



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

T.135

(02/98)

SÉRIE T: TERMINAUX DES SERVICES TÉLÉMATIQUES

**Transactions entre système de réservation et
utilisateurs dans les conférences T.120**

Recommandation UIT-T T.135

(Antérieurement Recommandation du CCITT)

RECOMMANDATIONS UIT-T DE LA SÉRIE T
TERMINAUX DES SERVICES TÉLÉMATIQUES



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

RECOMMANDATION UIT-T T.135

TRANSACTIONS ENTRE SYSTEME DE RESERVATION ET UTILISATEURS DANS LES CONFERENCES T.120

Résumé

La présente Recommandation présente un protocole qui permet à une application de réservation implantée dans un terminal utilisateur de communiquer électroniquement avec le système de réservation d'un fournisseur de service, afin de réserver des ressources pour des conférences multimédias, de modifier ou d'annuler des réservations antérieures et d'exécuter d'autres transactions associées. Le modèle architectural défini par la présente Recommandation est conçu pour des transactions dans la bande, c'est-à-dire pour des transactions entre un système de réservation et un terminal utilisateur conforme à la Recommandation T.120 en conférence active.

Source

La Recommandation UIT-T T.135, élaborée par la Commission d'études 16 (1997-2000) de l'UIT-T, a été approuvée le 6 février 1998 selon la procédure définie dans la Résolution n° 1 de la CMNT.

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.

La Conférence mondiale de normalisation des télécommunications (CMNT), qui se réunit tous les quatre ans, détermine les thèmes d'études 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 n° 1 de la CMNT.

Dans certains secteurs de la technologie 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.

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 1998

Droits de reproduction réservés. Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'UIT.

TABLE DES MATIÈRES

		Page
1	Domaine d'application	1
2	Références normatives	1
3	Définitions	2
4	Abréviations	4
5	Conventions	4
6	Introduction	5
7	Modèle T.135	6
7.1	Modèle du système	6
7.2	Mode de fonctionnement	8
	7.2.1 Identification des applications de réservation.....	9
	7.2.2 Connexions de réservation.....	10
	7.2.3 Transactions de réservation	13
	7.2.4 Autres comportements.....	20
8	Description du service.....	20
8.1	Résumé du service	20
8.2	Description du service.....	22
	8.2.1 Règles applicables	22
	8.2.2 Primitive <i>RS-Check-Reservation-Systems</i>	22
	8.2.3 Primitive <i>RS-Connect</i>	23
	8.2.4 Primitive <i>RS-Disconnect</i>	25
	8.2.5 Primitive <i>RS-Conference-Reserve</i>	25
	8.2.6 Primitive <i>RS-Conference-Modify</i>	36
	8.2.7 Primitive <i>RS-Conference-Cancel</i>	39
	8.2.8 Primitive <i>RS-Conference-List-Inquire</i>	40
	8.2.9 Primitive <i>RS-Conference-Detail-Inquire</i>	42
	8.2.10 Primitive <i>RS-Conference-Check-Availability</i>	43
	8.2.11 Primitive <i>RS-Site-Record</i>	44
	8.2.12 Primitive <i>RS-Site-Modify</i>	49
	8.2.13 Primitive <i>RS-Site-Delete</i>	50
	8.2.14 Primitive <i>RS-Site-Directory-Inquire</i>	51
	8.2.15 Primitive <i>RS-Non-Standard-Request</i>	53
	8.2.16 Primitive <i>RS-Non-Standard-Data</i>	54
	8.2.17 Primitive <i>RS-Transaction-Allocation</i>	54
	8.2.18 Primitive <i>RS-Transaction-Acknowledge</i>	55
	8.2.19 Primitive <i>RS-Transaction-Cancel</i>	55
	8.2.20 Primitive <i>RS-Transaction-Error</i>	56

	Page
9	Spécification du protocole 56
9.1	Codage des unités PDU des transactions URST 56
9.2	Module ASN.1 des transactions URST 56

Recommandation T.135

TRANSACTIONS ENTRE SYSTEME DE RESERVATION ET UTILISATEURS DANS LES CONFERENCES T.120

(Genève, 1998)

1 Domaine d'application

La présente Recommandation décrit un protocole normalisé pour les transactions de réservation de conférence multimédia entre utilisateurs et fournisseurs de services, lorsque ces utilisateurs font appel à un terminal de type T.120.

L'objectif principal est de permettre aux participants à la conférence d'effectuer des opérations telles que la réservation de ressources pour leur prochaine conférence ou d'interagir directement à partir de la conférence à laquelle ils participent afin de modifier l'utilisation des ressources de la conférence.

Le modèle architectural et le protocole spécifiés dans la présente Recommandation font appel à un ensemble de paramètres qui permettent aux participants à la conférence de communiquer simultanément avec plusieurs fournisseurs de service.

2 Références normatives

Les Recommandations UIT-T et autres références suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation. Au moment de la publication, les éditions indiquées étaient en vigueur. Toute Recommandation ou autre référence est sujette à révision; tous les utilisateurs de la présente Recommandation sont donc invités à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et autres références indiquées ci-après. Une liste des Recommandations UIT-T en vigueur est publiée régulièrement.

- Recommandation UIT-T H.231 (1997), *Unités de commande multipoint pour les systèmes audiovisuels utilisant des canaux numériques fonctionnant à des débits inférieurs ou égaux à 1920 kbit/s.*
- Recommandation UIT-T H.242 (1997), *Procédures pour l'établissement de communications entre terminaux audiovisuels sur des canaux numériques d'un débit allant jusqu'à 2 Mbit/s.*
- Recommandation UIT-T H.243 (1997), *Procédures pour l'établissement de communications entre trois terminaux audiovisuels ou plus sur des canaux numériques d'un débit allant jusqu'à 1920 kbit/s.*
- Recommandation UIT-T H.261 (1993), *Codec vidéo pour services audiovisuels à $p \times 64$ kbit/s.*
- Recommandation UIT-T H.262 (1995), *Technologies de l'information – Codage générique des images animées et du son associé: données vidéo.*
- Recommandation UIT-T H.320 (1997), *Systèmes et équipements terminaux visiophoniques à bande étroite.*
- Recommandation UIT-T Q.931 (1993), *Spécification de la couche 3 de l'interface usager-réseau RNIS pour la commande de l'appel de base.*

- Recommandation T.50 du CCITT (1992), *Alphabet international de référence (ancien alphabet international n° 5 ou AI5) – Technologies de l'information – Jeux de caractères codés à 7 bits pour l'échange d'informations.*
- Recommandation UIT-T T.120 (1996), *Protocoles de données pour conférence multimédia.*
- Recommandation UIT-T T.122 (1993), *Service de communication multipoint pour la définition des services de conférence audiographique et conférence audiovisuelle.*
- Recommandation UIT-T.123 (1996), *Piles protocolaires de données propres au réseau pour conférences multimédias.*
- Recommandation UIT-T T.124 (1995), *Commande de conférence générique.*
- Recommandation UIT-T T.125 (1994), *Spécification de protocole du service de communication multipoint.*

3 Définitions

La présente Recommandation définit les termes suivants:

- 3.1 protocole d'application:** protocole normalisé ou non normalisé qui est utilisé par une application.
- 3.2 entité de protocole d'application:** nom donné à une instance de protocole d'application dans le cadre de l'infrastructure de communication T.120.
- 3.3 ressource de conférence:** partie d'un dispositif de conférence multimédia qui peut être utilisée au cours d'une conférence.
- 3.4 conférence multimédia:** association d'un certain nombre de dispositifs de conférence multimédia qui sont capables d'échanger des informations audiographiques et audiovisuelles au moyen de divers réseaux de communication. Dans la présente Recommandation, une conférence multimédia est censée être à base T.120 pour une partie ou pour la totalité de son système de transmission de données.
- 3.5 dispositif de conférence multimédia:** dispositif utilisé dans une conférence multimédia, qui peut se composer de terminaux, de multiaccès, de ponts MCU et d'autres machines.
- 3.6 multipoint:** capacité d'échanger une partie ou la totalité des signaux audio, vidéo ou de données simultanément entre de multiples dispositifs de conférence multimédia, par opposition à la capacité point à point dans laquelle les signaux audio, vidéo et/ou de données sont échangés entre deux nœuds directement connectés.
- 3.7 pont de conférence; unité de commande multipoint:** dispositif multiaccès couramment appelé *unité MCU* ou *pont MCU*, qui sert à connecter des terminaux et d'autres ponts MCU en mode multipoint. Un pont MCU n'est pas principalement destiné à servir d'extrémité pour une communication d'utilisateurs.
- 3.8 terminal multiaccès:** terminal d'utilisateur qui possède également la capacité de relayer des informations T.120. On trouvera au paragraphe 3/T.124 la définition complète d'un terminal multiaccès.
- 3.9 site enregistré:** site qui, compte tenu d'un système de réservation donné, est connu par ce système (et dont la description est mémorisée dans (une des) la base(s) de données du système de réservation) et qui peut être identifié sans ambiguïté par ce système au moyen de son *identificateur de site*.

- 3.10 application de réservation:** application utilisant une entité conforme à la Recommandation T.135 révisée.
- 3.11 connexion de réservation:** connexion de données établie point à point entre une application de réservation cliente et une application de réservation serveur, au moyen de laquelle un utilisateur s'inscrit dans un système de réservation.
- 3.12 dispositif de réservation:** nom générique donné aux éléments constituant d'un système de réservation conçu en architecture répartie.
- 3.13 domaine de réservation:** ensemble de dispositifs de conférence multimédia gérés par un système de réservation. Le terme *gestion* est utilisé dans le sens d'un contrôle direct des ressources de programmation dans le temps et, dans certains cas, dans le sens d'une commande en temps réel des dispositifs de conférence.
- 3.14 nœud serveur de réservation:** dispositif de réservation particulier qui possède la capacité de représenter le fournisseur de service et qui est un élément constituant d'une conférence multimédia.
- 3.15 système de réservation:** système capable de gérer l'ensemble du domaine de réservation associé.
- 3.16 transaction de réservation:** interaction entre deux applications de réservation qui est globalement fondée sur un modèle d'interrogation/réponse.
- 3.17 fournisseur de service:** organisation qui possède ou contrôle un ou plusieurs système(s) de réservation et qui les utilise pour fournir des services de dérivation pour une conférence multimédia.
- 3.18 abonné au service:** personne, organisation ou entité intermédiaire quelconque, connue (c'est-à-dire enregistrée dans (une des) la base(s) de données) par un système de réservation et reconnue par lui comme étant un utilisateur *enregistré* du fournisseur de service. L'abonné est identifié sans ambiguïté à l'intérieur du système de réservation du fournisseur de service, au moyen de son *identificateur d'abonné*.
- 3.19 utilisateur du service:** personne, organisation ou entité intermédiaire quelconque, qui utilise les services offerts par un fournisseur de services.
- 3.20 site:** élément informationnel global qui décrit un membre d'une conférence multimédia. L'information de site comprend les renseignements relatifs au dispositif de conférence multimédia utilisé, au réseau auquel le dispositif est directement connecté (adresse(s) réseau du dispositif comprise(s)), à l'emplacement physique du dispositif et aux participants humains utilisant le dispositif.
- 3.21 identificateur du site:** identification univoque permettant de reconnaître un site comme étant enregistré par un système de réservation.
- 3.22 identificateur d'abonné:** identification univoque permettant à un système de réservation de reconnaître un abonné au service.
- 3.23 terminal d'utilisateur:** terminal qu'un utilisateur utilise pratiquement pour participer à des conférences multimédias.

4 Abréviations

La présente Recommandation utilise les abréviations suivantes:

APE	entité du protocole d'application (<i>application protocol entity</i>)
API	interface de programmation d'application (<i>application programming interface</i>)
ATM	mode de transfert asynchrone (<i>asynchronous transfer mode</i>)
GCC	commande de conférence générique (<i>general conference control</i>)
GUI	interface graphique d'utilisateur (<i>graphical user interface</i>)
LAN	réseau local (<i>local area network</i>)
MCE	équipement de conférence multimédia (<i>multimedia conferencing equipment</i>)
MCS	service de communication multipoint (<i>multipoint communication service</i>)
MCU	pont de conférence; unité de commande multipoint (<i>multipoint control unit</i>)
PDU	unité de données de protocole (<i>protocol data unit</i>)
RD	dispositif de réservation (<i>reservation device</i>)
RDCC	réseau pour données à commutation de circuits
RNIS	réseau numérique à intégration de services
RSN	nœud serveur de réservations (<i>reservation server node</i>)
RSN-RA	application de réservation par nœud serveur de réservations (<i>reservation server node reservation application</i>)
RTC	réseau téléphonique commuté
RTGC	réseau téléphonique général commuté
URST	transactions entre système de réservation et utilisateurs (<i>user-to-reservation system transactions</i>)
UT	terminal utilisateur (<i>user terminal</i>)
UT-RA	application de réservation par terminal utilisateur (<i>user terminal reservation application</i>)

5 Conventions

Les codes suivants sont utilisés pour les paramètres contenus dans les primitives des services abstraits définis dans la présente Recommandation:

M	le paramètre est obligatoire
C	le paramètre est conditionnel
O	le paramètre est facultatif (option)
Blanc	le paramètre est absent
(=)	la valeur du paramètre est identique à celle du paramètre correspondant de la primitive précédente, l'ordre de préséance étant: demande, indication, réponse et confirmation.
(=RQ)	la valeur du paramètre est identique à celle du paramètre correspondant dans une primitive précédente, avec RQ = demande, IN = indication, RS = réponse et CF = confirmation.

Les primitives des services ne peuvent être que de l'un des quatre types suivants: demande, indication, réponse et confirmation. Certaines primitives peuvent prendre en charge tous ces types, alors que d'autres ne le peuvent pas. Les quatre types sont définis comme suit:

- primitives de demande:** primitives envoyées par une application de réservation à une entité APE effectuant des transactions URST (URST APE) afin de lancer une transaction URST ou de commander un service spécifique.
- primitives d'indication:** primitives envoyées par une entité URST APE à une application de réservation pour faire suite soit à la réception d'une unité PDU de demande ou d'indication soit au lancement d'une action par l'entité URST APE.
- primitives de réponse:** primitives envoyées par une application de réservation à une entité URST APE en réponse à une primitive d'indication nécessitant une réponse.
- primitives de confirmation:** primitives envoyées par une entité URST APE à une application de réservation pour faire suite à la réception d'une unité PDU de réponse ou directement en réponse à une primitive de demande.

Les unités PDU se divisent en trois catégories. Les noms de toutes les unités PDU contiennent les mots *demande*, *indication* ou *réponse*, qui indiquent leur utilisation prévue. Les unités PDU sont définies comme suit:

- unités PDU de demande** unités qui nécessitent une unité PDU de réponse en retour. Une entité URST APE qui reçoit de son homologue une unité PDU d'indication envoie à l'application de réservation concernée une primitive de service de type indication.
- unités PDU d'indication** unités qui ne nécessitent pas de réponse (par exemple celles qui ne contiennent que des informations). Une entité URST APE qui reçoit de son homologue une unité PDU de demande envoie à l'application de réservation concernée une primitive de service de type indication.
- unités PDU de réponse:** unités qui apportent une réponse à une unité PDU de demande particulière. Une unité PDU de réponse est envoyée par une entité URST APE à son homologue, soit automatiquement soit en réponse à une primitive de service de type réponse correspondante, envoyée par une application de réservation. Une entité URST APE qui reçoit de son homologue une unité PDU de réponse envoie à l'application de réservation concernée une primitive de service de type confirmation.

6 Introduction

Pour réserver une conférence multimédia, un utilisateur peut faire appel à un terminal T.120 (abrégé dans la présente Recommandation en *UT* ou en *terminal utilisateur*) pour réserver les ressources nécessaires et indiquer les équipements requis lors du déroulement d'une conférence. La présente Recommandation contient une spécification du protocole d'application ainsi qu'une description des services permettant à une application de réservation, située dans un terminal utilisateur, d'entrer en interaction avec une autre application de réservation située chez une entité de fournisseur de services et appelée ci-après *système de réservation*. Bien que le protocole décrit dans la présente Recommandation soit traité par une *entité du protocole d'application (APE, entité du protocole*

d'application) conforme à la Recommandation T.120, qui fait appel aux services MCS et à la commande GCC, ce protocole peut être utilisé dans d'autres infrastructures de communication. Cet aspect est cependant hors du domaine d'application de la présente Recommandation.

Le service de *transactions entre système de réservation et utilisateurs* (URST, *user-to-reservation system transaction*), fourni par une entité URST APE, est en mode connexion en ce sens qu'il permet à un utilisateur, défini par son identificateur d'abonné, de s'inscrire dans un système de réservation. Il permet à cet utilisateur de programmer dans le temps des conférences et de modifier ou d'annuler des réservations antérieures. Les conférences peuvent être programmées sur un ou plusieurs ponts MCU régis par un même système de réservation; mais l'utilisateur a la possibilité d'insérer des ponts MCU "externes" dans la liste des sites définissant la conférence. Le service de transactions URST fait également usage d'équipements d'interrogation tels que l'énumération (listage) et la lecture des conférences programmées ou engagées, la recherche des ressources disponibles et l'accès au service d'annuaire des sites.

