom binaire du script de numérotation doit donc être converti en une chaîne de quatre caractères hexadécimaux avant d'être inséré dans le paramètre de script de numérotation.

Valeurs par défaut: néant

9.3.1.1.11 Vitesse

Nom du paramètre: Speed

Identificateur du paramètre: sp (0x000b)

Description: la vitesse relative de lecture d'annonce exprimée sous la forme d'une différence positive (plus rapide) ou négative (plus lente) en pourcentage par rapport à la vitesse normale de lecture. La vitesse réelle de lecture exprimée en pourcentage de la vitesse normale est égale à la valeur de ce paramètre plus 100.

Type: entier

En option: oui

Valeurs possibles: valeurs supérieures ou égales à -99.

Valeur par défaut: 0

9.3.1.1.12 Volume

Nom du paramètre: Volume

Identificateur du paramètre: vl (0x000c)

Description: le volume relatif de lecture d'annonce exprimé sous la forme d'une différence positive (plus fort) ou négative (plus faible) en décibels par rapport au volume normal de lecture.

Type: entier

En option: oui

Valeurs possibles: dépendent de l'implémentation.

Valeur par défaut: 0

9.3.1.1.13 Décalage

Nom du paramètre: Offset

Identificateur du paramètre: off (0x000d)

Description: spécifie l'endroit dans l'invite initiale où commencer à la passer. Un décalage positif est un décalage vers l'avant à partir du début de l'invite. Un décalage négatif est un décalage vers l'arrière à partir de la fin de l'invite. Les décalages sont exprimés en dizaines de millisecondes.

Les décalages sont utiles lorsque le contrôleur MGC contrôle la collecte de chiffres à un niveau atomique (c'est-à-dire en utilisant un script de numérotation très simple et en utilisant le signal aasdc/playcol pour passer des invites). Exemple d'application: l'utilisateur appuie sur une touche DTMF, le signal playcol détermine le chiffre correspondant et envoie un événement de succès de la lecture et de la collecte au contrôleur MGC – événement qui inclut la valeur du chiffre et la durée de l'invite déjà passée – et le contrôleur MGC décide d'ignorer la touche et indique au serveur audio de reprendre la lecture à l'endroit où elle a été interrompue. Une autre application consiste à permettre à l'utilisateur d'aller plus en avant ou plus en arrière dans une invite.

Type: entier

En option: non

Valeurs possibles: 0, valeurs positives, valeurs négatives. La valeur absolue ne peut pas dépasser la durée de l'invite initiale.

Valeur par défaut: 0

9.3.1.1.14 Touche de redémarrage

Nom du paramètre: RestartKey

Identificateur du paramètre: rsk (0x000e)

Description: définit une séquence comprenant une touche de commande suivie facultativement d'une ou de plusieurs touches. Cette séquence a pour objet: d'éliminer les éventuels chiffres collectés jusqu'au moment où la séquence de commande est lancée, de repasser l'invite et de reprendre la collecte des chiffres. L'utilisation de cette touche ne constitue pas une tentative de saisie de séquence d'utilisateur (autrement dit, elle n'est pas comptée dans le nombre de tentatives spécifié par le paramètre de nombre maximal de tentatives). Ce paramètre est traité localement par le serveur audio et n'est pas retourné au contrôleur MGC.

Type: chaîne

En option: oui

Valeurs possibles: une séquence d'un ou de plusieurs caractères de l'ensemble 0-9, A-D ou a-d, * et # représentant des chiffres DTMF.

Valeurs par défaut: néant

9.3.1.1.15 Touche de nouvelle saisie

Nom du paramètre: ReinputKey

Identificateur du paramètre: rik (0x000f)

Description: définit une séquence comprenant une touche de commande suivie facultativement d'une ou de plusieurs touches. Cette séquence a pour objet: d'éliminer les éventuels chiffres collectés jusqu'au moment où la séquence de commande est lancée et de reprendre la collecte des chiffres. L'utilisation de cette touche ne constitue pas une tentative de saisie de séquence d'utilisateur (autrement dit, elle n'est pas comptée dans le nombre de tentatives spécifié par le paramètre de nombre maximal de tentatives). Ce paramètre est traité localement par le serveur audio et n'est pas retourné au contrôleur MGC.

Type: chaîne

En option: oui

Valeurs possibles: une séquence d'un ou de plusieurs caractères de l'ensemble 0-9, A-D ou a-d, * et # représentant des chiffres DTMF.

Valeurs par défaut: néant

9.3.1.1.16 Touche de retour

Nom du paramètre: ReturnKey

Identificateur du paramètre: rtk (0x0010)

Description: définit une séquence comprenant une touche de commande suivie facultativement d'une ou de plusieurs touches. Cette séquence a pour objet: d'achever la tentative de collecte en cours et de renvoyer la séquence au contrôleur MGC. Pendant un enregistrement, tous les chiffres à l'exception des touches de redémarrage, de nouvelle saisie et de retour (si elles sont définies) sont ignorés et introduits dans l'enregistrement.

Type: chaîne

En option: oui

Valeurs possibles: une séquence d'un ou de plusieurs caractères de l'ensemble 0-9, A-D ou a-d, * et # représentant des chiffres DTMF.

Valeurs par défaut: néant

9.3.1.1.17 Itérations

Nom du paramètre: Iterations

Identificateur du paramètre: it (0x0011)

Description: nombre maximal de fois qu'une annonce doit être passée. Identique au paramètre "Iterations" dans le signal "Play" du paquetage de base de serveur audio évolué. Ce paramètre est aussi nécessaire dans le signal "PlayCollect" puisque la partie INAP ne fait aucune différence entre une simple annonce et une séquence d'annonces/d'invites.

Type: entier

En option: oui

Valeurs possibles: voir le signal "Play" dans le paquetage aasb.

Valeur par défaut: 1

9.3.1.1.18 Intervalle

Nom du paramètre: Interval

Identificateur du paramètre: iv (0x0012)

Description: intervalle de silence à insérer entre lectures itératives, exprimé en dizaines de millisecondes. Identique au paramètre "Interval" dans le signal "Play" du paquetage de base de serveur audio évolué. Ce paramètre est aussi nécessaire dans le signal "PlayCollect" puisque la partie INAP ne fait aucune différence entre une simple annonce et une séquence d'annonces/d'invites.