En plus de l'ensemble de fonctions défini par la présente Recommandation, le protocole du service de transactions URST offre la possibilité de dialogues non normalisés entre un utilisateur et un fournisseur de services.

La présente Recommandation ne spécifie aucune interface graphique d'utilisateur (GUI, *interface graphique d'utilisateur*) car les applications de réservation mises au point pour les terminaux d'utilisateur sont censées offrir cette caractéristique. En d'autres termes, la description du service URST se rapporte aux services offerts par le protocole URST, la façon dont ces derniers sont utilisés par une application pour présenter un assortiment de services à un utilisateur étant hors du domaine d'application de la présente Recommandation.

7 Modèle T.135

Ce modèle part du principe que le terminal utilisé pour accéder au système de réservation est conforme à la Recommandation T.120 et participe à une conférence T.120 en cours. Bien qu'une telle conférence soit considérée comme tenue par des utilisateurs dans le sens courant du terme, le modèle n'exclut pas la possibilité que des fournisseurs de services organisent des conférences T.120 particulières, servant à fournir un accès à des services de réservation pour utilisateurs. Cet aspect est considéré comme étant une option offerte au fournisseur de services et est hors du domaine d'application de la présente Recommandation.

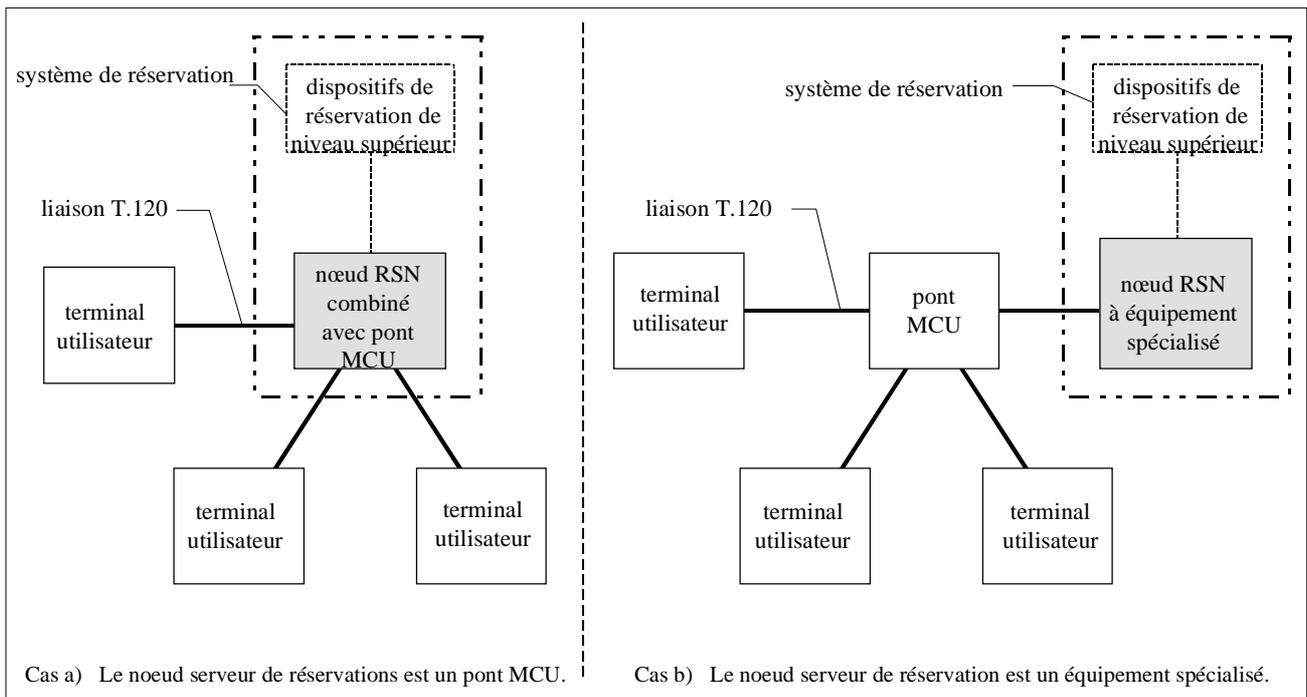
7.1 Modèle du système

Le système de réservation est décrit sous la forme d'une entité fonctionnelle qui donne suite et répond à des demandes de réservation pour des conférences entre utilisateurs, c'est-à-dire qui recherche, affecte et commande le pont de conférence (MCU, *multipoint control unit*) nécessaire et d'autres ressources réseau. Le système de réservation est censé comporter des extrémités électroniques conformes à la Recommandation T.120, appelées *nœuds serveurs de réservation* (RSN, *reservation server node*). Ces nœuds participent aux conférences entre utilisateurs et donnent ainsi accès à des services de réservation. En dehors des nœuds RSN, l'architecture exacte du système de réservation n'est pas spécifiée par la présente Recommandation. Elle peut être plus ou moins complexe et comporte un ou plusieurs dispositifs électroniques ainsi que des opérateurs humains. Il convient cependant de noter que le service-protocole URST permet aux utilisateurs d'interagir en cours de conférence, ce qui fait penser, intuitivement, à un processus automatisé.

Du point de vue architectural, le nœud RSN peut être une partie interne (ou un agent intermédiaire) d'un pont MCU ou d'un terminal multiaccès exploité par le fournisseur de services, ou être un dispositif de gestion quelconque, intégralement ou partiellement consacré aux fins de réservation.

Ces possibilités relèvent d'un choix du fournisseur de services et sont laissées hors du domaine d'application de la présente Recommandation. De toute façon, du point de vue du terminal de réservation, le nœud RSN est considéré comme étant le système de réservation. Dans la présente Recommandation, les composants non RSN d'un système de réservation sont appelés *dispositifs de réservation (RD, reservation device)*.

La Figure 7-1 montre des exemples de configuration. Les nœuds RSN sont indiqués par des cadres grisés. Les traits gras représentent des liaisons T.120 tandis que les traits interrompus correspondent aux liaisons de type non spécifié. Le cas a) montre le cas d'un nœud RSN implanté dans un pont MCU. Le cas b) montre le cas d'un nœud RSN implanté dans un équipement spécialisé.



T1602080-97

Figure 7-1/T.135 – Le nœud serveur de réservations dans le modèle du système

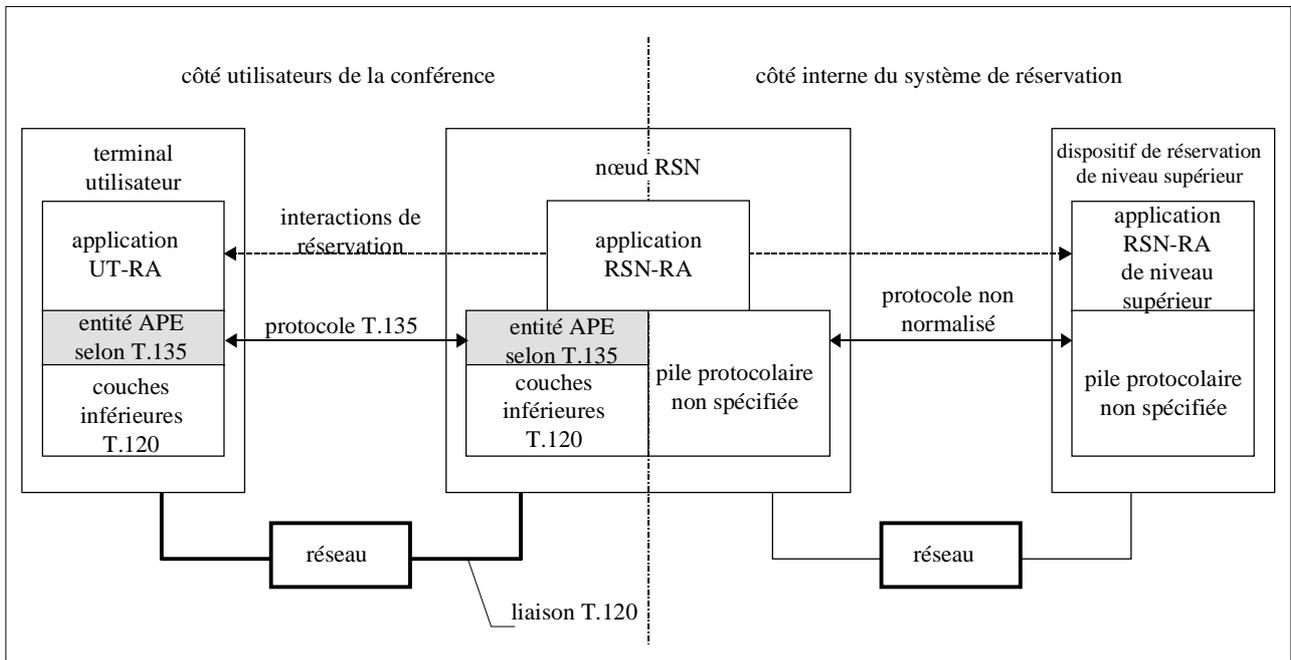
Bien que, par souci de simplicité, la Figure 7-1 ne représente qu'un seul nœud RSN participant à la conférence d'utilisateurs, il convient de noter que le modèle autorise plusieurs nœuds RSN par conférence. D'autre part, différents nœuds RSN représentant différents systèmes de réservation (et donc différents fournisseurs de service) peuvent être raccordés à une conférence d'utilisateurs.

Le nœud RSN est un dispositif T.120 qui comporte une application de réservation par nœud serveur de réservations (RSN-RA, *reservation server node reservation application*) et qui utilise le protocole exploité par une entité URST APE. Au moyen du protocole de transactions URST, l'application RSN-RA dialogue avec une application de terminal utilisateur (UT-RA, *user terminal reservation application*) située dans le terminal utilisateur qui dépend également d'une entité URST APE. Selon le choix de la réalisation, l'entité URST APE peut être une entité indépendante qui offre ses services à l'application (aux applications) de réservation ou qui y est intégrée. L'acheminement, par une entité URST APE, des informations contenues dans les unités de données du protocole (PDU, *protocol data unit*) de transactions URST jusqu'à une application de réservation est modélisé par les primitives de service URST qui sont spécifiées dans la présente Recommandation. Le modèle part du

principe que l'entité URST APE est chargée de communiquer avec les entités MCS et GCC sous-jacentes.

Le rôle joué par l'application RSN-RA dépend de l'architecture du système de réservation. Sauf s'il est totalement autonome, le nœud RSN dérivera le domaine de la conférence d'utilisateurs vers un dispositif de réservation de niveau supérieur, contenant une entité modélisée sous la forme d'une application RSN-RA capable de traiter des transactions de réservation. La pile de protocoles de réservation et de transport utilisée entre le nœud RSN et le dispositif de réservation de niveau supérieur est hors du domaine d'application de la présente Recommandation. Quel que soit le rôle joué par l'application RSN-RA en combinaison avec des applications RSN-RA de niveau supérieur contenues dans des dispositifs de réservation de niveau supérieur, le nœud RSN doit apparaître à la conférence d'utilisateurs comme un tout conforme à l'ensemble des fonctions définies par le protocole URST.

La Figure 7-2 montre le modèle T.135 d'infrastructure de communication. Les cadres grisés représentent les parties faisant l'objet des prescriptions de la présente Recommandation, tandis que les cadres non grisés représentent des parties hors du domaine d'application de la présente Recommandation. La ligne tiretée verticale représente la frontière virtuelle entre le domaine de conférence de l'utilisateur et l'arrière du nœud RSN. Par souci de simplicité, la Figure 7-2 représente le côté interne du système de réservation avec un seul dispositif de réservation de niveau supérieur. La ligne pointillée horizontale représente le dialogue virtuel de réservation qui a lieu entre une application RSN-RA de niveau supérieur contenue dans le dispositif de réservation et l'application UT-RA qui est contenue dans le terminal de l'utilisateur. Dans une configuration réelle, le côté interne du système de réservation peut comporter une chaîne de plusieurs dispositifs de réservation.



T1602090-97

Figure 7-2/T.135 – Modèle du système et portée de la présente Recommandation

7.2 Mode de fonctionnement

Les interactions entre une application UT-RA et une application RSN-RA sont de type point à point et sont modélisées par des opérations de type question/réponse. Elles sont appelées *transactions de*

réserve dans la présente Recommandation. Les transactions de réservation ont lieu au moyen d'un objet appelé *connexion de réservation*. L'application UT-RA est toujours à l'origine des connexions de réservation, ainsi que le côté interrogation de toutes les transactions de réservation définies. Il est cependant prévu de permettre, dans de futures versions, d'autres configurations.

Une connexion de réservation implique une tentative réussie d'inscription de l'application UT-RA dans le système de réservation, au moyen d'un *identificateur d'abonné*. Les transactions de réservation peuvent être effectuées une fois que les actions suivantes ont été réalisées en séquence:

- l'application UT-RA contenue dans le terminal utilisateur a reconnu l'application RSN-RA;
- l'application UT-RA a réussi à ouvrir une connexion de réservation auprès de l'application RSN-RA.

Une entité URST APE utilise exclusivement le service *MCS-Send-Data* pour envoyer et recevoir des unités PDU de transactions URST. Chaque côté envoie des données sur le canal indiqué par l'identificateur d'utilisateur de service MCS homologue.

7.2.1 Identification des applications de réservation

Une fois raccordée à la conférence entre utilisateurs, une application RSN-RA qui souhaite se mettre à la disposition d'autres nœuds doit d'abord obtenir un identificateur d'utilisateur de service MCS au moyen de la commande *MCS-ATTACH-USER* du service MCS puis s'inscrire à la *session d'enregistrement* URST (voir la Recommandation T.121) par la commande *GCC-Application-Enroll* du service GCC. Conformément aux directives de la Recommandation T.121, l'inscription à la session d'enregistrement doit être effectuée avec le fanion indicateur *Inscription/Désinscription* à la valeur *Inscription* et avec le fanion indicateur *Activité/Inactivité* à la valeur *Inactivité*. Cependant, l'application RSN-RA **DOIT** toujours transmettre son identificateur d'utilisateur du service MCS au moyen du paramètre *Identificateur d'utilisateur d'application* inséré dans la primitive *GCC-Application-Enroll-request*.

Pour ouvrir une connexion de réservation, une application UT-RA doit suivre le même processus. Mais comme la présente version de la Recommandation T.135 définit les connexions de réservation comme étant toujours ouvertes par des applications UT-RA, celles-ci peuvent attendre d'avoir besoin d'ouvrir une connexion de réservation pour demander un identificateur d'utilisateur du service MCS et pour s'inscrire dans la session d'enregistrement.

Une application de réservation doit utiliser ses *Capacités d'application sans condensation* (voir la Recommandation T.124) associées pour signaler son type à ses homologues, c'est-à-dire pour indiquer si elle est implantée dans un nœud RSN représentant un système de réservation (c'est-à-dire un fournisseur de services) ou dans un terminal utilisateur. Les capacités normalisées sans condensation qui sont présentées dans le Tableau 7-1 sont définies à cette fin.

Tableau 7-1/T.135 – Capacités normalisées sans condensation

Nom	ID	Données d'application associées	Dépendance
type d'application de réservation	0	longueur d'un octet, interprété comme une valeur d'entier non signée qui doit être mise à : <ul style="list-style-type: none"> • 0 pour les applications RSN-RA • 1 pour les applications UT-RA d'utilisateur 	obligatoire
nom du système de réservation	1	longueur non définie, interprétée comme une chaîne de caractères codés conformément aux règles de la Recommandation T.50.	obligatoire pour les RSN-RA, sinon non applicable.
identificateur du système de réservation	2	chaîne d'octets dont la longueur est comprise dans l'étendue [4...255] et qui est codée en valeur ASN.1 du type <i>H221NonStandardIdentifier</i> tel que spécifié dans la Rec. T.124.	facultatif pour les RSN-RA, sinon non applicable.

Toutes ces capacités et leurs données d'application assignées sont des paramètres d'initialisation pour l'application de réservation. La liste des applications RSN-RA inscrites dans la session d'enregistrement URST et de leurs capacités normalisées associées peut être retrouvée par l'application UT-RA au moyen du service *RS-Check-Reservation-Systems*.

Les données d'application en chaînes de caractères, assignées à la capacité *système de réservation*, sont destinées à être présentées à l'utilisateur par l'application UT-RA afin de lui permettre d'identifier le système de réservation (c'est-à-dire le fournisseur de services). Son contenu doit donc être interprétable par une personne.

Les données d'application en chaînes d'octets, assignées à la capacité *Identificateur du système de réservation*, sont destinées à être interprétées par les machines. Leur objet est de permettre l'automatisation de l'identification des systèmes de réservation. Sur accord préalable, un fournisseur de services peut indiquer à un utilisateur un motif alphanumérique identifiant son système de réservation. L'utilisateur peut ensuite configurer son application UT-RA de façon à reconnaître automatiquement ce motif et à le distinguer des autres. La chaîne d'octets doit contenir un paramètre ASN.1 du type *H221NonStandardIdentifier* défini dans la Recommandation T.124, les quatre premiers octets de son champ de valeur contenant l'indicatif du pays et du fournisseur de services, conformément à la spécification figurant dans l'Annexe A/H.221 pour les séquences d'échappement *NS-cap* et *NS-comm*.

Lorsqu'une application RSN-RA est inscrite dans la session d'enregistrement, des applications UT-RA homologues peuvent tenter d'ouvrir une connexion de réservation avec elle. Une application RSN-RA se rendra indisponible en se désinscrivant de la session d'enregistrement.

7.2.2 Connexions de réservation

7.2.2.1 Pointeurs et identificateurs de connexion

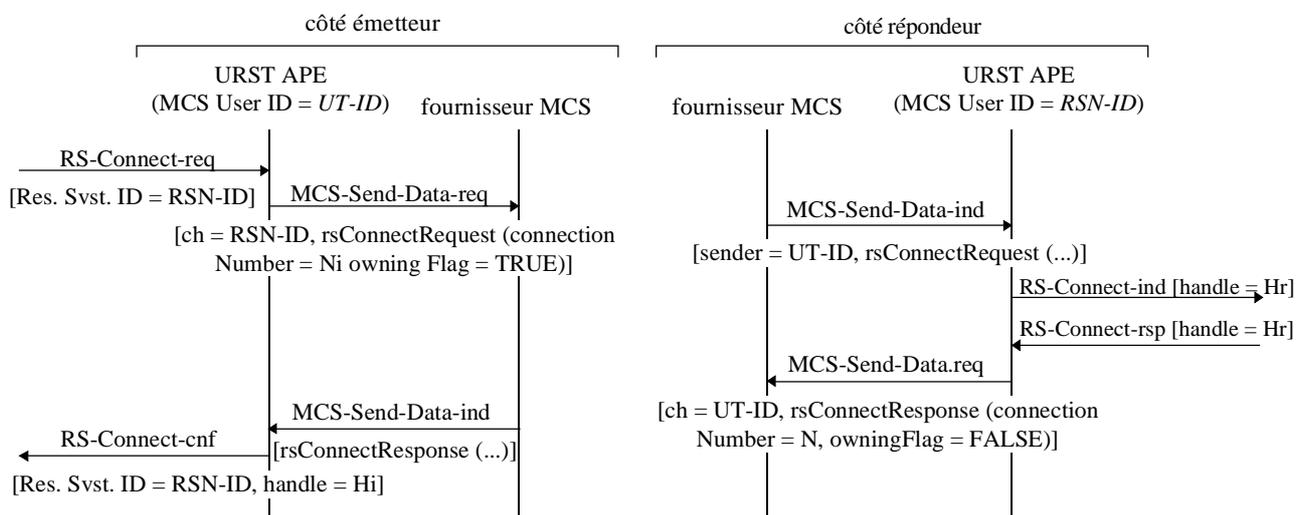
Le service de transactions URST permet à une application de réservation de traiter simultanément plusieurs connexions de réservation. Entre une application de réservation et l'entité URST APE sous-jacente, chaque connexion de réservation ouverte est identifiée localement par le paramètre *Pointeur de connexion* contenu dans les primitives du service URST. Le format et le mécanisme de construction des pointeurs de connexion sont considérés par la présente Recommandation comme relevant d'une décision locale.

Au niveau du protocole (c'est-à-dire entre deux entités URST APE distantes), l'identification d'une connexion de réservation est effectuée au moyen du paramètre *ConnectionID* contenu dans les unités PDU URST. Ce paramètre est du type ASN.1 *ConnectionID* qui consiste en un nombre entier compris dans l'étendue [0...65535], appelé *numéro de connexion*, et en un fanion booléen, appelé *fanion de possession*, qui indique au récepteur d'une unité PDU URST si le numéro de la connexion a été ou non attribué par l'émetteur de l'unité PDU (c'est-à-dire si la connexion de réservation en cause a été ouverte par l'émetteur de l'unité PDU)¹.

L'entité URST APE située du côté ouverture d'une connexion de réservation est chargée d'attribuer le numéro de connexion. Aux deux extrémités de la connexion, les entités URST APE sont chargées d'attribuer le pointeur de connexion et de tenir à jour le mappage avec son numéro de connexion correspondant.

7.2.2.2 Ouverture d'une connexion de réservation

Dès réception de la primitive du service *RS-Connect-request* (demande de connexion de réservation) issue de l'application de réservation, l'entité URST APE située du côté émetteur doit d'abord attribuer un numéro de connexion avec son pointeur de connexion associé. Elle doit ensuite envoyer une unité PDU *rsConnectRequest* à l'extrémité homologue en la faisant passer par le canal indiqué par l'identificateur de l'utilisateur du service MCS de l'entité URST APE homologue. Elle doit ensuite construire le paramètre *ConnectionID* contenant le numéro de connexion attribué et le fanion de possession activé (c'est-à-dire ayant la valeur TRUE). L'entité URST APE située du côté homologue doit répondre à cette unité en envoyant une unité PDU de type *rsConnectResponse* vers l'extrémité émettrice, en la faisant passer par son propre canal d'identificateur d'utilisateur du service MCS et en réutilisant le même numéro de référence pour construire la valeur du paramètre *ConnectionID* mais sans activer le fanion de possession (c'est-à-dire en lui donnant la valeur FALSE). Le pointeur de connexion attribué à l'extrémité d'ouverture est indiqué par le paramètre *Pointeur de connexion* contenu dans la primitive *RS-Connect-Response*, dès réception de l'unité PDU *rsConnectResponse*.



T1602100-97

Figure 7-3/T.135 – Ouverture d'une connexion de réservation

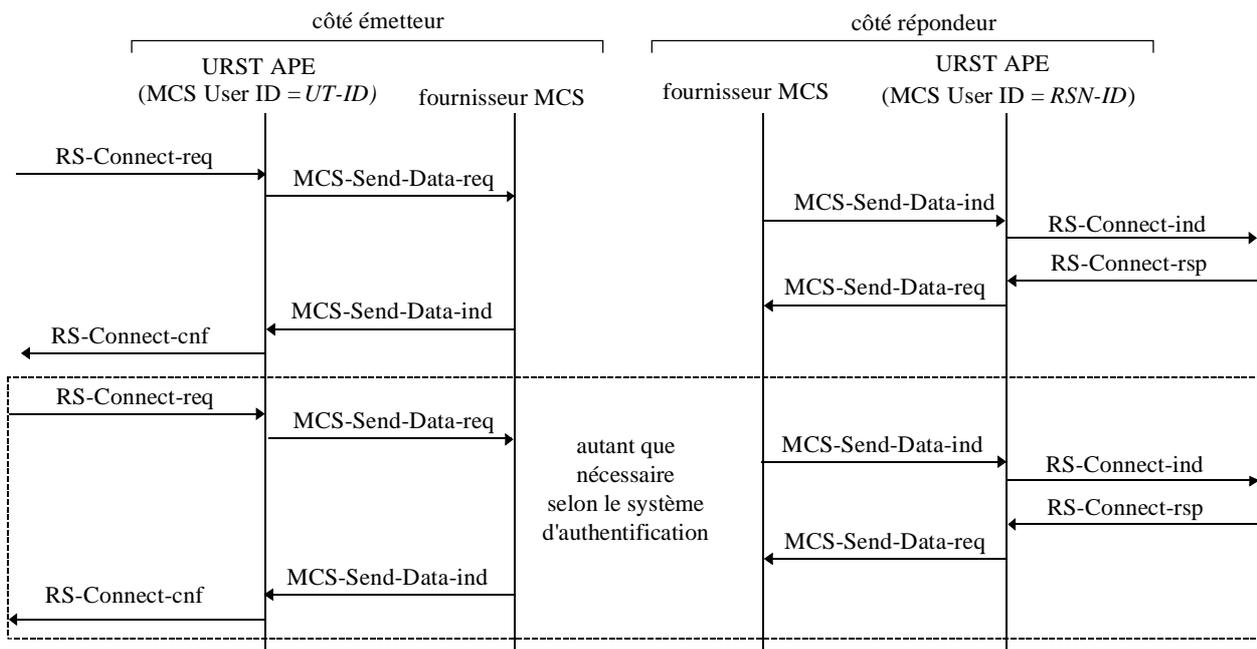
¹ Cette règle a pour objet de permettre la définition d'une connexion lancée par le système de réservation dans de futures révisions de la présente Recommandation.

7.2.2.3 Scénarios d'authentification

Le service et le protocole de transactions URST permettent de mettre en œuvre des systèmes d'authentification faibles (fondés sur de simples mots de passe) ou des systèmes d'authentification forts (fondés sur des algorithmes unidirectionnels ou à réponse sur épreuve rédhibitoire).

Le système d'authentification peut se composer de multiples passes, consistant chacune d'une unité PDU *rsConnectRequest* envoyée par l'extrémité émettrice à l'extrémité réceptrice, ainsi que d'une unité PDU *rsConnectResponse* envoyée par l'extrémité réceptrice à l'extrémité émettrice. Lorsque de multiples passes sont effectuées, seul le paramètre *resultCode* contenu dans la dernière unité PDU *rsConnectResponse* doit toujours être mis à la valeur "*authenticationInProgress*".

La Figure 7-4 montre la séquence d'initialisation d'une connexion de réservation.



T1602110-97

Figure 7-4/T.135 – Séquence d'initialisation d'une connexion de réservation

7.2.2.4 Fermeture d'une connexion de réservation

Une connexion de réservation est fermée lorsqu'une quelconque des applications UT-RA ou RSN-RA envoie une primitive de service *RS-Disconnect-request* (demande de déconnexion) à son entité URST APE locale, lorsqu'un quelconque des terminaux utilisateurs ou nœuds RSN actifs se déconnecte de la conférence, ou lorsqu'il est mis fin à la conférence.

Dans le premier cas, l'entité URST APE située du côté fermeture doit envoyer une unité PDU *rsDisconnectIndication* à son homologue puis abandonner le pointeur de connexion et le numéro de référence protocolaire correspondants. Dès réception de l'unité PDU *rsDisconnectIndication*, l'entité URST APE située du côté homologue envoie une primitive *RS-Disconnect-indication* à l'application de réservation supérieure et abandonne le pointeur de connexion correspondant.

Dans les autres cas, chaque entité URST APE propage une primitive *RS-Disconnect-indication* vers l'application de réservation locale puis abandonne le pointeur de connexion et le numéro de référence protocolaire correspondants.

7.2.3 Transactions de réservation

7.2.3.1 Identificateurs et pointeurs de transaction

Le service de transactions URST permet à une application de réservation de traiter simultanément plusieurs transactions de réservation. Entre une application de réservation et l'entité URST APE sous-jacente, chaque transaction de réservation ouverte est identifiée localement par le paramètre *Pointeur de transaction* contenu dans les primitives du service URST. Le format et le mécanisme de construction des pointeurs de transaction sont considérés par la présente Recommandation comme relevant d'une décision locale.

Au niveau du protocole (c'est-à-dire entre deux entités URST APE distantes), l'identification d'une transaction de réservation est effectuée au moyen du paramètre *transactionID* contenu dans les unités PDU URST. Ce paramètre est du type ASN.1 *TransactionID* qui consiste en un nombre entier compris dans l'étendue [0...65535], appelé *numéro de transaction*, et en un fanion booléen, appelé *fanion de possession*, qui indique au récepteur d'une unité PDU URST si le numéro de la transaction a été ou non attribué par l'émetteur de l'unité PDU (c'est-à-dire si la transaction de réservation en cause a été ouverte par l'émetteur de l'unité PDU).

L'entité URST APE située du côté ouverture d'une transaction de réservation est chargée d'attribuer le numéro de transaction. Aux deux extrémités de la transaction, les entités URST APE sont chargées d'attribuer le pointeur de transaction et de tenir à jour le mappage avec son numéro de transaction correspondant.

Les primitives de service énumérées ci-après lancent les transactions de réservation:

- *RS-Conference-Detail-Inquire-request* (demande de consultation détaillée sur une conférence RS);
- *RS-Conference-Cancel-request* (demande d'annulation de conférence RS);
- *RS-Conference-Check-Availability-request* (demande de contrôle de disponibilité de conférence RS);
- *RS-Conference-List-Inquire-request* (demande de consultation de liste de conférences RS);
- *RS-Conference-Modify-request* (demande de modification de conférence RS);
- *RS-Conference-Reserve-request* (demande de programmation de conférence);
- *RS-Site-Delete-request* (demande de suppression de site RS);
- *RS-Site-Directory-Inquire-request* (demande de consultation d'annuaire de sites RS);
- *RS-Site-Modify-request* (demande de modification de site RS);
- *RS-Site-Record-request* (demande d'enregistrement de site RS);
- *RS-Non-Standard-Transaction-request* (voir 7.2.3.4) (demande de transaction RS non normalisée).

Lorsque les nouveaux numéros et pointeurs de transactions sont attribués, l'entité URST APE répond immédiatement en envoyant à l'application de réservation une primitive *RS-Transaction-Allocation-indication* indiquant le nouvel identificateur au moyen du paramètre *Pointeur de transaction* de cette primitive, avant d'envoyer à l'entité URST APE homologue l'unité PDU *rsXXXRequest* appropriée.

7.2.3.2 Séquences et procédures transactionnelles

Toute transaction commence par l'envoi d'une unité PDU *rsXXXRequest* par l'entité URST APE située du côté origine, sur demande de l'application de réservation. L'entité URST APE homologue répond à l'unité PDU de demande en renvoyant l'unité PDU *rsXXXResponse* correspondante, qui contient un paramètre *acknowledgmentType* qui définit trois classes de transactions: *explicitement acquittées*, *acquittées* et *non acquittées*.

Une transaction non acquittée est fermée normalement lorsque l'entité URST APE du côté origine reçoit à son tour l'unité PDU de réponse correspondante, *rsXXXResponse*. A ce moment-là, l'entité URST APE envoie à l'application de réservation la primitive de service *RS-XXX-confirm* correspondante et abandonne l'identificateur de transaction. Du côté récepteur d'une transaction non acquittée, l'entité URST APE peut abandonner l'identificateur de transaction correspondant dès que l'unité PDU *rsXXXResponse* est envoyée à l'extrémité homologue.

Les transactions acquittées sont fermées normalement d'une manière analogue, mais avec la différence suivante. Lorsqu'elle reçoit l'unité PDU *rsXXXResponse* indiquant une transaction acquittée, l'entité URST APE située du côté origine doit envoyer à l'application de réservation la primitive de service *RS-XXX-confirm* correspondante et envoyer automatiquement à l'extrémité homologue une unité PDU *rsTransactionAcknowledgeIndication* avant d'abandonner l'identificateur de transaction. Du côté homologue, l'entité URST APE doit produire une primitive d'indication *RS-Transaction-Acknowledge-indication* dès qu'elle reçoit l'unité *rsTransactionAcknowledgeIndication* et doit abandonner à cet instant l'identificateur de transaction.

Les transactions explicitement acquittées sont fermées normalement, comme suit. Lorsqu'elle reçoit l'unité PDU *rsXXXResponse* indiquant une transaction explicitement acquittée, l'entité URST APE située du côté origine doit l'indiquer à l'application de réservation par le paramètre *Acquittement explicite requis* contenu dans la primitive de service *RS-XXX-confirm* correspondante. L'entité URST APE doit ensuite attendre de recevoir, en provenance de l'application de réservation, une primitive *RS-Transaction-Acknowledge-request*. Lorsque cela se produit, l'entité APE doit envoyer à l'extrémité homologue une unité PDU *rsTransactionAcknowledgeIndication* avant d'abandonner l'identificateur de transaction. Du côté homologue, l'entité URST APE doit produire une primitive d'indication *RS-Transaction-Acknowledge-indication* dès qu'elle reçoit l'unité *rsTransactionAcknowledgeIndication*, avant d'abandonner l'identificateur de transaction.

Les transactions normalisées ci-après sont toujours du type acquitté:

- *RS-Conference-Cancel* (annulation de conférence RS);
- *RS-Site-Delete* (suppression de site RS);
- *RS-Site-Modify* (modification de site RS);
- *RS-Site-Record* (enregistrement de site RS).