Type: entier

En option: oui

Valeurs possibles: voir le signal "Play" dans le paquetage aasb.

Valeurs par défaut: néant

9.3.1.1.19 Touche de fin de saisie

Nom du paramètre: EndInputKey

Identificateur du paramètre: eik(0x0013)

Description: ce paramètre indique le chiffre employé pour signaler la fin de la saisie. Lorsque le nombre maximal de chiffres est égal au nombre minimal de chiffres, la touche de fin de saisie (peut être présente mais) n'a pas de signification. Ce paramètre peut comporter un ou deux chiffres. Lorsque le nombre maximal de chiffres dépasse le nombre minimal de chiffres, ce qui suit s'applique:

si la touche de fin de saisie n'est pas présente, la fin de la saisie est indiquée:

- lorsque la temporisation entre les chiffres expire;
- lorsque le nombre de chiffres valables reçus est égal au nombre maximal de chiffres.

Si la touche de fin de saisie est présente, la fin de la saisie est indiquée:

- lorsque la temporisation entre les chiffres expire;
- lorsque le chiffre de fin de saisie est reçu;
- lorsque le nombre de chiffres valables reçus est égal au nombre maximal de chiffres.

Si la temporisation entre les chiffres expire ou que la touche de fin de saisie est activée et que le nombre de chiffres valables reçus est inférieur au nombre minimal de chiffres, la saisie est spécifiée comme étant erronée.

Ce paramètre correspond au paramètre de la partie INAP 'endOfReplyDigit'.

Type: chaîne d'octets (dimension (1..2))

En option: oui

Valeurs possibles: dépendent de l'implémentation.

Valeurs par défaut: néant

9.3.1.1.20 Inclusion de touche de fin de saisie

Nom du paramètre: IncludeEndInputKey

Identificateur du paramètre: iek (0x0014)

Description: le paramètre 'EndInputKey' n'est par défaut pas inclus dans les chiffres collectés renvoyés à l'agent d'appel. Si ce paramètre est mis à TRUE, le paramètre 'EndInputKey' sera renvoyé à l'agent d'appel avec les chiffres collectés.

Type: booléen

En option: oui

Valeurs possibles: TRUE ou FALSE.

Valeur par défaut: FALSE

9.3.1.1.21 Information vocale

Nom du paramètre: VoiceInformation

Identificateur du paramètre: vi (0x0015)

Description: spécifie comment la fonction de relais de liaison d'accès (ARF, *access link relay function*) accepte la parole. La valeur par défaut "dtmfonly" (DTMF seulement) indique que la fonction ARF n'accepte que les chiffres DTMF. La valeur "voiceonly" (parole seulement) indique que la fonction ARF n'accepte que la parole. La valeur "dtmfandvoice" (DTMF et parole) indique que la fonction ARF accepte les chiffres DTMF et la parole et communique tant les chiffres DTMF que les chiffres prononcés. La valeur "dtmfkills" (le mode DTMF supprime la voix) indique que la fonction ARF accepte les deux modes. Lorsqu'un chiffre DTMF est reconnu, tous les chiffres collectés par reconnaissance de la parole sont écartés et aucune parole n'est ultérieurement acceptée.

Ce paramètre correspond au paramètre de la partie INAP "voiceInformation".

Type: énumération
En option: non
Valeurs possibles: – "dtmfonly" (0),
– "voiceonly" (1),
– "dtmfandvoice" (2),
– "dtmfkills" (3)
Valeur par défaut: "dtmfonly"

9.3.1.1.22 Retour vocal

Nom du paramètre: VoiceBack
Identificateur du paramètre: vc (0x0016)
Description: spécifie comment des chiffres reconnus sont repassés à l'utilisateur en cas de reconnaissance de la parole. La valeur "novoiceback" (pas de retour vocal) indique que les chiffres ne sont pas repassés. La valeur "stepbystep" (pas à pas) indique que les chiffres sont repassés dès qu'ils sont reconnus. La valeur "atend" (à la fin) indique que les chiffres sont repassés à l'utilisateur à la fin de la saisie. La valeur "arfcontrolled" (contrôlé par la fonction ARF) indique que la méthode de retour vocal est contrôlée par la fonction ARF. La valeur "vbDTMF" (retour vocal des chiffres DTMF) indique que seuls les chiffres DTMF sont repassés.

Ce paramètre correspond au paramètre de la partie INAP "voiceBack".

Type: énumération
En option: oui
Valeurs possibles: – "novoiceback" (0),
– "stepbystep" (1),
– "atend" (2),
– "arfcontrolled" (3),
– "vbDTMF" (4)
Valeur par défaut: "novoiceback"

9.3.1.1.23 Temporisation de l'invite de la partie INAP

Nom du paramètre: InapPromptTimer
Identificateur du paramètre: ipt (0x0017)

Description: temps maximal de lecture et de lecture éventuelle d'une annonce ou d'une tonalité. Spécifié en centaines de millisecondes. Pas de valeurs par défaut. En tant que partie d'un signal PlayCollect, ce paramètre spécifie la durée de l'invite initiale. Dans le cas d'une tonalité, le paramètre *InapPromptTimer* est obligatoire. Dans le cas d'une annonce, le paramètre *InapPromptTimer* est facultatif et peut s'ajouter aux paramètres *Iterations* et *Interval*.

Le traitement particulier suivant doit être envisagé:

- lorsque le paramètre *InapPromptTimer* est égal à zéro et que le paramètre *Iterations* n'est pas spécifié, cela veut dire que l'invite initiale doit être lue indéfiniment;
- lorsque les paramètres *InapPromptTimer* et *Iterations* sont spécifiés dans le même signal, cela veut dire que la fin de l'invite initiale a lieu à la fin de la temporisation de l'invite de la partie INAP ou à la fin des itérations, selon que l'une ou l'autre se produit la première.

Le paramètre *InapPromptTimer* inclut tous les intervalles spécifiés.

Type: entier

En option: oui

Valeurs possibles: de 0 à plusieurs dizaines de secondes

Valeurs par défaut: néant

9.4 Statistiques

Néant.