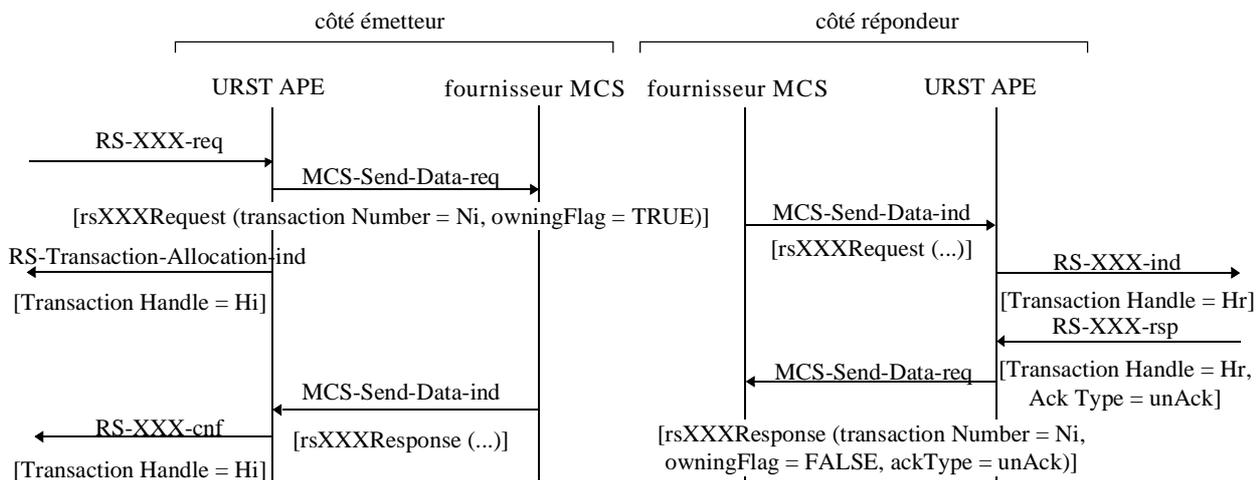
Les transactions normalisées ci-après sont soit du type acquitté soit du type explicitement acquitté:

- *RS-Conference-Modify* (modification de conférence RS);
- *RS-Conference-Reserve* (programmation de conférence).

D'autres transactions normalisées dans la présente Recommandation sont du type acquitté, tandis que les transactions non normalisées ou subtransactionnelles peuvent être d'un type quelconque (voir 7.2.3.4).

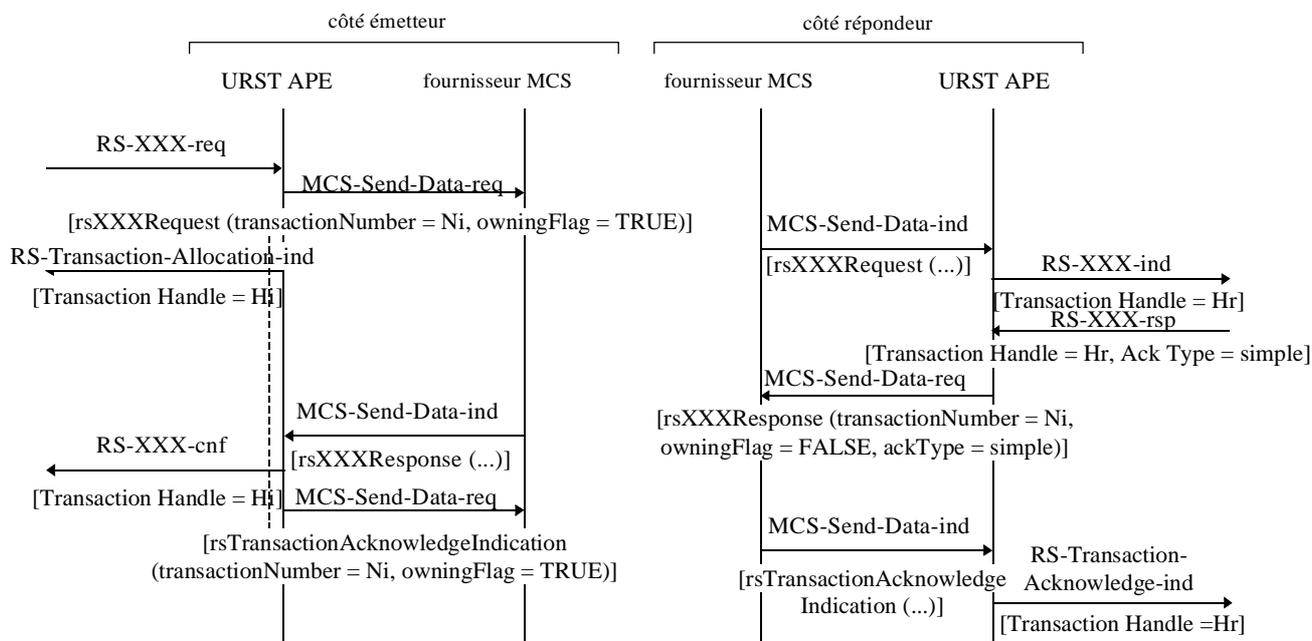
Une application de réservation du côté origine d'une transaction peut demander explicitement sa terminaison en émettant une primitive de service *RS-Transaction-Cancel-request*. Dans ce cas, l'entité URST APE enverra une unité PDU *rsTransactionCancelIndication* à l'extrémité homologue et abandonnera l'identificateur de transaction. Si l'entité URST APE reçoit l'unité PDU de réponse plus tard (c'est-à-dire si elle reçoit une unité PDU de réponse dont l'identificateur de transaction indique que c'est de son propre côté que se trouvait l'émetteur de la transaction et qui ne correspond à aucun identificateur attribué), cette entité doit ignorer cette unité PDU. Une entité URST APE qui reçoit une unité PDU *rsTransactionCancelIndication* doit envoyer à l'application de réservation une primitive *RS-Transaction-Cancel-indication*.

Les Figures 7-5 à 7-7 montrent les séquences normales pour chaque type de transaction normalisée.



T1602120-97

Figure 7-5/T.135 – Séquence pour transactions normalisées non acquittées



T1602130-97

Figure 7-6/T.135 – Séquence pour transactions normalisées acquittées

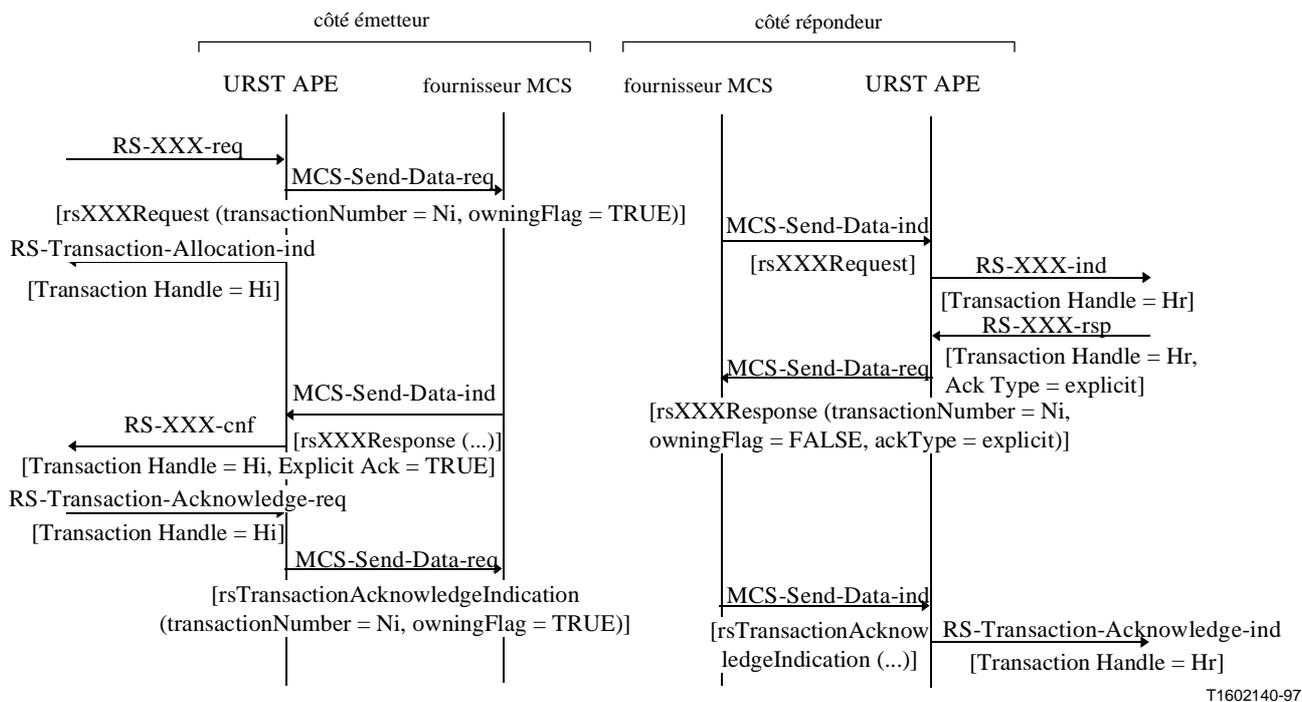


Figure 7-7/T.135 – Séquence pour transactions normalisées explicitement acquittées

7.2.3.3 Paramètres contrôlés du système de réservation

La plupart des paramètres des transactions normalisées acquittées, énumérées au 7.2.3.2, sont destinés à être configurés par l'utilisateur quelle que soit la présentation de ces paramètres par l'application UT-RA. La plupart de ces paramètres correspondent cependant plus à une préférence de l'utilisateur qu'à une stricte prescription relative à la transaction. Le paramètre *Méthode de commutation vidéo* de la transaction *RS-Conference-Reserve* en est un exemple. De façon à ne pas contraindre le système de réservation à refuser systématiquement une transaction parce qu'une telle préférence d'utilisateur ne correspond pas à son ensemble de capacités, le système de réservation est autorisé à accepter une transaction tout en modifiant la valeur du paramètre correspondant dans la primitive/l'unité PDU de réponse de la transaction. Ces paramètres sont donc appelés *Paramètres contrôlés du système de réservation* et sont désignés par l'abréviation "(RC)" placée à la suite de leur nom dans les tableaux du paragraphe 8.

Dans une primitive de service réponse, le système de réservation indiquera qu'il a "fait le tour" d'un groupe de tels paramètres en activant le paramètre *Acquittement explicite requis*. Dans ce cas, l'entité URST APE située dans le nœud RSN produira l'unité PDU de réponse avec le paramètre *acknowledgmentType* indiquant qu'un acquittement explicite est requis et sera effectué comme expliqué au 7.2.3.2.

7.2.3.4 Echanges d'informations non normalisées

Les services *RS-Non-standard-Transaction* et *RS-Non-Standard-Data* de la présente Recommandation permettent d'émettre et de recevoir trois types d'unité PDU: *rsNonStandardRequest*, *rsNonStandardResponse* et *rsNonStandardIndication*, appelées collectivement "unités PDU non normalisées" à cause de leur nom, bien qu'elles soient spécifiées dans la présente Recommandation.

Les unités PDU non normalisées contiennent le paramètre facultatif *protocolKey* qui peut être utilisé pour identifier un protocole ou format de données, normalisé ou non normalisé, utilisé dans le champ de données de ces unités. Le format de ce paramètre est analogue à celui qui est défini dans la

Recommandation T.124 (voir 3.3/T.124) pour une *Clé de protocole d'application*. Le paramètre *protocolKey* correspond au paramètre *Clé de protocole* des primitives de service *RS-Non-Standard-Transaction* et *RS-Non-Standard-Data*.

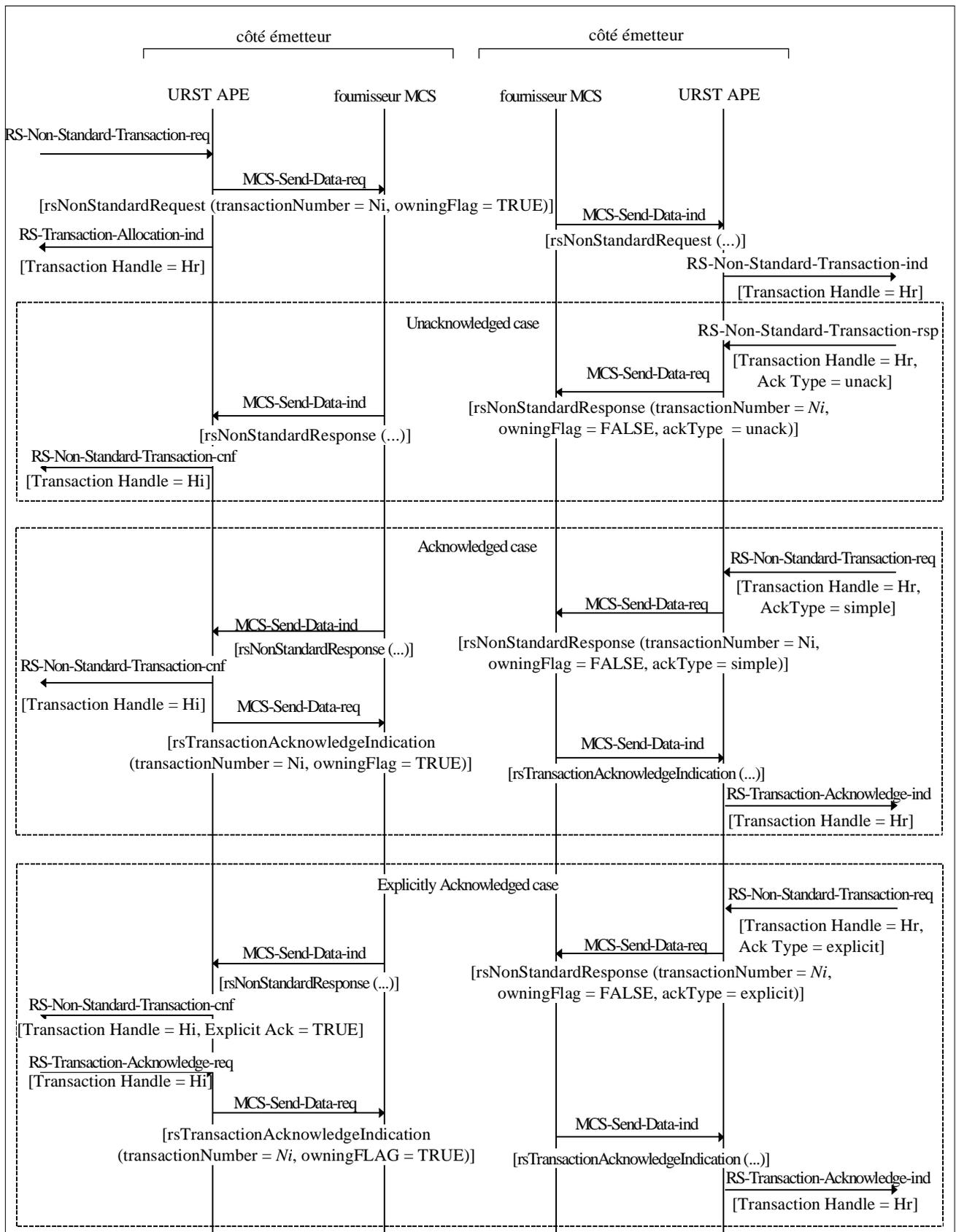
Comme décrit aux 7.2.3.4.1 à 7.2.3.4.4, le service *RS-Non-Standard-Transaction* est utilisé pour manipuler des transactions non normalisées ou des *sous-transactions*, tandis que le service *RS-Non-Standard-Data* est utilisé pour *l'acheminement d'informations relatives à une transaction* ou pour *l'acheminement d'informations asynchrones*.

7.2.3.4.1 Transactions non normalisées

Les transactions non normalisées sont effectuées au moyen des unités PDU *rsNonStandardRequest* et *rsNonStandardResponse*, produites respectivement au moyen des primitives de service *RS-Non-Standard-Transaction-request* et *RS-Non-Standard-Transaction-response* et indiquées au moyen des primitives *RS-Non-Standard-Transaction-indication* et *RS-Non-Standard-Transaction-confirm*.

Dès réception d'une primitive *RS-Non-Standard-Transaction-request*, l'entité URST APE doit attribuer un identificateur à la transaction non normalisée et l'indiquer à l'application de réservation en lui envoyant une primitive *RS-Transaction-Allocation-indication*, comme pour des transactions normalisées. Les paramètres *Type d'acquittement* et *Acquittement explicite requis* contenus dans les primitives *RS-Non-Standard-Transaction-response* et *RS-Non-Standard-Transaction-indication* permettent à des transactions non normalisées d'être explicitement acquittées, acquittées ou non acquittées, selon leur type.

La Figure 7-8 montre l'utilisation du service *RS-Non-Standard-Transaction* pour des transactions non normalisées.

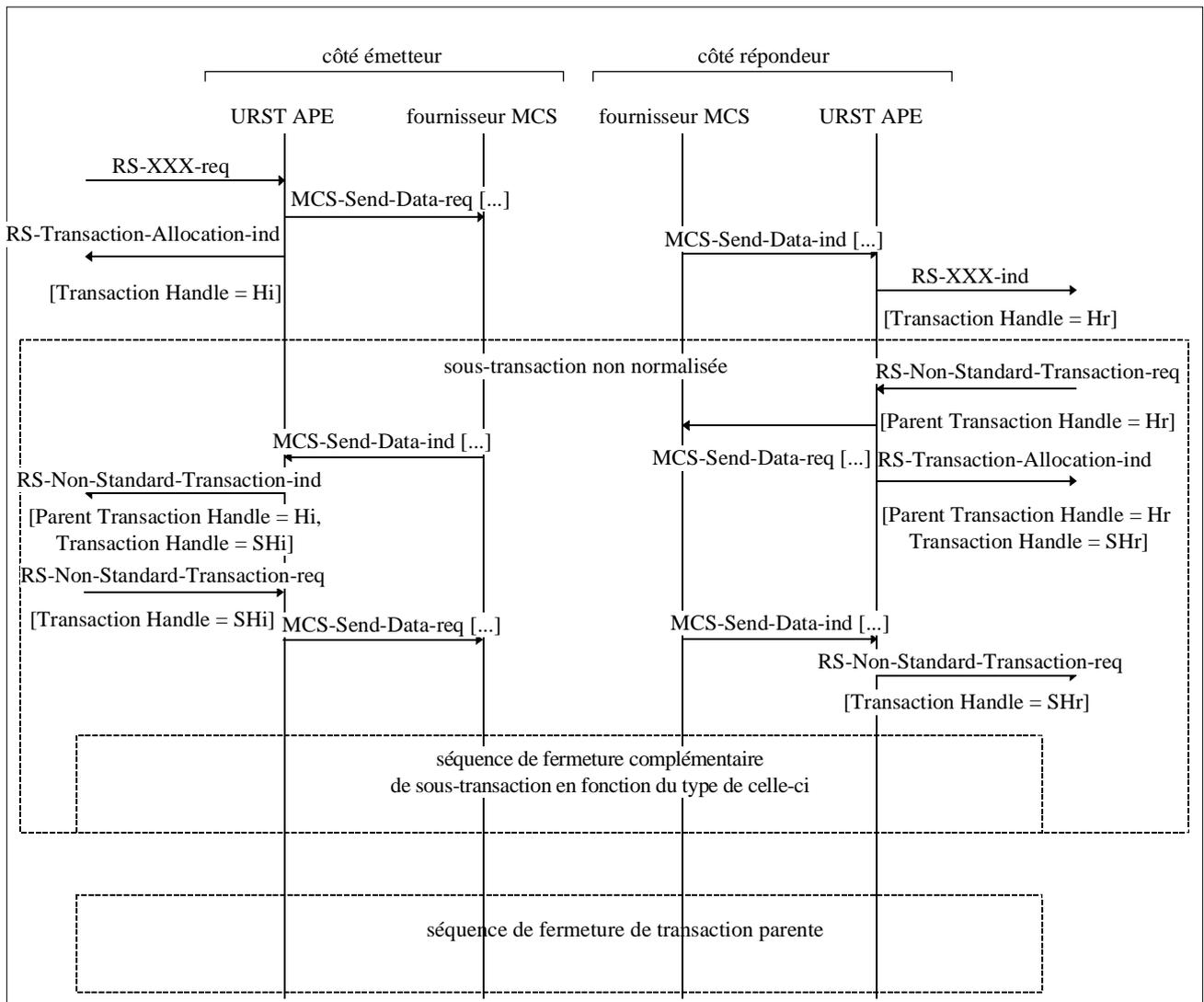


T1602150-97

Figure 7-8/T.135– Séquences types de transactions non normalisées

7.2.3.4.2 Sous-transactions non normalisées

Avant de répondre à une demande de transaction, l'application répondante peut avoir besoin d'autres éléments d'information de l'extrémité homologue. A cette fin, elle est autorisée à émettre une *sous-transaction non normalisée* avant de répondre à la transaction originale. Une sous-transaction non normalisée s'effectue de la même façon qu'une transaction non normalisée (voir 7.2.3.4.1) mais elle est également caractérisée par un identificateur de transaction parente qui l'associe à sa transaction d'origine. Le paramètre *Identificateur de transaction parente* contenu dans la primitive de service *RS-Non-Standard-Transaction* est prévu à cette fin. Les sous-transactions sont fermées de la même façon que les transactions (voir 7.2.3.2) et sont automatiquement abandonnées si leur transaction parente est fermée.



T1602160-97

Figure 7-9/T.135 – Séquence type d'une sous-transaction non normalisée

7.2.3.4.3 Acheminement d'informations relatives à une transaction

Pendant le déroulement d'une transaction ou d'une sous-transaction, les deux extrémités peuvent également envoyer des éléments d'information asynchrones, relatifs à cette transaction ou sous-transaction. Les informations asynchrones n'ont pas d'identificateur de transaction mais possèdent un identificateur de transaction parente qui doit être mis à la valeur de l'identificateur de la transaction

ou sous-transaction à laquelle il se rapporte. Les informations asynchrones sont envoyées dans l'unité PDU *rsNonStandardIndication* qui est produite au moyen de la primitive de service *RS-Non-Standard-Data-request*.

7.2.3.4.4 Acheminement d'informations asynchrones

En variante de l'acheminement d'informations relatives à une transaction, chacune des extrémités d'une connexion de réservation peut, à tout moment, envoyer des éléments d'information asynchrones, non associés à une transaction ou sous-transaction en cours. Cette opération est toujours effectuée au moyen de l'unité PDU *rsNonStandardIndication* sans identificateur de transaction parente, cette unité étant produite par le service *RS-Non-Standard-Data-request* sans insertion du paramètre *Identificateur de transaction parente*. Comme pour l'acheminement d'informations relatives à une transaction, le paramètre *protocolKey* peut être utilisé pour identifier un protocole ou un format de données.

Les mécanismes relatifs aux unités PDU non normalisées visent à faciliter les dialogues non normalisés entre un système de réservation et un utilisateur, lorsque les transactions normalisées selon la présente Recommandation ne sont pas suffisantes. Ils permettent par exemple à un système de réservation de mettre des ressources non normalisées à la disposition d'un utilisateur en engageant une sous-transaction d'une transaction de type *RS-Conference-Reserve*.

7.2.4 Autres comportements

7.2.4.1 Mode présidé

Les entités URST APE sont transparentes au mode présidé. En d'autres termes, elles doivent fonctionner de la même façon, que la conférence soit ou non en mode présidé.

7.2.4.2 Invocation distante

Les entités URST APE ne doivent pas utiliser le service *GCC-Application-Invoke-request*.

8 Description du service

8.1 Résumé du service

Le Tableau 8-1 énumère les primitives du service de transactions URST. Si applicable, les dernières colonnes indiquent si la transaction correspondante est du type non acquitté (U), acquitté (A) ou explicitement acquitté (EA).

Tableau 8-1/T.135 – Primitives du service de transactions URST

Primitive du service	Sous-paragraphes	Description	Type de transaction		
			U	A	EA
RS-Check-Reservation-Systems	8.2.2	consultation de la liste des systèmes de réservation présents dans la conférence	non applicable		
RS-Conference-Detail-Inquire	8.2.9	importation de la description complète d'une conférence programmée ou engagée	√		
RS-Conference-Cancel	8.2.7	annulation d'une conférence programmée		√	

Tableau 8-1/T.135 – Primitives du service de transactions URST (fin)

Primitive du service	Sous-paragraphes	Description	Type de transaction		
			U	A	EA
RS-Conference-Check-Availability	8.2.10	obtention d'une liste de dates de disponibilité avec intervalles entre conférences décrites en termes de largeur de bande, nombre de sites participants et tous autres paramètres relatifs à l'attribution des ressources	√		
RS-Conference-List-Inquire	8.2.8	consultation d'une liste abrégée des conférences programmées ou engagées	√		
RS-Conference-Modify	8.2.6	modification d'une conférence programmée		√	√
RS-Conference-Reserve	8.2.5	programmation dans le temps d'une conférence		√	√
RS-Connect	8.2.3	établissement d'une connexion de réservation	non applicable		
RS-Disconnect	8.2.4	libération d'une connexion de réservation	non applicable		
RS-Non-Standard-Data	8.2.16	échange de données de transactions non normalisées (acheminement d'informations relatives à une transaction et acheminement d'informations asynchrones)	non applicable		
RS-Non-Standard-Request	8.2.15	transactions ou sous-transactions non normalisées	√	√	√
RS-Site-Delete	8.2.13	suppression d'un enregistrement de site dans une base de données de système de réservation		√	
RS-Site-Directory-Inquire	8.2.14	consultation d'une liste de sites et d'informations associées, sur la base d'un ensemble de valeurs-critères tels que l'intervalle alphabétique, l'emplacement géographique, etc.	√		
RS-Site-Modify	8.2.12	modification d'un enregistrement de site dans une base de données de système de réservation		√	
RS-Site-Record	8.2.11	création d'un enregistrement de site dans une base de données de système de réservation		√	
RS-Transaction-Acknowledge	8.2.18	acquiescement d'une transaction de réservation	non applicable ^{a)}		
RS-Transaction-Allocation	8.2.17	attribution d'un identificateur de transaction	non applicable		
RS-Transaction-Cancel	8.2.19	annulation d'une transaction	non applicable		
RS-Transaction-Error	8.2.20	service de signalisation d'erreur	non applicable		

^{a)} Primitive utilisée pour fermer des transactions de type A ou EA.

8.2 Description du service

8.2.1 Règles applicables

8.2.1.1 Identificateurs généraux

De nombreux paramètres de protocole ou de service ont besoin d'identificateurs pour que l'on puisse y faire référence dans des transactions du service URST. Par exemple, une conférence normalement réservée est rappelée au moyen d'une référence de conférence/réservation contenue dans d'ultérieures modifications de la réservation (voir au 8.2.5 le paramètre *Référence de conférence/réservation*). Sauf spécification contraire, tout paramètre de transaction URST jouant le rôle d'identificateur reflète un paramètre de niveau protocolaire et du type défini en ASN.1 *GenericID* (voir le paragraphe 9), qui offre le choix entre un entier non signé et une chaîne alphanumérique.

8.2.1.2 Codes de résultat normalisés

Les formes réponse de la plupart des primitives du service URST comportent un paramètre *Code de résultat* qui signale une erreur locale ou une erreur notifiée par l'extrémité homologue.

La liste suivante énumère un ensemble de codes de résultat pouvant être utilisés comme forme de réponse dans toute primitive du service URST contenant le paramètre *Code de résultat*: "Succès", "Pointeur de connexion erroné", "Pointeur de transaction erroné", "Paramètres erronés", "Valeurs paramétriques erronées", "Panne locale", "Absence de réponse de l'homologue", "Refus de transaction par l'homologue", "Code de panne par défaut".

Pour chaque primitive du service URST, l'ensemble des codes de résultat normalisés supplémentaires est énuméré dans le sous-paragraphe concerné.

8.2.2 Primitive *RS-Check-Reservation-Systems*

Cette primitive est utilisée par une application UT-RA pour consulter la liste des applications UT-RA inscrites à la session d'enregistrement URST. Pour traiter une demande *RS-Check-Reservation-Systems-request*, une entité URST APE doit envoyer au fournisseur de commande GCC local une demande *GCC-Application-Roster-Inquire* spécifiant la clé de session d'enregistrement URST. Elle doit ensuite filtrer les informations incluses dans la primitive correspondante *GCC-Application-Roster-Inquire-confirm* afin de construire la primitive *RS-Check-Reservation-Systems-confirm* appropriée pour l'application UT-RA.

Si le fournisseur GCC local ne prend pas en charge le service *GCC-Application-Roster-Inquire*, l'entité URST APE est chargée de tenir à jour les informations nécessaires au moyen de chaque primitive *GCC-Roster-Report-indication* reçue du fournisseur GCC local.

Les paramètres de cette primitive sont énumérés dans le Tableau 8-1 bis.

Tableau 8-1 bis/T.135 – Primitive *RS-Check-Reservation-Systems*

Paramètre	Req	Cnf
liste des systèmes de réservation		O

Le paramètre *Liste de systèmes de réservation* se compose d'une liste contenant autant d'entrées qu'il existe d'entités URST APE du type nœud RSN inscrites à la session d'enregistrement URST. L'absence de ce paramètre dans la primitive de confirmation indique que la liste est vide. Chaque entrée contient les paramètres énumérés dans le Tableau 8-2.

Tableau 8-2/T.135 – Paramètre *Liste des systèmes de réservation* – Structure élémentaire

Paramètre	Cnf
identificateur de nœud RSN	M
identificateur d'application de réservation	M
nom du système de réservation	M
identificateur du système de réservation	C

L'élément *Identificateur de nœud RSN* est l'identificateur du nœud serveur de réservations (RSN), conformément à la définition d'un identificateur de nœud, donnée dans la Recommandation T.124.

L'élément *Identificateur d'application de réservation* est l'identificateur de l'utilisateur du service MCS associé à l'application de réservation RSN-RA inscrite dans ce nœud, conformément à la définition d'un identificateur d'utilisateur de service MCS, donnée dans la Recommandation T.124.

L'élément *Nom du système de réservation* est la chaîne contenue dans les données associées à la capacité non condensable *Nom du système de réservation* qui est définie au 7.2.1.

L'élément *Identificateur du système de réservation* est l'identificateur contenu dans les données associées à la capacité non condensable *Identificateur du système de réservation* qui est définie au 7.2.1. Cet élément est obligatoire si la capacité a été fournie par l'application RSN-RA.

8.2.3 Primitive *RS-Connect*

Cette primitive est utilisée pour ouvrir une connexion de réservation. Elle permet à l'utilisateur d'obtenir l'entrée dans le système de réservation. Ses paramètres sont énumérés dans le Tableau 8-3.

Tableau 8-3/T.135 – Primitive *RS-Connect*

Paramètre	Req	Ind	Rsp	Cnf
identificateur d'application de réservation	C			C(=RQ)
identificateur d'abonné	O	C(=)	C(=)	C(=)
données d'authentification	O	O(=)	O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
pointeur de connexion	C(=CF)	M	M(=IN)	M
protocoles supérieurs acceptés	O	O(=)	O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

Le paramètre *Identificateur d'application de réservation* permet à l'utilisateur de viser un système de réservation particulier. Il doit correspondre à l'un des identificateurs d'application de réservation obtenus par le service *RS-Check-Reservation-Systems*. Dans les primitives de réponse et de confirmation, ce paramètre permet à l'application UT-RA requérante de corrélérer la réponse avec la demande originale. Ce paramètre est obligatoire dans les premières formes des primitives de demande et de confirmation associées à cette connexion. Il est absent dans toutes les autres primitives de type *RS-Connect* concernant cette connexion.

Le paramètre *Identificateur d'abonné* désigne l'utilisateur auprès du système de réservation. Il est du type "identificateur général" qui est défini au 8.2.1. Il est obligatoire dans les premières formes des primitives de demande et de confirmation associées à cette connexion, s'il a été fourni dans la primitive *RS-Connect-request* émise par l'application UT-RA; sinon, il est absent de ces primitives.

Il est facultatif dans la première primitive *RS-Connect-request*, de façon que les utilisateurs non enregistrés puissent essayer de se connecter. Lorsque le paramètre *Identificateur d'abonné* n'est pas fourni, il appartient au système de réservation d'accepter ou de refuser la connexion. Dans les premières formes des primitives de réponse et de confirmation, il permet à l'application UT-RA de corréler la réponse avec la demande originale. Il est absent dans toutes les autres primitives de type *RS-Connect* concernant cette connexion.

Le paramètre *Données d'authentification* peut prendre plusieurs formes selon le numéro de l'étape d'émission ou de réception de la primitive de service, selon que le système d'authentification est unilatéral ou réciproque, et selon les algorithmes d'authentification choisis. Ce paramètre se subdivise en deux parties logiques, chacune étant facultative dans chaque forme de la primitive. La première partie est réservée aux données fournies à l'extrémité homologue et aux données qui lui sont demandées. Les données fournies à l'extrémité homologue peuvent consister soit en un mot de passe communiqué directement à l'extrémité homologue soit en une épreuve préalable requise par cette extrémité, tandis que la deuxième partie consiste en une demande d'épreuve. Plusieurs séquences de passe sont autorisées, comme indiqué au 7.2.2.3. On trouvera au paragraphe 9 la définition protocolaire détaillée de ce paramètre.

Le paramètre *Code de résultat* peut prendre les valeurs normalisées supplémentaires suivantes: "*Authentification en cours*" et "*Echec d'authentification*". Ce paramètre est toujours obligatoire dans les formes réponse et confirmation de la primitive mais doit être mis à la valeur "*Authentification en cours*" pour toutes ces formes sauf pour les dernières.