9.5 Procédures

Afin d'utiliser efficacement le signal de lecture et de collecte, le contrôleur MGC doit activer l'événement de succès de la lecture et de la collecte. Il devrait aussi activer l'événement d'échec d'opération audio si des détails sur le motif de l'échec sont souhaités. (Dans le cas contraire et si une notification d'échec du signal est requise, on peut utiliser l'événement d'achèvement de signal du paquetage générique.)

En pratique, le contrôleur MGC fournira un script de numérotation qui spécifie entièrement une ou plusieurs séquences valables pouvant être saisies par l'utilisateur. Ainsi, les capacités de la passerelle média à filtrer les séquences de commande et à relancer automatiquement une invite sont entièrement utilisées.

Autre utilisation pratique du signal, suggérée dans la description du paramètre de décalage: le contrôleur MGC spécifie un script de numérotation auquel n'importe quelle touche DTMF est conforme, reçoit les chiffres un par un et redémarre le signal de lecture et de collecte avec un décalage égal à la durée déjà passée. Si l'échange de messages entre la passerelle média et le contrôleur MGC est suffisamment rapide, l'utilisateur entend l'invite initiale comme une séquence audio pratiquement continue. Le seul intérêt qu'il y a à utiliser le signal de lecture et de collecte au lieu du signal *aasb/play* dans ce cas est la possibilité de passer automatiquement une invite en cas d'absence de chiffres.

Si le signal de lecture et de collecte est invoqué avec le paramètre de type de signal mis à TO (la valeur par défaut), la passerelle média doit interpréter le paramètre de durée comme limitant la durée totale de l'opération de collecte de chiffres, et non la durée de la lecture de l'invite initiale. Si la temporisation expire avant qu'une séquence de chiffres valable soit collectée, le signal s'achève

avec le motif "expiration de temporisation" et un événement d'échec d'opération audio est généré avec le code de retour 617. Si le paramètre de type de signal est mis à BR ou OO, l'opération se poursuit jusqu'à ce qu'elle soit interrompue par un événement ou par une modification du descripteur de signal, ou jusqu'à ce qu'elle s'achève, conduisant à un succès ou à un échec. Dans le cas de l'achèvement, le signal s'achève "normalement".

Une séquence de commande comprend une touche de commande (ou d'échappement) suivie facultativement par une ou plusieurs touches. Une application qui définit plusieurs séquences de commande utilisera généralement la même touche de commande (par exemple *) pour toutes les séquences de commande. Chaque séquence de commande doit être unique par rapport à toutes les autres. Les applications peuvent prendre en charge des séquences de commande autres que <touche de redémarrage>, <touche de nouvelle saisie> et <touche de retour>.

Pour permettre à la passerelle média de traiter les séquences de commande, les applications doivent choisir une touche de commande qui ne figure dans aucun script de numérotation. Si une touche de commande est rencontrée, le traitement par rapport au script de numérotation s'arrêtera et les touches subséquentes seront traitées comme une séquence de commande jusqu'à ce qu'une séquence de commande soit reconnue ou jusqu'à ce qu'il soit clair qu'une séquence de commande ne peut pas être reconnue, auquel cas l'erreur 618 "détection d'une séquence de commande non valable" est renvoyée.

Le paramètre de nombre de tentatives renvoyé dans l'événement de succès de la lecture et de la collecte peut servir à améliorer la mise à disposition de la fonction de lecture et de collecte.

Le contrôleur MGC doit veiller à lever le drapeau KeepActive pour tous les événements qu'il active et qui ne sont pas censés interrompre l'opération de lecture et de collecte. Cela s'applique tout particulièrement si le contrôleur MGC active un événement pour chaque chiffre ou un événement d'achèvement de script de numérotation. Cette activation n'est pas nécessaire pour que l'opération de lecture et de collecte aboutisse.

9.5.1 Modèle de traitement des chiffres issus de l'opération de lecture et de collecte

Le traitement des chiffres collectés repose sur un script de numérotation actif au niveau de la terminaison et désigné par le paramètre correspondant du signal playcol. Le modèle de traitement des chiffres est analogue à celui qui s'applique dans le cas des scripts de numérotation ordinaires, à deux exceptions près:

- la possibilité de redémarrer le processus avec de nouvelles invites sans intervention du contrôleur MGC;
- la possibilité de détecter et d'exécuter des séquences de commande.

Par défaut, le serveur audio prend en charge une fonctionnalité permettant de prendre de l'avance. Autrement dit, la détection et l'accumulation de chiffres dans la mémoire tampon de collecte de chiffres pour la comparaison avec les séquences de commande et avec le script de numérotation commencent dès que la commande playcol devient active. On peut désactiver cette fonctionnalité en spécifiant que la collecte de chiffres commence uniquement après la lecture de l'invite initiale.

La logique détaillée de collecte des chiffres est précisée ci-après, les références aux paramètres du signal playcol étant mises entre crochets angulaires <> pour qu'elles ressortent bien. Cette logique repose sur deux mémoires tampons logiques: une mémoire tampon de collecte de chiffres qui reçoit tous les chiffres composés par l'utilisateur, qu'ils fassent partie de la séquence de chiffres saisie par l'utilisateur ou d'une séquence de commande, et une mémoire tampon d'accumulation de la chaîne alors composée par rapport au script de numérotation. Le contenu de la mémoire tampon de collecte de chiffres ne peut dépasser un chiffre que si la saisie de chiffres est autorisée pendant la lecture d'une invite ininterrompue ou pendant l'accumulation d'une séquence de commande à plusieurs chiffres, mais la logique traite le cas général.

- 1) La commande playcol devient active. Le nombre de tentatives vaut zéro. Mettre "l'invite courante" sur <invite initiale>. Effacer la mémoire tampon de collecte de chiffres (qui recevra tous les chiffres, qu'ils fassent partie de la séquence de chiffres saisie par l'utilisateur ou d'une séquence de commande).
- 2) Boucle de collecte. Incrémenter le nombre de tentatives. Initialiser le traitement par rapport au script de numérotation. Si <effacement de la mémoire tampon de chiffres> est à TRUE, effacer la mémoire tampon de collecte de chiffres.
- 3) Procéder selon l'un des trois cas suivants selon qu'il convient:
 - a) <lecture ininterrompible > est à TRUE:
commencer la lecture de l'invite courante. Si <conservation des chiffres> est à TRUE, conserver le contenu courant de la mémoire tampon de collecte de chiffres et autoriser (la poursuite de) l'accumulation des chiffres pendant la lecture. Si <conservation des chiffres> est à FALSE, effacer la mémoire tampon de collecte de chiffres et ignorer les chiffres détectés pendant la lecture. Lorsque la lecture de l'invite courante est achevée, aller à l'étape 4).
 - b) <lecture ininterrompible > est à FALSE et la mémoire tampon de collecte de chiffres n'est pas vide:
Ne pas passer l'invite courante. Procéder immédiatement au traitement des chiffres (étape 7).
 - c) <lecture ininterrompible > est à FALSE et la mémoire tampon de collecte de chiffres est vide:
commencer la lecture de l'invite courante. En cas de détection de chiffres pendant la lecture de l'invite, arrêter immédiatement la lecture et procéder au traitement des chiffres (étape 7). Dans le cas contraire, aller à l'étape suivante.
- 4) La lecture de l'invite courante s'achève. Commencer l'accumulation des chiffres si elle n'a pas encore démarré. Déclencher la temporisation au premier chiffre pour le script de numérotation.
- 5) En cas de détection de chiffres, procéder au traitement des chiffres (étape 7). Dans le cas contraire, aller à l'étape suivante.
- 6) Vérifier le nombre de tentatives. S'il est égal à <nombre maximal de tentatives>, passer l'<annonce d'échec> si une telle annonce a été spécifiée, sortir et générer un événement d'échec d'opération audio avec le code de retour 620 "Absence de chiffre". Sinon, mettre l'annonce courante à <invite en cas d'absence de chiffre> et retourner à l'étape 2).
- 7) Traitement des chiffres. Traiter les chiffres accumulés dans la mémoire tampon de collecte de chiffres et les chiffres suivants à mesure qu'ils arrivent, en les comparant d'abord aux séquences de commande puis au script de numérotation. Concernant le traitement par rapport au script de numérotation, les règles de temporisation du § 7.1.14/H.248.1 s'appliquent. Si une séquence de commande <touche de redémarrage> est reconnue, aller à l'étape 8). Si une séquence de commande <touche de nouvelle saisie> est reconnue, aller à l'étape 9). Si une séquence de commande <touche de retour> est reconnue, aller à l'étape 10). En cas d'échec de mise en correspondance avec le script de numérotation (aucune séquence ne correspond entièrement), aller à l'étape 11). Enfin, si la mise en correspondance avec le script de numérotation aboutit (une séquence correspond entièrement), passer l'<annonce de succès> si une telle annonce a été spécifiée, générer un événement de succès de la lecture et de la collecte avec les chiffres collectés et sortir.
- 8) La séquence de commande <touche de redémarrage> est reconnue. Décrémenter le nombre de tentatives, conserver le contenu de la mémoire tampon d'accumulation des chiffres à partir de la séquence de commande <touche de redémarrage>, mettre "l'invite courante" sur <invite initiale> et retourner à l'étape 2).

- 9) La séquence de commande <touche de nouvelle saisie> est reconnue. Réinitialiser le mappage avec le script de numérotation, conserver le contenu de la mémoire tampon d'accumulation des chiffres à partir de la séquence de commande <touche de nouvelle saisie> et retourner à l'étape 7).
- 10) La séquence de commande <touche de retour> est reconnue. Passer l'<annonce de succès> si une telle annonce a été spécifiée, générer un événement de succès de la lecture et de la collecte avec la séquence de commande <touche de retour> à la place des chiffres collectés et sortir.
- 11) Echec de la mise en correspondance avec le script de numérotation. Vérifier le nombre de tentatives. S'il est égal à <nombre maximal de tentatives>, passer l'<annonce d'échec> si une telle annonce a été spécifiée, sortir et générer un événement d'échec d'opération audio avec le code de retour 619 "Nombre maximal de tentatives dépassé". Sinon, mettre l'annonce courante sur <nouvelle invite>, conserver le contenu de la mémoire tampon d'accumulation des chiffres au-delà des chiffres déjà traités (autrement dit en éliminant le chiffre qui "a cassé la séquence") et retourner à l'étape 2).

10 Paquetage d'enregistrement de serveur AAS

Nom du paquetage: AAS recording package

Identificateur du paquetage: aasrec (0x0035)

Description: paquetage d'enregistrement de serveur AAS: étend le paquetage de base de serveur AAS en prenant en charge des signaux et des événements permettant de coordonner la collecte de séquences vocales enregistrées avec la lecture d'invites.

Version: 1

Extension de: aasb (0x0033) version 1

10.1 Propriétés

10.1.1 Durée de vie maximale d'un enregistrement temporaire

Nom de la propriété: Maximum temporary record life

Identificateur de la propriété: maxtrl (0x0003)

Description: détermine la durée de vie maximale d'un enregistrement temporaire, en secondes, après la fin de l'enregistrement. Les enregistrements faits par le signal de lecture et d'enregistrement sont temporaires sauf s'ils ont explicitement été rendus persistants au moyen du signal de transformation en segment persistant. Les enregistrements temporaires sont supprimés à l'expiration de maxtrl ou au moment de la destruction de la terminaison au niveau de laquelle l'enregistrement a été fait, si cette destruction intervient auparavant.

Type: entier

Valeurs possibles: valeurs supérieures ou égales à 1

Valeurs par défaut: néant

Défini dans: TerminationState

Caractéristiques: lecture/écriture

10.2 Evénements

10.2.1 Echech d'opération audio

Nom de l'événement: Audio operation failure

Identificateur de l'événement: audfail (0x0001)

Description: ce paquetage définit les autres valeurs suivantes pour le code de retour renvoyé par l'événement d'échec d'opération audio défini au § 8.2.1. Il est à noter que les codes 617 et 618 sont également pris en charge par le paquetage de collecte de chiffres de serveur AAS.