Le paramètre *Message de résultat* est une chaîne facultative de caractères qui peut être utilisée afin de compléter le code de résultat. Sa présence est facultative dans les dernières formes réponse et confirmation de la primitive associées à cette connexion et sa présence n'est pas requise dans les précédentes formes.

Le paramètre *Pointeur de connexion* identifie la connexion comme indiqué au 7.2.2. Il est absent dans la première primitive *RS-Connect-request* émise par l'application UT-RA et il est obligatoire dans toutes les autres primitives relatives à cette connexion.

NOTE – Les pointeurs de connexion permettent théoriquement à un utilisateur particulier d'exécuter de multiples entrées simultanées dans un même système de réservation. Toutefois, cette capacité est hors du domaine d'application de la présente Recommandation et il appartient à chaque système de réservation de l'accepter ou de la refuser.

Le paramètre *Protocoles supérieurs acceptés* est une liste de clés de protocoles permettant à chaque extrémité d'indiquer la liste des protocoles ou des formats de données qu'elle prend en charge au moyen du service *RS-Data*. Chaque clé de protocole est soit un identificateur ASN.1 de type OBJECT IDENTIFIER conforme à une Recommandation, soit un protocole normalisé ou non normalisé, ou en variante, un identificateur non normalisé utilisant les conventions de codage de la Recommandation H.221.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions qui sont hors du domaine d'application de la présente Recommandation.

8.2.4 Primitive *RS-Disconnect*

Tableau 8-4/T.135 – Primitive *RS-Disconnect*

Paramètre	Req	Ind
pointeur de connexion	M	M
code de cause	M	M(=)
message de cause	O	O(=)
données d'utilisateur	O	O(=)

Cette primitive est la dernière étape d'une connexion de réservation. Un terminal de réservation ne se dissociera pas d'une conférence avant d'avoir fermé toutes les connexions de réservation ouvertes par son intermédiaire.

Le paramètre *Pointeur de connexion* identifie la connexion. Voir le 7.2.2.

Le paramètre *Code de cause* indique la cause de la déconnexion. Les valeurs normalisées sont "*Déconnexion déclenchée par l'homologue*", "*Homologue déconnecté de la conférence*" et "*Code de déconnexion par défaut*".

Le paramètre *Message de cause* est une chaîne de caractères facultative qui peut être utilisée pour compléter le code de cause.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions qui sont hors du domaine d'application de la présente Recommandation.

8.2.5 Primitive *RS-Conference-Reserve*

Cette primitive définit une transaction de réservation de conférence. Ses paramètres sont énumérés dans le Tableau 8-5.

Tableau 8-5/T.135 – Primitive *RS-Conference-Reserve*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
nom de la conférence (RC)	M	M(=)	O	O(=)
description de la conférence	O	O(=)	O(=)	O(=)
mode de ponts MCU en cascade	O	O(=)	O(=)	O(=)
commande de présidence audiovisuelle	O	O(=)	O(=)	O(=)
mot de passe de président audiovisuel	O	O(=)	O(=)	O(=)
organisateur T.120 externe	O	O(=)	O(=)	O(=)
mode présidé T.120	O	O(=)	O(=)	O(=)
liste de privilèges de président T.120	O	O(=)	O(=)	O(=)
liste de privilèges communs T.120	O	O(=)	O(=)	O(=)
mot de passe de président T.120	O	O(=)	O(=)	O(=)
mot de passe commun	O	O(=)	O(=)	O(=)
débit utile par défaut (RC)	M	M(=)	O	O(=)

Tableau 8-5/T.135 – Primitive *RS-Conference-Reserve* (fin)

Paramètre	Req	Ind	Rsp	Cnf
méthode de commutation vidéo (RC)	O	O(=)	O	O(=)
format vidéo initial (RC)	O	O(=)	O	O(=)
algorithme vidéo initial (RC)	O	O(=)	O	O(=)
algorithme audio initial (RC)	O	O(=)	O	O(=)
politique de listes d'attente	O	O(=)		
mode de facturation (RC)	O	O(=)	O	O(=)
compte de facturation de l'organisateur	O	O(=)	O(=)	O(=)
identificateur du détenteur de la conférence	O	O(=)	O(=)	O(=)
date	M	M(=)	O(=)	O(=)
heure	M	M(=)	O(=)	O(=)
durée	M	M(=)	O(=)	O(=)
liste des sites de réservation (RC)	M	M(=)	O	O(=)
méthode par défaut de raccordement des sites	M	M(=)	O(=)	O(=)
mot de passe de modificateur de conférence	O	O(=)	O(=)	O(=)
référence de conférence/réservation			C	C(=)
acquiescement explicite requis			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

L'absence de *paramètres contrôlés du système de réservation* (marqués du symbole "(RC)", voir 7.2.3.3) dans les formes réponse et confirmation de la primitive doit être interprétée comme un acquiescement implicite par le système de réservation. En variante, celui-ci peut fournir de tels paramètres dans la forme de réponse de la primitive, même s'il ne modifie pas leurs valeurs, afin d'acquiescer explicitement les paramètres requis.

Le paramètre *Pointeur de connexion* identifie la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* identifie la transaction. Voir 7.2.3.

Les paramètres *Nom de la conférence* et *Description de la conférence* sont des chaînes de caractères qui nomment et résument la conférence. En exploitation, ces paramètres s'appliqueront aux deux paramètres *Nom de la conférence* et *Description de la conférence* définis dans la Recommandation T.124. Dans ce cas, ce paramètre est obligatoire dans les formes réponse et confirmation de la primitive.

Le paramètre *Mode de ponts MCU en cascade* indique le type de mise en cascade qui sera effectuée au moment de l'engagement de la conférence entre les ponts MCU du domaine de réservation du système requis et les ponts MCU externes (c'est-à-dire les ponts MCU qui sont inclus dans la liste des sites de réservation – voir la description du paramètre *Liste des sites de réservation* au 8.2.5.2 – ou qui leur sont associés). Il indique également le rôle qui sera joué par ces points, conformément aux règles spécifiées dans la Recommandation H.243. Les valeurs normales de ce paramètre sont "*Cascade simple*", "*Pont maître externe*", "*Pont maître interne*" ou "*Pont non spécifié*". La valeur "*Cascade simple*" indique que la mise en cascade, si elle est effectuée, doit être de la forme simple,

c'est-à-dire que les ponts MCU sélectionnés dans le domaine de réservation requis doivent se comporter comme des sites simples vis-à-vis des ponts externes. La valeur "*Pont maître externe*" indique que la mise en cascade doit se fonder sur une relation maître/esclave et que le pont MCU maître est externe. La valeur "*Pont maître interne*" indique que la mise en cascade doit se fonder sur une relation maître/esclave et que le pont MCU maître doit appartenir au domaine de réservation. La valeur "*Pont non spécifié*" indique que le mode de mise en cascade doit être décidé au moment de la connexion. L'absence de ce paramètre équivaut à sélectionner la valeur "*Pont non spécifié*".

Le paramètre *Commande de présidence audiovisuelle* est un fanion booléen qui indique que les ponts MCU qui seront sélectionnés dans le domaine de réservation doivent prendre en charge la commande présidentielle H.243. L'absence de ce paramètre équivaut à l'absence de sa demande.

Le paramètre *Mot de passe de président audiovisuel* est un mot de passe facultatif qui spécifie le mot de passe H.243 qui doit être utilisé au moment de la connexion par un nœud qui a l'intention de prendre la présidence H.243. L'absence de ce paramètre ou l'envoi d'une chaîne "néant" signifie que la présidence H.243 n'est pas protégée par mot de passe.

NOTE – Comme spécifié dans la Recommandation H.243, la commande présidentielle H.243 et la mise en cascade H.243 sont des fonctions distinctes. En d'autres termes, lorsque la liste des sites de réservation comporte des ponts MCU externes, le site de présidence H.243 ne présidera certainement pas l'ensemble de la conférence H.320. L'ensemble des équipements qui permettront l'application de la position présidentielle H.243 d'un terminal particulier dépend du type de mise en cascade effectuée entre chaque sous-domaine H.320. En général, lorsqu'une réservation est effectuée élément par élément (c'est-à-dire lorsque la liste des sites de réservation comporte des ponts MCU externes), il appartient aux réservateurs d'assurer la compatibilité entre les éléments distincts. La présente Recommandation ne spécifie pas les combinaisons particulières qu'un système de réservation doit rejeter. L'acceptation ou le rejet d'une demande de réservation est du ressort de chaque système de réservation.

Le paramètre *Organisateur T.120 externe* est un fanion booléen qui indique si la conférence T.120 sera créée par un site inclus dans le paramètre *Liste des sites de réservation* (ou s'il sera attaché à l'un de ces sites lors de l'engagement de la conférence). L'absence de ce paramètre équivaut à indiquer qu'il n'y a pas d'organisateur externe.

Le paramètre *Mode présidé T.120* est un fanion booléen qui indique si la conférence doit être en mode présidé T.120. Ce paramètre n'est pas applicable si le paramètre *Organisateur T.120 externe* est mis à la valeur TRUE. L'absence de ce paramètre tandis que le paramètre *Organisateur T.120 externe* indique qu'il n'y a pas d'organisateur externe équivaut à programmer la conférence en mode non présidé.

Le paramètre *Liste de privilèges de président T.120* indique facultativement les privilèges du président conformément à la Recommandation T.124, lorsque le mode présidé est programmé. Ce paramètre n'est applicable que si le paramètre *Mode présidé T.120* est applicable et mis à TRUE. L'absence de ce paramètre lorsqu'il est applicable indique que la conférence doit être créée conformément à l'ensemble de règles décrit dans la Recommandation T.124.

Le paramètre *Liste de privilèges communs T.120* indique facultativement les privilèges des nœuds autres que celui du président si la conférence doit être créée en mode présidé, ou les privilèges de n'importe quel nœud si la conférence doit être créée en mode non présidé. Ce paramètre n'est applicable que si le paramètre *Organisateur T.120 externe* indique qu'il n'y a pas d'organisateur externe. L'absence de ce paramètre lorsqu'il est applicable indique que la conférence doit être créée conformément à l'ensemble de règles décrit dans la Recommandation T.124.

Le paramètre *Mot de passe de président T.120* est un mot de passe facultatif qui spécifie le mot de passe T.120 particulier qui doit être utilisé au moment de la connexion par un nœud qui a l'intention de prendre la présidence T.120. Ce paramètre n'est applicable que si le paramètre *Mode présidé T.120* est applicable et indique que la conférence doit être créée en mode présidé. La

propagation d'une chaîne "néant" équivaut à ne pas fournir ce paramètre, c'est-à-dire à ne pas protéger par mot de passe la conférence.

NOTE – Les interactions entre présidence H.243 et présidence T.120 sont spécifiées dans la Recommandation H.243.

Le paramètre *Mot de passe commun* indique le mot de passe pour l'entrée des nœuds communs dans la conférence, c'est-à-dire pour les nœuds à présidence non audiovisuelle et non T.120 (à noter que ce mot de passe servira également à ces nœuds dans le cas d'une présidence audiovisuelle ou T.120 non protégée par mot de passe). La propagation d'une chaîne "néant" équivaut à ne pas fournir ce paramètre, c'est-à-dire à ne pas protéger par mot de passe la conférence pour les nœuds communs.

Le paramètre *Débit utile par défaut (RC)* indique le débit de transmission nominal qui est demandé pour la conférence. Ce paramètre représente la largeur de bande globale dont chaque terminal doit disposer pour les besoins combinés de la mise en trames H.221, du formatage au besoin, des signaux audio, vidéo et de données. Ce paramètre peut indiquer de multiples valeurs, comprises entre 64 kbit/s et 1920 kbit/s. Il peut également être remplacé, au niveau du site, par le paramètre *Débit utile du site* contenu dans le paramètre *Liste des sites de réservation* (voir 8.2.5.2).

NOTE – Ce paramètre donne au système de réservation une indication par défaut pour l'attribution des ressources en accès. La largeur de bande réellement utilisée lors d'une conférence engagée pourra cependant être différente car les terminaux connectés à un réseau numérique soumis à des restrictions peuvent être inclus dans le paramètre *Liste des sites de réservation*. Des renseignements spécifiques concernant un site particulier, comme la compatibilité $N \times 64-64 \times N$ (voir la commande *SM-comp* dans la Recommandation H.242), les types de réseau d'accès correspondant, etc. sont donnés au cours de la phase d'enregistrement du site (voir 8.2.11). Sur la base de ces renseignements, le système de réservation déterminera le type de connexion nécessaire ainsi que l'applicabilité de la conférence.

Le paramètre *Méthode de commutation vidéo (RC)* indique l'algorithme de commutation vidéo préféré. Ce paramètre a une structure qui varie selon la méthode choisie. Il contient de toute façon un premier sous-paramètre *Sélecteur d'algorithme* qui désigne l'algorithme de commutation vidéo. Les algorithmes possibles sont les suivants: "*Détection d'activité vocale par défaut*", "*Rotation périodique*", "*Site diffuseur fixe*", "*Mosaïque des derniers orateurs*", "*Mosaïque des listes fixes*". L'absence du paramètre *Méthode de commutation vidéo* dans les formes réponse/indication de la primitive équivaut à sélectionner le mode de détection d'activité vocale par défaut. Les Tableaux 8-6 à 8-10 du 8.2.5.1 décrivent la structure pour chaque cas.

Le paramètre *Format vidéo initial (RC)* peut, facultativement, être utilisé par le demandeur pour indiquer un format vidéo préféré au début de la conférence. Les valeurs normalisées de ce paramètre sont: "*CIF*", "*QCIF*", "*SIF*", "*SQCIF*" et "*Choix à l'engagement*". L'absence de ce paramètre équivaut à choisir l'option "*Choix à l'engagement*", c'est-à-dire que le choix du format vidéo est laissé à la discrétion de l'ensemble des ponts MCU ou des autres dispositifs de gestion qui seront mis à contribution.

Le paramètre *Algorithme vidéo initial (RC)* peut, facultativement, être utilisé par le demandeur pour indiquer un algorithme de codage vidéo préféré au début de la conférence. Les valeurs normalisées pour chaque élément de la liste de ce paramètre sont: "*H.261*", "*H.262*", "*H.263*", et "*Choix à l'engagement*". L'absence de ce paramètre équivaut à choisir l'option "*Choix à l'engagement*", c'est-à-dire que le choix du format vidéo est laissé à la discrétion de l'ensemble des ponts MCU ou des autres dispositifs de gestion qui seront mis à contribution.

Le paramètre *Algorithme audio initial (RC)* peut, facultativement, être utilisé par le demandeur pour indiquer un algorithme de codage audio préféré au début de la conférence. Les valeurs normalisées de ce paramètre sont: "*G.711-loi A*", "*G.711-loi μ* ", "*G.722*", "*G.723*", "*G.728*", "*G.729*", "*Audio MPEG*", et "*Choix à l'engagement*". L'absence de ce paramètre équivaut à choisir l'option "*Choix à*

l'engagement", c'est-à-dire que le choix du format vidéo est laissé à la discrétion de l'ensemble des ponts MCU ou des autres dispositifs de gestion qui seront mis à contribution.

Le paramètre *Politique de listes d'attente* est un indicateur qui permet au demandeur de définir la façon dont le système de réservation doit traiter la réservation si celle-ci ne peut pas être entièrement acceptée en raison d'un manque de ressources en termes de ponts MCU ou d'éléments de réseau. Ce paramètre peut prendre les valeurs suivantes: "*Liste de tous les sites en attente*", "*Traitement au mieux*" ou "*Pas de liste d'attente*". La valeur "*Liste de tous les sites en attente*" sert à demander que l'ensemble de la conférence soit placé dans la liste d'attente, tandis que la valeur "*Traitement au mieux*" signifie que la conférence doit être réservée aux participants qui trouvent des ressources tandis que l'ensemble des sites de réservation restants doit être placé dans la liste d'attente. Lorsque la valeur "*Traitement au mieux*" est sélectionnée, l'ordre des sites dans le paramètre *Liste des sites de réservation* détermine leur priorité relative quant à leur position dans la liste d'attente. L'absence de ce paramètre équivaut à choisir la valeur "*Pas de liste d'attente*", c'est-à-dire à ne demander aucun service de liste d'attente. Dans les formes réponse/confirmation de la primitive, la valeur spéciale "*Liste d'attente*" du paramètre *Code de résultat* doit être renvoyée si certains sites de réservation ou toute la conférence ont été placés dans la liste d'attente. Chaque site placé dans la liste d'attente doit ensuite être indiqué au moyen du paramètre facultatif *Indicateur de liste d'attente* de la structure décrivant un élément du paramètre *Liste des sites de réservation* (voir 8.2.5.2). Le paramètre *Code de résultat* mis à la valeur "*Liste d'attente*" tandis que le paramètre *Indicateur de liste d'attente* est présent pour tous les sites de réservation ou pour aucun d'eux indique que l'ensemble de la conférence a été mis en liste d'attente.