617 Fin d'opération prématurée. L'opération audio a été interrompue avant son achèvement normal, par la reconnaissance d'un événement avec le drapeau KeepActive non levé, par le remplacement du descripteur de signal sans poursuite du signal ou par l'expiration de la temporisation liée à la durée du signal.

618 Détection d'une séquence de commande non valable.

622 Aucune parole n'a été collectée après un nombre d'invites égal au <nombre maximal de tentatives>.

623 Hors mémoire.

624 Incapacité à supprimer un segment audio temporaire. A l'expiration de maxtrl ou au moment de la destruction de la terminaison, un segment audio enregistré qui n'avait pas été rendu persistant n'a pas pu être supprimé. Dans le cas de l'expiration de temporisation, il se peut que le segment soit utilisé par une autre opération au niveau de la même terminaison.

10.2.1.1 Paramètres du descripteur d'événement

Voir le § 8.2.1.

10.2.1.2 Paramètres du descripteur d'événement observé

Voir le 8.2.1.

10.2.2 Succès de la lecture et de l'enregistrement

Nom de l'événement: PlayRecord success

Identificateur de l'événement: presucc (0x0002)

Description: indique l'aboutissement d'un signal playrec.

10.2.2.1 Paramètres du descripteur d'événement

Néant.

10.2.2.2 Paramètres du descripteur d'événement observé

10.2.2.2.1 Durée passée

Nom du paramètre: Amount played

Identificateur du paramètre: ap (0x0001)

Description: la durée de l'invite initiale qui est passée, si cette invite a été interrompue, en dizaines de millisecondes.

Type: entier
En option: non
Valeurs possibles: valeurs supérieures ou égales à 0
Valeur par défaut: 1

10.2.2.2.2 Nombre de tentatives

Nom du paramètre: Number of attempts
Identificateur du paramètre: na (0x0002)
Description: le nombre de fois que l'utilisateur a été invité à faire un enregistrement.

Type: entier
En option: non
Valeurs possibles: valeurs supérieures ou égales à 1.
Valeur par défaut: 1

10.2.2.2.3 Résultat d'enregistrement

Nom du paramètre: Recording result
Identificateur du paramètre: res (0x0003)
Description: la manière dont le processus d'enregistrement s'est achevé.
Type: énumération
En option: non
Valeurs possibles: "normal" (0): un segment audio temporaire a été enregistré et la fin de la parole a été détectée avant l'expiration de la temporisation de durée d'enregistrement.
"trunc" (1): un segment audio temporaire a été enregistré et il a été tronqué à l'expiration de la temporisation de durée d'enregistrement.
"keyend" (2): la séquence de commande touche de retour a été détectée. Aucune séquence audio enregistrée n'a été conservée.
Valeurs par défaut: néant

10.2.2.2.4 Identificateur d'enregistrement

Nom du paramètre: Recording id
Identificateur du paramètre: ri (0x0004)
Description: un identificateur URI attribué au segment physique enregistré pendant un signal playrec. Ce paramètre n'est renvoyé que si le paramètre d'identificateur d'enregistrement du signal playrec a été mis sur le caractère générique, "\$", correspondant à un identificateur quelconque. Si c'est le cas, le serveur audio attribue un identificateur URI unique, l'associe au segment nouvellement enregistré et le renvoie au contrôleur MGC. S'il est mis fin à l'opération de lecture et d'enregistrement par la séquence de commande touche de retour, l'attribution de l'identificateur URI est annulée et ce paramètre ne doit pas être présent dans la notification d'événement.

Type:	chaîne
En option:	oui
Valeurs possibles:	tout identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est un identificateur URI http://, il ne doit pas avoir de partie query.
Valeurs par défaut:	néant

10.2.2.2.5 Durée d'enregistrement

Nom du paramètre:	Record duration
Identificateur du paramètre:	rdur (0x0005)
Description:	la durée totale du segment audio enregistré, exprimée en dizaines de millisecondes.
Type:	entier
En option:	oui
Valeurs possibles:	valeurs supérieures ou égales à 0. S'il a été mis fin à l'opération par la séquence de commande touche de retour, rdur ne doit pas être présent dans la notification d'événement.
Valeurs par défaut:	néant

10.3 Signaux

10.3.1 Lecture et enregistrement

Nom du signal:	PlayRecord
Identificateur du signal:	playrec (0x0002)
Description:	<p>lecture d'une invite (facultatif) et enregistrement de la parole de l'utilisateur. Le modèle le plus complet pris en charge par le signal playrec est analogue à celui qui est pris en charge par le signal aasdc/playcol, sauf qu'il n'y a pas de reconnaissance de parole non valable. Il y a une invite initiale, une nouvelle invite si l'utilisateur ne parle pas, une annonce de succès qui est passée lorsqu'un enregistrement a pu être collecté et une annonce d'échec qui est passée si la tentative de collecte d'un enregistrement échoue. Des valeurs par défaut sont attribuées si aucune annonce particulière n'est spécifiée dans ce modèle, comme indiqué dans la description des différents paramètres.</p> <p>Les paramètres de temporisation de durée d'enregistrement et d'identificateur d'enregistrement doivent être spécifiés. Tous les autres paramètres sont facultatifs.</p>
Type de signal:	la valeur par défaut est TO
Durée:	la valeur par défaut est 30000 (5 minutes) ou la valeur mise à disposition pour la terminaison.

10.3.1.1 Paramètres additionnels

10.3.1.1.1 Invite initiale

Nom du paramètre:	InitialPrompt
-------------------	---------------

Identificateur du paramètre: ip (0x0001)
Description: l'annonce initiale invitant l'utilisateur à parler pour l'enregistrement. Elle peut être constituée d'un ou de plusieurs segments audio.
Type: chaîne
En option: oui
Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6. La prise en charge d'aspects facultatifs de cette syntaxe, pour ce paramètre d'annonce et les autres, est indiquée par la présence des paquetages associés au niveau de la terminaison. Si ce paramètre n'est pas spécifié, la passerelle média passe immédiatement à la phase d'enregistrement.
Valeurs par défaut: néant

10.3.1.1.2 Invite en cas d'absence de parole

Nom du paramètre: NoSpeechPrompt
Identificateur du paramètre: ns (0x0002)
Description: cette invite est passée si l'utilisateur n'a pas parlé après une invite. Elle est constituée d'un ou de plusieurs segments audio.
Type: chaîne
En option: oui
Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6. La valeur par défaut est l'invite initiale.
Valeurs par défaut: néant

10.3.1.1.3 Annonce de succès

Nom du paramètre: SuccessAnnouncement
Identificateur du paramètre: sa (0x0003)
Description: cette annonce est passée lorsque l'enregistrement a abouti. Elle est constituée d'un ou de plusieurs segments audio.
Type: chaîne
En option: oui
Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6. Aucune annonce n'est passée si ce paramètre n'est pas spécifié.
Valeurs par défaut: néant

10.3.1.1.4 Annonce d'échec

Nom du paramètre: FailureAnnouncement
Identificateur du paramètre: fa (0x0004)
Description: cette annonce est passée lorsque toutes les tentatives d'enregistrement ont échoué. Elle est constituée d'un ou de plusieurs segments audio.
Type: chaîne
En option: oui

Valeurs possibles: toute spécification d'annonce conforme à la syntaxe décrite au § 6. Aucune annonce n'est passée si ce paramètre n'est pas spécifié.

Valeurs par défaut: néant

10.3.1.1.5 Nombre maximal de tentatives

Nom du paramètre: MaxAttempts

Identificateur du paramètre: mxatt (0x0005)

Description: le nombre maximal de fois où l'utilisateur est invité à parler. Les invites résultant de l'utilisation de <touche de redémarrage> ne sont pas incluses. Si le <nombre maximal de tentatives> est atteint, l'<annonce d'échec> est passée si elle est spécifiée et un événement d'échec d'opération audio est généré avec le code de retour 622 "Absence de parole".

Type: entier

En option: oui

Valeurs possibles: valeurs supérieures ou égales à 1.

Valeurs par défaut: néant

10.3.1.1.6 Temporisation avant parole

Nom du paramètre: PreSpeechTimer

Identificateur du paramètre: prt (0x0006)

Description: durée d'attente avant que l'utilisateur ne commence à parler, exprimée en dizaines de millisecondes.

Type: entier

En option: non

Valeurs possibles: valeurs supérieures ou égales à 1.

Valeurs par défaut: néant

10.3.1.1.7 Temporisation après parole

Nom du paramètre: PostSpeechTimer

Identificateur du paramètre: pst (0x0007)

Description: la durée du silence nécessaire après la fin du dernier segment de parole pour pouvoir considérer l'enregistrement comme complet, exprimée en dizaines de millisecondes. Une fois que la temporisation après parole a expiré, la passerelle média passe l'<annonce de succès> si elle a été spécifiée et génère un événement de succès de la lecture et de l'enregistrement indiquant un achèvement normal.

Type: entier

En option: non

Valeurs possibles: valeurs supérieures ou égales à 1.

Valeurs par défaut: néant

10.3.1.1.8 Temporisation de durée d'enregistrement

Nom du paramètre: RecordLengthTimer

Identificateur du paramètre: rlt (0x0008)

Description: la durée maximale admissible de l'enregistrement, les silences avant ou après parole étant exclus. Cette durée est exprimée en dizaines de millisecondes. Si la durée de l'enregistrement dépasse (la temporisation de durée d'enregistrement – la temporisation après parole), la passerelle média passe l'<annonce de succès> si elle a été spécifiée et génère un événement de succès de la lecture et de l'enregistrement indiquant que l'enregistrement est tronqué. La valeur 0 (zéro) signifie que la durée de l'enregistrement n'est pas limitée. L'enregistrement est ouvert et il appartient à l'application de gérer la mémoire utilisée par l'enregistrement.

Type: entier

En option: non

Valeurs possibles: valeurs supérieures ou égales à 0.

Valeurs par défaut: néant

10.3.1.1.9 Identificateur d'enregistrement

Nom du paramètre: RecordingIdentifier

Identificateur du paramètre: rid (0x0009)

Description: spécifie un identificateur URI à attribuer au segment physique qui doit être enregistré par le signal playrec. Si ce paramètre est mis sur le caractère générique CHOOSE, "\$", correspondant à un identificateur à choisir, le serveur audio attribuera l'identificateur URI, l'associera au segment nouvellement enregistré et le renverra à l'agent d'appel avec l'événement d'achèvement d'opération. Ce paramètre est obligatoire.

Type: chaîne

En option: non

Valeurs possibles: soit "\$", soit un identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est une adresse URL http://, il ne doit pas avoir de partie query.

Valeurs par défaut: néant

10.3.1.1.10 Vitesse

Nom du paramètre: Speed

Identificateur du paramètre: sp (0x000a)

Description: la vitesse relative de lecture d'annonce exprimée sous la forme d'une différence positive (plus rapide) ou négative (plus lente) en pourcentage par rapport à la vitesse normale de lecture. La vitesse réelle de lecture exprimée en pourcentage de la vitesse normale est égale à la valeur de ce paramètre plus 100.

Type: entier

En option: oui

Valeurs possibles: valeurs supérieures ou égales à -99.

Valeur par défaut: 0

10.3.1.1.11 Volume

Nom du paramètre:	Volume
Identificateur du paramètre:	vl (0x000b)
Description:	le volume relatif de lecture d'annonce exprimé sous la forme d'une différence positive (plus fort) ou négative (plus faible) en décibels par rapport au volume normal de lecture.
Type:	entier
En option:	oui
Valeurs possibles:	dépendent de l'implémentation.
Valeur par défaut:	0

10.3.1.1.12 Décalage

Nom du paramètre:	Offset
Identificateur du paramètre:	off (0x000c)
Description:	<p>spécifie l'endroit dans l'invite initiale où commencer à la passer. Un décalage positif est un décalage vers l'avant à partir du début de l'invite. Un décalage négatif est un décalage vers l'arrière à partir de la fin de l'invite. Les décalages sont exprimés en dizaines de millisecondes.</p> <p>Les décalages sont utiles pour permettre à l'utilisateur d'aller plus en avant ou plus en arrière dans une invite, notamment lorsque cette invite est en fait un enregistrement d'utilisateur qui est repassé.</p>
Type:	entier
En option:	oui
Valeurs possibles:	0, valeurs positives, valeurs négatives. La valeur absolue ne peut pas dépasser la durée de l'invite initiale.
Valeur par défaut:	0