NOTE 1 – Le placement d'un site en liste d'attente ne concerne que les ponts MCU et les ressources réseau. La réservation du terminal correspondant (s'il s'agit d'un terminal qui peut être réservé) est une fonction complètement distincte.

NOTE 2 – Lorsqu'un site ou un ensemble de sites de réservation sont placés dans la liste d'attente, la façon dont les utilisateurs sont ensuite avertis de la décision finale est hors du domaine d'application de la présente Recommandation et laissée à la discrétion de chaque système de réservation (c'est-à-dire le fournisseur de services). La prise en charge de l'une quelconque des options du service de liste d'attente par un système de réservation est hors du domaine d'application de la présente Recommandation.

Le paramètre *Mode de facturation (RC)* indique facultativement la méthode de facturation demandée. Ce paramètre peut prendre les valeurs normalisées suivantes: "*Organisateur*", "*Participants*", "*Mode mixte*" ou "*Mode de facturation implicite*", en plus des valeurs non normalisées. La valeur "*Organisateur*" signifie que le coût total de la conférence doit être facturé à l'organisateur (c'est-à-dire à la personne ou à l'entité désignée par le paramètre *Identificateur d'abonné* de la primitive *RS-Connect*). La valeur "*Participants*" signifie que le coût total de la conférence doit être partagé entre les sites participants et la valeur "*Mode mixte*" indique que chacun des sites participants et l'organisateur doivent être facturés selon un barème défini par le paramètre *Quote-part de facturation des participants* décrit au 8.2.5.2. La valeur "*Mode de facturation implicite*" permet au demandeur de faire référence à un mode de facturation prédéfini dont le barème a fait l'objet d'un accord préalable entre l'abonné et le fournisseur de services par des moyens qui sont hors du domaine d'application de la présente Recommandation. L'absence de ce paramètre équivaut à sélectionner l'option "*Mode de facturation implicite*".

Le paramètre *Compte de facturation de l'organisateur* désigne un compte de facturation particulier que le fournisseur de services doit utiliser pour la facturation de la conférence, si l'abonné connecté a accès à plusieurs comptes. Ce paramètre est du type "identificateur général" défini au 8.2.1. La non-fourniture de ce paramètre équivaut à la sélection d'un compte de facturation implicite dont la prédétermination a été effectuée par des moyens hors du domaine d'application de la présente Recommandation.

Le paramètre *Identificateur du détenteur de la conférence* est un identificateur d'abonné en variante qui permet au demandeur de réserver la conférence pour le compte d'un abonné tiers. Par défaut (c'est-à-dire lorsque ce paramètre n'est pas fourni), le détenteur de la conférence est implicitement désigné par le paramètre *Identificateur d'abonné* fourni au moment de la connexion (c'est-à-dire contenu dans la primitive *RS-Connect*).

Les paramètres *Date*, *Heure* et *Durée* indiquent respectivement la date, l'heure d'ouverture et la durée de la conférence. Le temps doit être exprimé en unités UTC.

Le paramètre *Liste des sites de réservation (RC)* est une liste structurée de sites participants. Chaque élément a la structure décrite au 8.2.5.2.

Le paramètre *Méthode par défaut de raccordement des sites* indique la méthode par défaut qui sera utilisée par un site quelconque qui figure dans la liste des sites de réservation pour entrer dans la conférence (site appelant/site appelé), sauf respecification au niveau du site par le paramètre *Liste des sites de réservation* (voir 8.2.5.2). Les valeurs possibles sont "*Site appelé*" et "*Site appelant*".

Le paramètre *Mot de passe de modificateur de conférence* est un mot de passe facultatif qui, une fois défini, permet à un abonné de système de réservation ou à des abonnés non enregistrés, dont les privilèges d'accès ne leur permettent normalement pas de le faire, de modifier ou d'annuler la réservation émise. L'utilisateur devra fournir ce mot de passe en même temps que la référence de conférence/réservation de façon à la modifier ou à l'annuler (voir ci-dessous et aux 8.2.6 et 8.2.7 le paramètre *Référence de conférence/réservation*).

Le paramètre *Référence de conférence/réservation* désigne la conférence enregistrée dans le système de réservation. Il est du type "identificateur général" défini au 8.2.1. Toute nouvelle tentative de modification ou d'annulation de la conférence doit être effectuée au moyen de cette référence. Ce paramètre est attribué par le système de réservation et son retour dans les formes réponse et confirmation de la primitive est obligatoire sauf si la réservation est refusée.

Le paramètre *Acquittement explicite requis* est un fanion booléen qui indique si l'extrémité d'origine de la transaction doit acquitter celle-ci explicitement au moyen du service *RS-Transaction-Acknowledge*. Voir 7.2.3.

Le paramètre *Code de résultat* peut prendre les valeurs normalisées et supplémentaires "*Liste d'attente*" et "*Absence de ressources disponibles*".

Le paramètre *Message de résultat* est une chaîne de caractères facultative qui peut être utilisée pour compléter le code de résultat.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions hors du domaine d'application de la présente Recommandation.

8.2.5.1 Paramètre *Méthode de commutation vidéo*

**Tableau 8-6/T.135 – Paramètre *Méthode de commutation vidéo* –
Sous-paramètre *Détection d'activité vocale par défaut***

Paramètre	Req	Ind	Rsp	Cnf
sélecteur d'algorithme (= " <i>Détection d'activité vocale par défaut</i> ")	M	M(=)	O	O(=)

La méthode de détection d'activité vocale par défaut diffuse le site locuteur à tous les autres sites. Le site locuteur obtient l'image du site locuteur précédent. Lorsque ce mode est réservé, les sites peuvent changer le mode en temps réel, comme spécifié dans les Recommandations H.243 et H.245.

**Tableau 8-7/T.135 – Paramètre *Méthode de commutation vidéo* –
Sous-paramètre *Rotation périodique***

Paramètre	Req	Ind	Rsp	Cnf
sélecteur d'algorithme ("rotation périodique")	M	M(=)	O	O(=)
période de rotation	M	M(=)	O	O(=)

La méthode de rotation périodique diffuse alternativement à chaque site de la conférence pendant une période donnée. Le paramètre *Période de rotation* exprime en secondes la durée pendant laquelle un site est diffusé avant commutation.

**Tableau 8-8/T.135 – Paramètre *Méthode de commutation vidéo* –
Sous-paramètre *Site diffuseur fixe***

Paramètre	Req	Ind	Rsp	Cnf
sélecteur d'algorithme ("Site diffuseur fixe")	M	M(=)	O	O(=)
identificateur de site diffuseur	M	M(=)		

La méthode du site diffuseur fixe définit le site identifié facultativement par le paramètre *Identificateur de site diffuseur* comme étant un site diffuseur permanent. S'il est présent, l'identificateur de site diffuseur doit correspondre à l'un des identificateurs de site indiqués dans le paramètre *Liste des sites de réservation*.

**Tableau 8-9/T.135 – Paramètre *Méthode de commutation vidéo* –
Sous-paramètre *Mosaïque des derniers orateurs***

Paramètre	Req	Ind	Rsp	Cnf
sélecteur d'algorithme ("Mosaïque des derniers orateurs")	M	M(=)	O	O(=)
nombre de sources	M	M(=)	O	O(=)

La méthode de la mosaïque des derniers orateurs diffuse une image mixte composée du site locuteur et des derniers (*Nombre de sources* – 1) sites locuteurs. Le site locuteur reçoit une image mixte composée du nombre de derniers sites locuteurs indiqué par le paramètre *Nombre de sources*. Ce nombre ne doit pas être supérieur à celui des sites figurant dans le paramètre *Liste des sites de réservation*.

**Tableau 8-10/T.135 – Paramètre *Méthode de commutation vidéo* –
Sous-paramètre *Mosaïque des listes fixes***

Paramètre	Req	Ind	Rsp	Cnf
sélecteur d'algorithme ("Mosaïque des listes fixes")	M	M(=)	O	O(=)
liste des identificateurs de site	M	M(=)	O	O(=)

La méthode de la mosaïque des listes fixes diffuse une image mixte composée des sites indiqués par le paramètre *Liste des identificateurs de site*, qui est une liste dont chaque élément doit correspondre à l'un des identificateurs de site indiqués dans le paramètre *Liste des sites de réservation*. Le nombre de sources de la mosaïque est déterminé par les dimensions du paramètre *Liste des identificateurs de site*.

NOTE – Etant donné qu'aucune commande du signal BAS n'existe dans la Recommandation H.242 pour sélectionner en temps réel l'une des méthodes suivantes: "Rotation périodique", "Site diffuseur fixe", "Mosaïque des derniers orateurs" et "Mosaïque des listes fixes", la sélection ou la désélection d'une telle méthode ne peut être effectuée autrement que par l'obtention d'une entrée dans le système de réservation en cours de conférence. Le projet de Recommandation T.130 pourra inclure des commandes en temps réel pour ces modes. Pour une conférence sans participant T.120 ou sans participant T.120 possédant la capacité d'accès au système de réservation en cours de conférence, aucun terminal de conférence ne possédera la capacité de sélectionner ou désélectionner ces algorithmes de commutation vidéo conformément à la présente Recommandation.

8.2.5.2 Paramètre *Liste des sites de réservation*

Chaque élément de cette liste a la structure indiquée dans le Tableau 8-11.

Tableau 8-11/T.135 – Paramètre Liste des sites de réservation – Structure élémentaire

Paramètre	Req	Ind	Rsp	Cnf
identificateur de site	M	M(=)	M(=)	M(=)
temps d'entrée d'un site (RC)	O	O(=)	O	O(=)
durée d'un site (RC)	O	O(=)	O	O(=)
débit utile d'un site (RC)	O	O(=)	O	O(=)
méthode d'entrée d'un site (RC)	O	O(=)	O	O(=)
adresse réseau attribuée			C	C(=)
indicateur de liste d'attente			O	O(=)
quote-part de facturation des participants (RC)	O	O(=)	O	O(=)
liste des participants à la conférence	O	O(=)	O(=)	O(=)

Ce paramètre peut contenir des *paramètres contrôlés par le système de réservation* (marqués par les lettres "(RC)", voir 7.2.3.3), qui suivent les règles déjà indiquées.

Le paramètre *Identificateur de site* est une référence au site qui a été attribuée au système de réservation lorsque l'application de réservation cliente a enregistré ce site au moyen du service *RS-Site-Record* ou par tout autre moyen hors du domaine d'application de la présente Recommandation. Ce paramètre est du type *identificateur général* défini au 8.2.1.

Les paramètres facultatifs *Temps d'entrée d'un site (RC)* et *Durée d'un site (RC)* permettent de ne programmer le site que pour une partie de la conférence. L'absence de l'un de ces paramètres dans les formes demande/indication de la primitive équivaut à utiliser les valeurs par défaut transmises par les paramètres *Heure* et *Durée* du Tableau 8-5. Le paramètre *Temps d'entrée d'un site* représente le temps qui s'écoule entre le début de la conférence (défini par le paramètre *Heure* du Tableau 8-5) et l'entrée du site dans la conférence (c'est-à-dire que la valeur indiquée par le paramètre *Temps d'entrée d'un site*, ajoutée à celle qui est indiquée par le paramètre *Durée d'un site* ne doit pas dépasser la valeur indiquée par le paramètre *Durée* du Tableau 8-5).

Le paramètre *Débit utile d'un site (RC)* permet au demandeur d'indiquer que le dispositif de conférence multimédia contenu dans ce site est destiné à se connecter au débit utile indiqué et non pas à celui qui est indiqué par le paramètre *Débit utile global*. L'absence de ce paramètre dans les formes demande/indication des primitives équivaut à utiliser le débit utile global.

Le paramètre *Méthode d'entrée d'un site (RC)* sert à indiquer si le site doit être appelé ou s'il appellera son pont MCU de rattachement au moment de son entrée. Si ce paramètre est fourni dans

les formes demande/indication de la primitive, il remplace (en priorité) la méthode par défaut indiquée dans le paramètre *Méthode par défaut d'entrée des sites*. Les valeurs possibles sont: "Site appelant" et "Site appelé".

Le paramètre *Adresse réseau attribuée* est obligatoire si la primitive de réponse/confirmation programme le site en tant que site appelant. Ce paramètre donne des renseignements sur le type de connexion (RNIS ou RDCC commuté, débit utile, etc.), sur les adresses réseau et sur le type de profil que le site appelant devra utiliser pour entrer dans la conférence. Ce paramètre est décrit au 8.2.5.2.1.

Le paramètre *Indicateur de liste d'attente* est un fanion qui indique facultativement que les ressources en ponts MCU pour le site sont mises en liste d'attente (voir le paramètre *Politique de listes d'attente* au 8.2.5).

Le paramètre *Quote-part de facturation des participants (RC)* est un nombre entier facultatif contraint dans l'intervalle [0..100] qui indique la partie du coût de conférence du site participant qui ne doit pas être facturée à l'organisateur. L'absence de ce paramètre alors que le paramètre *Mode de facturation* est présent dans la demande avec la valeur "Mode mixte" équivaut à fournir le paramètre avec la valeur 100.

NOTE – La sélection d'un mode de facturation mixte tout en fixant la quote-part de facturation des participants à 100 pour un site de réservation quelconque ne doit pas être considérée comme équivalente à la sélection du mode de facturation avec la valeur "Participants". Par exemple, le système de réservation (c'est-à-dire le fournisseur de services) peut considérer qu'un mode de facturation mixte, dont toutes les quotes-parts de facturation des participants ont la valeur 100 (explicitement ou implicitement), signifie que seul le coût de réservation sera facturé à l'organisateur, chaque participant étant facturé du coût de sa communication tout en partageant avec les autres participants le coût de réservation en cas de mode de facturation de type "Participants". En général, l'interprétation exacte de chaque mode est laissée à la discrétion des systèmes de réservation.

Le paramètre facultatif *Liste des participants à la conférence* est utilisé pour donner des informations aux participants situés dans un site de réservation (ou qui lui sont rattachés). Ce paramètre s'applique normalement aux salles de vidéoconférence ou aux ponts MCU externes. Les informations qu'il fournit peuvent être utilisées afin de constituer des items informationnels purs (tels que les noms de participant dans un profil de conférence T.120, voir la Recommandation T.124) ou, pour certains éléments comme les numéros de télécopie ou les adresses de courrier électronique, afin d'envoyer des notifications de réservation aux participants. Chaque élément de cette liste a la structure indiquée au Tableau 8-16.

8.2.5.2.1 Paramètre *Adresse réseau attribuée*

Le paramètre *Adresse réseau attribuée* qualifie la connexion que le site appelant doit établir pour entrer dans la conférence. Il a la structure indiquée dans le Tableau 8-12.

Tableau 8-12/T.135 – Paramètre *Adresse réseau attribuée*

Paramètre	Rsp	Cnf
descripteur de connexion	M	M(=)
interface réseau à utiliser	M	M(=)
profil à appliquer	M	M(=)

Le paramètre *Descripteur de connexion* a une structure qui dépend du type d'interface réseau sélectionnée par le système de réservation.

Pour les connexions numériques par RNIS ou RDCC, ce paramètre a la structure indiquée au Tableau 8-13.