10.3.1.1.13 Touche de redémarrage

Nom du paramètre:	RestartKey
Identificateur du paramètre:	rsk (0x000d)
Description:	définit une séquence comprenant une touche de commande suivie facultativement d'une ou de plusieurs touches. Cette séquence a pour objet: d'ignorer tout enregistrement qui a été fait jusqu'au moment où la séquence de commande est lancée, de repasser l'invite et de tenter à nouveau de détecter et d'enregistrer de la parole. La nouvelle invite imposée par cette touche n'est pas comptée dans le nombre de tentatives spécifié par le paramètre de nombre maximal de tentatives.
Type:	chaîne
En option:	oui
Valeurs possibles:	une séquence d'un ou de plusieurs caractères de l'ensemble 0-9, A-D ou a-d, * et # représentant des chiffres DTMF. La valeur par défaut est qu'aucune séquence n'est définie (cette valeur peut être remplacée par une autre valeur par la mise à disposition).

Valeurs par défaut: néant

10.3.1.1.14 Touche de nouvelle saisie

Nom du paramètre: ReinputKey

Identificateur du paramètre: rik (0x000e)

Description: définit une séquence comprenant une touche de commande suivie facultativement d'une ou de plusieurs touches. Cette séquence a pour objet: d'ignorer tout enregistrement qui a été fait jusqu'au moment où la séquence de commande est lancée et de tenter à nouveau de détecter et d'enregistrer de la parole sans passer de nouvelle invite.

Type: chaîne

En option: oui

Valeurs possibles: une séquence d'un ou de plusieurs caractères de l'ensemble 0-9, A-D ou a-d, * et # représentant des chiffres DTMF.

Valeur par défaut: néant

10.3.1.1.15 Touche de retour

Nom du paramètre: ReturnKey

Identificateur du paramètre: rtk (0x000f)

Description: définit une séquence comprenant une touche de commande suivie facultativement d'une ou de plusieurs touches. Cette séquence a pour objet: d'achever la tentative d'enregistrement en cours et de supprimer toute parole enregistrée jusque là, de passer l'<annonce de succès> si elle est spécifiée et de générer un événement de succès de la lecture et de l'enregistrement indiquant que l'opération s'est achevée avec la <touche de retour>. Pendant un enregistrement, tous les chiffres à l'exception des touches de redémarrage, de nouvelle saisie et de retour (si elles sont définies) sont ignorés et introduits dans l'enregistrement.

Type: chaîne

En option: oui

Valeurs possibles: une séquence d'un ou de plusieurs caractères de l'ensemble 0-9, A-D ou a-d, * et # représentant des chiffres DTMF.

Valeurs par défaut: néant

10.3.2 Transformation en segment persistant

Non du signal: Make persistent

Identificateur du signal: makepers (0x0003)

Description: transforme le segment audio temporaire identifié par l'identificateur URI donné en segment audio persistant. Si cela n'est pas fait, le segment audio temporaire sera supprimé au moment de la destruction de la terminaison au niveau de laquelle il a été créé ou au moment de l'expiration de la durée de vie fixée par la propriété aasrc/maxtrl.

Type de signal: la valeur par défaut est BR

Durée: non applicable pour le type par défaut.

10.3.2.1 Paramètres additionnels

10.3.2.1.1 Identificateur d'enregistrement

Nom du paramètre:	Recording Identifier
Identificateur du paramètre:	rid (0x0001)
Description:	identifie le segment audio qui doit être rendu persistant.
Type:	chaîne
En option:	non
Valeurs possibles:	un identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est une adresse URL http://, il ne doit pas avoir de partie query.
Valeurs par défaut:	néant

10.4 Statistiques

Néant.

10.5 Procédures

La logique de l'enregistrement est beaucoup plus simple que la logique de la collecte de chiffres. Le nombre de tentatives commence à zéro. Chaque fois qu'une invite est passée, le nombre de tentatives est incrémenté. Une nouvelle invite est lancée si aucune parole de l'utilisateur n'est détectée pendant l'intervalle fixé par la <temporisation avant parole>. La fin de la parole est reconnue lorsque l'utilisateur s'arrête de parler pendant une durée donnée par la <temporisation après parole>.

Si la passerelle média reconnaît une séquence de commande DTMF, elle procède comme suit:

- si une <touche de redémarrage> est détectée, toute séquence audio enregistrée est supprimée, l'invite initiale est repassée sans que le compteur de tentatives ne soit incrémenté et la tentative en cours est reprise;
- si une <touche de nouvelle saisie> est détectée, toute séquence audio enregistrée est supprimée et la tentative en cours est reprise sans qu'une nouvelle invite ne soit lancée;
- si une <touche de retour> est détectée, toute séquence audio enregistrée est supprimée. Si la passerelle média a attribué un identificateur URI identifiant le segment audio enregistré, l'attribution de cet identificateur URI est annulée. L'<annonce de succès>, s'il en existe une, est passée et un événement de succès de la lecture et de l'enregistrement est généré indiquant que l'opération s'est achevée avec la <touche de retour>.

Les applications peuvent prendre en charge des séquences de commande autres que <touche de redémarrage>, <touche de nouvelle saisie> et <touche de retour>.

Lorsqu'un échec d'opération audio est généré par le signal de lecture et d'enregistrement, toute parole enregistrée est supprimée et l'attribution par la passerelle média de tout identificateur URI comme identificateur de segment est annulée.

Les segments audio persistants sont globaux du point de vue de la passerelle média. Ainsi, un segment persistant créé au niveau d'une terminaison peut être cité en référence dans un signal invoqué au niveau d'une autre terminaison. Toutefois, les segments audio temporaires ne peuvent être cités en référence que dans des opérations au niveau de la terminaison où ils ont été enregistrés.

Tout échec du signal de transformation en segment persistant doit être signalé par un code d'erreur approprié dans la réponse à la transaction qui a invoqué ce signal. Autrement dit, la réponse ne doit

pas être renvoyée au contrôleur MGC tant que le résultat de l'opération de transformation en segment persistant n'est pas connu.