Tableau 8-13/T.135 – Descripteur de connexion pour RNIS/RDCC

Paramètre	Description
circuits combinés	il s'agit d'une liste de circuits numériques, caractérisés chacun par un mode de transfert et par une adresse réseau de type E.164 étendu. Le mode de transfert est soit par canal numérique à 56 kbit/s pour les RDCC, à 64 kbit/s pour les RNIS ou RDCC, ou par canal numérique à 2×64 kbit/s, à 384 kbit/s, à 1536 kbit/s, à 1920 kbit/s ou à débit multiple $n \times 64$ kbit/s pour les RNIS. Dans ce dernier cas, une valeur d'entier comprise dans l'intervalle [1..30] indique le débit utile de transfert du canal numérique. A l'exception du canal à 56 kbit/s, ces modes correspondent aux codes indiqués dans la Recommandation Q.931 pour l'octet " <i>Débit de transfert d'information</i> " de l'élément d'information " <i>Capacité support</i> ". Pour les RNIS, l'adresse de type E.164 étendu est facultativement accompagnée d'une liste des codes binaires utilisables pour l'élément d'information <i>Compatibilité des couches supérieures (HLC, high layer compatibility)</i> lors de l'établissement de l'appel. Les adresses réseau de type E.164 étendu comme la liste des codes HLC sont décrites dans le Tableau 8-14
agrégation de canaux	il s'agit d'une liste des algorithmes d'agrégation des canaux qui peut être utilisée pour combiner les voies. Si le paramètre <i>Circuits combinés</i> n'indique qu'un seul circuit, ce paramètre est facultatif et son absence indique qu'aucune agrégation de canaux ne doit être effectuée. Sinon, ce paramètre est obligatoire et doit contenir au moins un élément. Les valeurs normales d'un élément de cette liste sont: " <i>H.221</i> ", " <i>H.244</i> " et " <i>ISO/CEI 13871</i> "

Tableau 8-14/T.135 – Adresse réseau de type E.164 étendu

Paramètre	Description
numéro international	chaîne d'une longueur maximale de 16 chiffres qui représente le numéro international complet du pont MCU qui doit être appelé par le site réservateur
sous-adresse (facultatif)	paramètre facultatif qui n'est valide que dans le cas de connexions RNIS et qui représente la sous-adresse RNIS du pont MCU qui doit être appelé par le site réservateur. Il s'agit d'une chaîne d'une longueur maximale de 40 chiffres
chaîne de numérotation complémentaire (facultatif)	paramètre facultatif qui indique la nécessité d'informations complémentaires pour atteindre l'unité de traitement de données une fois que la connexion physique a été établie avec le pont MCU. Dans le cas d'une connexion de conversation ou de données en bande vocale, par exemple, ce paramètre peut représenter des tonalités DTMF à transmettre au-dessus du canal vocal une fois celui-ci établi. En variante, le paramètre de numérotation complémentaire peut représenter un numéro de réseau privé virtuel. Il s'agit d'une chaîne de 255 caractères au plus qui peuvent être soit les chiffres de 1 à 9, le caractère "#", le caractère "*" ou ",", (la virgule). Le caractère de virgule est censé représenter un délai de 1 s que le site réservateur doit insérer avant les caractères qui suivent

Tableau 8-14/T.135 – Adresse réseau de type E.164 étendu (fin)

Paramètre	Description
information de compatibilité de couche supérieure (facultatif)	paramètre facultatif qui n'est valide que dans le cas de réseaux de type RNIS. Il indique le mode de fonctionnement de cette partie de la connexion. Cette information est requise pour les connexions établies dans certains RNIS. Les modes de fonctionnement sont un ou plusieurs des suivants: "téléphonie à 3 kHz", "téléphonie à 7 kHz", "vidéotéléphonie", "vidéoconférence", "audiographie", "audiovisuel" ou "multimédia". Si plusieurs de ces modes sont choisis, cela signifie que le site réservateur peut utiliser librement l'un des modes indiqués. Ces codes correspondent à ceux qui sont indiqués dans la Recommandation Q.931 pour l'élément d'information <i>Compatibilité de couche supérieure</i>

Pour d'autres types de connexion, le paramètre *Descripteur de connexion* ne comporte qu'une *adresse réseau générale*, qui offre le choix entre une adresse réseau de type E.164 étendu comme indiqué dans le Tableau 8-14, une adresse de transport (dont la structure est indiquée dans le Tableau 8-15), et une adresse non normalisée.

Tableau 8-15/T.135 – Adresse de transport

Paramètre	Description
adresse de point NSAP	chaîne d'octets d'une longueur maximale de 20 octets, qui représente le codage binaire préféré (conformément au A.8.3.1/X.213 <i>Livre bleu</i>) de l'adresse de point d'accès au service de couche réseau de l'unité de traitement de données à atteindre
sélecteur de transport	paramètre facultatif qui peut être utilisé pour sélectionner le point d'accès au service de couche transport dans l'unité de traitement de données à atteindre

Le paramètre *Interface réseau à utiliser* désigne l'interface réseau que le site appelant doit utiliser pour numéroter son entrée dans la conférence. C'est l'identificateur qui est fourni lorsque le site est enregistré (voir 8.2.11.2).

Le paramètre *Profil à appliquer* désigne une configuration de base d'une suite complexe de protocoles, que le site appelant doit exploiter dans la connexion décrite par le paramètre *Descripteur de connexion*. Les valeurs normales de ce paramètre sont: "parole", "téléphonie à 3 kHz", "téléphonie à 7 kHz", "bande vocale", "relais de trames", "t123-base-rtpc", "t123-base-rdcp", "t123-base-rnislb", "h310", "h320", "h321", "h322", "h323", "h324", "h324m", "v61", "v70", "profil-chargé-par-dsmcc", ou un profil non normalisé. Pour les profils multimédias (h.3xx, asvd (multiplex voix-données sur RTC analogique), dsvd (multiplex voix-données sur RTC numérique), un fanion booléen est associé au paramètre pour indiquer si les suites de protocoles T.120 doivent être mises en œuvre pour le support de données.

8.2.5.2.2 Structure d'une description de participant

Chaque description de participant énuméré dans la liste du paramètre *Liste des participants* a la structure indiquée dans le Tableau 8-16.

Tableau 8-16/T.135 – Paramètre *Liste des participants* – Structure élémentaire

Paramètre	Req	Ind	Rsp	Cnf
nom	O	O(=)	O(=)	O(=)
complément nominatif	O	O(=)	O(=)	O(=)
adresses postales	O	O(=)	O(=)	O(=)
numéros internationaux de téléphone	O	O(=)	O(=)	O(=)
numéros internationaux de télécopie	O	O(=)	O(=)	O(=)
adresses de messagerie électronique	O	O(=)	O(=)	O(=)
adresses mnémoniques X.400	O	O(=)	O(=)	O(=)
chaînes facultatives	O	O(=)	O(=)	O(=)

Le paramètre *Nom* est une chaîne de caractères facultative qui représente le nom du participant.

Le paramètre *Complément nominatif* est une chaîne de caractères facultative qui peut comprendre des informations nominatives complémentaires telles que l'entreprise du participant, sa position hiérarchique dans cette entreprise, etc.

Le paramètre *Adresses postales* est une liste facultative de chaînes de caractères représentant chacune une adresse postale possible.

Le paramètre *Numéros internationaux de téléphone* est une liste facultative de chaînes de 16 caractères numériques au plus, représentant chacune un numéro international de téléphone possible du participant.

Le paramètre *Numéros internationaux de télécopie* est une liste facultative de chaînes de 16 caractères numériques au plus, représentant chacune un numéro international de télécopie possible du participant.

Le paramètre *Adresses de messagerie électronique* est une liste facultative de chaînes de caractères, représentant chacune une adresse de messagerie électronique possible du participant.

Le paramètre *Adresses mnémoniques X.400* est une liste facultative de tableaux, chaque tableau représentant une adresse X.400 possible du participant. Chaque ligne de tableau représente un champ de l'adresse X.400 définie sous la forme mnémonique conformément aux règles spécifiées dans la série des Recommandations X.400 (voir le chapitre 9).

Le paramètre *Chaînes facultatives* est une liste de chaînes de caractères qui peut être utilisée pour donner des informations non spécifiées dans la présente Recommandation.

8.2.6 Primitive *RS-Conference-Modify*

Cette primitive définit une transaction de modification de réservation. Ses paramètres sont énumérés dans le Tableau 8-17.

Tableau 8-17/T.135 – Primitive *RS-Conference-Modify*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
référence de conférence/réservation	M	M(=)	O(=)	O(=)
mot de passe de modificateur de conférence	O	O(=)		

Tableau 8-17/T.135 – Primitive *RS-Conference-Modify* (fin)

Paramètre	Req	Ind	Rsp	Cnf
politique par défaut	O	O(=)		
nom de la conférence (RC)	O	O(=)	C(=)	C(=)
description de la conférence	O	O(=)	O(=)	O(=)
mode de mise en cascade des ponts MCU	O	O(=)	O(=)	O(=)
commande de présidence audiovisuelle	O	O(=)	O(=)	O(=)
mot de passe du président audiovisuel	O	O(=)	O(=)	O(=)
organisateur externe T.120	O	O(=)	O(=)	O(=)
mode présidé T.120	O	O(=)	O(=)	O(=)
liste des privilèges du président T.120	O	O(=)	O(=)	O(=)
liste des privilèges communs T.120	O	O(=)	O(=)	O(=)
mot de passe du président T.120	O	O(=)	O(=)	O(=)
mot de passe commun	O	O(=)	O(=)	O(=)
débit utile global (RC)	O	O(=)	O(=)	O(=)
méthode de commutation vidéo (RC)	O	O(=)	O	O(=)
format vidéo initial (RC)	O	O(=)	O	O(=)
algorithme vidéo initial (RC)	O	O(=)	O	O(=)
algorithme audio initial (RC)	O	O(=)	O	O(=)
mode de facturation (RC)	O	O(=)	O	O(=)
compte de facturation de l'organisateur	O	O(=)	O(=)	O(=)
identificateur du détenteur de la conférence	O	O(=)	O(=)	O(=)
date	O	O(=)	O(=)	O(=)
heure	O	O(=)	O(=)	O(=)
durée	O	O(=)	O(=)	O(=)
liste des sites de réservation (RC)	O	O(=)	O	O(=)
méthode par défaut de raccordement des sites	O	O(=)	O(=)	O(=)
nouveau mot de passe de modificateur de conférence.	O	O(=)	O(=)	O(=)
acquiescement explicite requis			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

Il y a lieu de noter que cette primitive peut être émise en cours de conférence (c'est-à-dire pour modifier une réservation correspondant à une conférence qui est en cours). En général, les paramètres de cette primitive sont conformes aux règles suivantes:

- un paramètre est prévu pour remplacer la valeur transmise lors de la réservation de la conférence ou lorsque celle-ci a été modifiée pour la dernière fois;
- le fait de ne pas fournir un paramètre équivaut à ne pas le modifier;

NOTE – Pour chaque type de paramètre (chaînes, nombres, etc.), la réalisation d'une entité APE selon la présente Recommandation doit fournir des valeurs spéciales ou des paramètres supplémentaires non définis dans la présente Recommandation afin d'indiquer la désactivation d'un paramètre (c'est-à-dire pour annuler un réglage antérieur du paramètre);

- les relations entre les paramètres (présence conditionnelle ou non-applicabilité selon la présence ou la valeur d'autres paramètres) sont les mêmes qu'au 8.2.5;
- les paramètres contrôlés du système de réservation, contenus dans la transaction de réservation de conférence, restent sans changement pour la transaction de modification de conférence.

Les paramètres énumérés dans le Tableau 8-17 et non décrits ci-dessous suivent ces règles.

Le paramètre *Pointeur de connexion* est le pointeur local de la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* désigne la transaction. Voir 7.2.3.

Le paramètre *Référence de conférence/réservation* désigne la réservation dans le système de réservation. C'est la référence obtenue lors de la réservation initiale au moyen d'une précédente transaction de réservation de conférence (c'est-à-dire que cette référence est indiquée par le paramètre *Référence de conférence/réservation* de la primitive *RS-Conference-Reserve-confirm*) ou par d'autres moyens hors du domaine d'application de la présente Recommandation.

Le paramètre *Mot de passe de modificateur de conférence* est le mot de passe défini au 8.2.5. Il doit être fourni s'il a été défini lors de la réservation initiale de la conférence et si les privilèges d'utilisateur connecté ne lui permettent pas de modifier cette réservation.

Le paramètre *Politique par défaut* définit le comportement du système de réservation lorsque les nouvelles caractéristiques de la conférence se traduisent par un rejet de la modification de réservation/conférence. Les valeurs possibles de ce paramètre sont les suivantes: "*Maintien de la réservation précédente*" pour demander le maintien de la réservation précédente; "*Annulation de la réservation précédente*" pour en demander l'abandon; "*Liste d'attente générale*" pour demander l'abandon de la réservation initiale et la mise de la conférence en liste d'attente; ou "*Traitement au mieux*" pour demander l'abandon de la réservation initiale et la réservation de la conférence conformément à la politique du "traitement au mieux" (voir le paramètre *Politique de listes d'attente* au 8.2.5). L'absence de ce paramètre équivaut à sélectionner la valeur "*Maintien de la réservation précédente*".

Le paramètre *Liste des sites de réservation (RC)* est semblable au paramètre homologue qui est décrit au 8.2.5 et au 8.2.5.2, avec la différence indiquée au 8.2.6.1.

Le paramètre facultatif *Nouveau mot de passe de modificateur de conférence* permet au demandeur de modifier le mot de passe du modificateur de conférence.

Le paramètre *Acquittement explicite requis* est un fanion booléen qui indique que l'extrémité d'origine de la transaction doit acquitter explicitement la transaction au moyen du service *RS-Transaction-Acknowledge*. Voir 7.2.3.

Le paramètre *Code de résultat* peut prendre les valeurs supplémentaires suivantes: "*Liste d'attente*", "*Droits absents*", "*Conférence absente*" et "*Absence de ressources disponibles*" ou "*Code de panne par défaut*".

Le paramètre *Message de résultat* est une chaîne de caractères facultative qui peut être utilisée pour compléter le code de résultat.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions hors du domaine d'application de la présente Recommandation.

8.2.6.1 Paramètre *Liste des sites de réservation*

Tableau 8-18/T.135 – Paramètre *Liste des sites de réservation* – Structure élémentaire

Paramètre	Req	Ind	Rsp	Cnf
type d'opération	M	M(=)		
identificateur du site	M	M(=)	M	M(=)
temps d'entrée d'un site (RC)	O	O(=)	O	O(=)
durée d'un site (RC)	O	O(=)	O	O(=)
débit utile d'un site (RC)	O	O(=)	O	O(=)
méthode d'entrée (RC)	O	O(=)	O	O(=)
interface réseau à utiliser (RC)	O	O(=)	O	O(=)
adresse réseau attribuée			O	O(=)
indicateur de liste d'attente			O	O(=)
quote-part de facturation des participants (RC)	O	O(=)	O	O(=)
liste des participants	O	O(=)	O	O(=)

Dans les formes demande/indication de la primitive, la liste des sites de réservation énumère les sites qui sont affectés par la transaction de modification. Le paramètre *Type d'opération* indique l'opération à effectuer à chaque site de réservation. Les valeurs possibles de ce paramètre sont: "*Suppression*", "*Adjonction*" et "*Modification*". La valeur du paramètre *Type d'opération* détermine comme suit les contraintes sur d'autres paramètres: si le type d'opération est une suppression, seul le paramètre *identificateur de site* est applicable et obligatoire; si le type d'opération est une modification ou une adjonction, d'autres paramètres sont contraints conformément au Tableau 8-18 et suivent les règles indiquées au 8.2.6.

Dans les formes réponse/confirmation de la primitive, la liste des sites de réservation énumère l'ensemble résultant de la transaction de modification; c'est-à-dire la réunion de l'ensemble des sites qui ont été réservés avant la transaction de modification avec l'ensemble des sites qui étaient présents dans les primitives *RS-Conference-Modify-request/indication* et dont le paramètre *Type d'opération* était à la valeur "*Modification*" ou "*Adjonction*". Les paramètres sont contraints conformément au Tableau 8-18 et suivent les règles indiquées au 8.2.6.

8.2.7 Primitive *RS-Conference-Cancel*

Tableau 8-19/T.135 – Primitive *RS-Conference-Cancel*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
référence de conférence/réservation	M	M(=)	O(=)	O(=)
mot de passe du modificateur de conférence	O	O(=)		
code de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

Cette primitive est utilisée pour annuler une réservation antérieure.

Le paramètre *Pointeur de connexion* est le pointeur local de la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* identifie la transaction. Voir 7.2.3.

Le paramètre *Référence de conférence/réservation* identifie la réservation à l'intérieur du système de réservation. Voir 8.2.5.

Le paramètre *Mot de passe du modificateur de conférence* est le mot de passe défini au 8.2.5.

Le paramètre *Code de résultat* peut prendre les valeurs normalisées et supplémentaires "*Conférence absente*" et "*Transaction refusée*".

Le paramètre *Message de résultat* est une chaîne de caractères facultative qui peut être utilisée pour compléter le code de résultat.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions hors du domaine d'application de la présente Recommandation.

8.2.8 Primitive *RS-Conference-List-Inquire*

Tableau 8-20/T.135 – Primitive *RS-Conference-List-Inquire*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
filtre d'identificateurs de détenteur de conférence	O	O(=)		
filtre de noms de détenteur de conférence	O	O(=)		
filtre de noms de conférence	O	O(=)		
filtre d'états de conférence	O	O(=)		
filtre de dates	O	O(=)		
liste des résumés de conférence			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)

Cette primitive permet à l'abonné d'obtenir une liste de conférences programmées ou en cours.

Le paramètre *Pointeur de connexion* est le pointeur local de la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* identifie la transaction. Voir 7.2.3.

Le paramètre facultatif *Filtre d'identificateurs de détenteur de conférence* indique une liste d'identificateurs d'abonné devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seules les conférences réservées par des abonnés identifiés par ce paramètre (ou réservées en leur nom) doivent être retournées.

Le paramètre *Filtre de noms de détenteur de conférence* indique un ensemble d'intervalles alphabétiques devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seules les conférences réservées par des abonnés dont les noms sont compris dans les intervalles indiqués (ou réservées pour le compte de ces abonnés) doivent être retournées. Ce paramètre peut prendre plusieurs formes. Le paragraphe 9 indique les possibilités prévues par le protocole.

Le paramètre facultatif *