11 Paquetage de gestion de segments de serveur audio évolué

Nom du paquetage: Advanced audio server segment management package

Identificateur du paquetage: aassm (0x0036)

Description: le paquetage de gestion de segments de serveur audio évolué prend en charge un mécanisme permettant de remplacer, de rétablir et de supprimer des segments audio persistants. Ce paquetage est défini au niveau d'une terminaison spéciale de commande de segment logique et non au niveau des différentes terminaisons où les annonces peuvent être passées. Il repose sur la nécessité d'avoir un espace de noms de segments audio qui soit global du point de vue de la passerelle média.

Le contrôleur MGC remplace un segment physique mis à disposition en spécifiant un autre segment physique persistant. L'identificateur URI du segment physique mis à disposition correspondra alors au segment physique persistant remplaçant. Ce segment peut ensuite être supprimé et le segment audio initial mis à disposition peut être rétabli.

Un segment physique mis à disposition peut être remplacé plusieurs fois. Dans ce cas, l'identificateur URI du segment physique mis à disposition se rapporte au dernier segment physique remplaçant. Lorsque ce segment est supprimé, le segment physique initial mis à disposition est rétabli, même s'il a été remplacé plusieurs fois.

Le remplacement de segment peut être utilisé lorsqu'un message d'accueil standard est passé à tous les clients appelant un magasin de détail. Parfois, le gestionnaire du magasin peut souhaiter appeler un numéro spécial et enregistrer un message d'accueil temporaire qui remplace le message d'accueil standard, par exemple un message d'accueil qui annonce des soldes ou éventuellement un certain type de message d'accueil saisonnier. Lorsque le gestionnaire ne veut plus de ce message d'accueil, il peut appeler le numéro spécial, annuler le message d'accueil temporaire et rétablir le message d'accueil standard.

Ce paquetage ne repose pas sur le paquetage de base de serveur audio évolué, il ne l'étend donc pas.

Version: 1

Extension de: néant

11.1 Propriétés

11.1.1 Nom de terminaison de commande de segment de serveur AAS

Nom de la propriété: AAS segment control termination name

Identificateur de la propriété: ctlnam (0x0001)

Description: nom de la terminaison de commande de segment de serveur AAS, s'il en existe une, prise en charge par la passerelle média.

Type: identificateur de terminaison de type ASN.1 ou identificateur de terminaison de type ABNF, suivant le codage utilisé.

Valeurs possibles:	valeurs mises à disposition dans la passerelle média. La valeur NE DOIT PAS contenir de caractère générique.
Valeurs par défaut:	néant
Définies dans:	TerminationState sur ROOT.
Caractéristiques:	lecture uniquement.

11.2 Evénements

Néant.

11.3 Signaux

Le paquetage de gestion de segments de serveur audio évolué prend en charge trois nouveaux signaux.

11.3.1 Suppression de segment persistant

Nom du signal:	Delete persistent
Identificateur du signal:	delpers (0x0001)
Description:	supprime un segment audio persistant identifié.
Type de signal:	BR
Durée:	non applicable.

11.3.1.1 Paramètres additionnels

11.3.1.1.1 Identificateur de segment

Nom du paramètre:	Segment Identifier
Identificateur du paramètre:	sid (0x0001)
Description:	identifie le segment audio qui doit être supprimé.
Type:	chaîne
En option:	non
Valeurs possibles:	un identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est un identificateur URI http://, il ne doit pas avoir de partie query.
Valeurs par défaut:	néant

11.3.2 Remplacement de segment audio

Nom du signal:	Override audio
Identificateur du signal:	override (0x0002)
Description:	superpose un segment audio persistant différent sur le segment audio mis à disposition spécifié. Si un segment est déjà superposé, le nouveau segment le remplace.
Type de signal:	BR
Durée:	non applicable.

11.3.2.1 Paramètres additionnels

11.3.2.1.1 Segment cible

Nom du paramètre:	Target Segment
Identificateur du paramètre:	tgtsid (0x0001)
Description:	identifie le segment qui doit être remplacé temporairement par un nouveau segment.
Type:	chaîne
En option:	non
Valeurs possibles:	un identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est un identificateur URI http://, il ne doit pas avoir de partie query.
Valeurs par défaut:	néant

11.3.2.1.2 Segment remplaçant

Nom du paramètre:	Overriding segment
Identificateur du paramètre:	oversid (0x0002)
Description:	identifie le segment qui doit être passé à la place du segment cible.
Type:	chaîne
En option:	non
Valeurs possibles:	un identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est un identificateur URI http://, il ne doit pas avoir de partie query.
Valeurs par défaut:	néant

11.3.3 Rétablissement de segment audio

Nom du signal:	RestoreAudio
Identificateur du signal:	restore (0x0003)
Description:	supprime un segment superposé précédemment imposé, de sorte que les références ultérieures au segment cible entraînent la lecture du contenu mis à disposition initialement.
Type de signal:	BR
Durée:	non applicable.

11.3.3.1 Paramètres additionnels

11.3.3.1.1 Segment cible

Nom du paramètre:	Target segment
Identificateur du paramètre:	tgtsid (0x0001)
Description:	identifie le segment sur lequel tout segment superposé doit être supprimé.
Type:	chaîne
En option:	non

Valeurs possibles: un identificateur de segment physique conforme à la syntaxe du § 6.2.5.2. Si l'identificateur est un identificateur URI http://, il ne doit pas avoir de partie query.

Valeurs par défaut: néant

11.4 Statistiques

Néant.

11.5 Procédures

La réponse à une transaction qui invoque des signaux de ce paquetage ne doit pas être renvoyée tant que le résultat des opérations invoquées n'est pas connu. Si une erreur se produit alors et si l'un des codes d'erreur définis au § 7 est applicable, ce code devrait être utilisé dans le descripteur d'erreur renvoyé.

SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
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	Gestion des télécommunications y compris le RGT et maintenance des réseaux
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, communication entre systèmes ouverts et sécurité
Série Y	Infrastructure mondiale de l'information, protocole Internet et réseaux de prochaine génération
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication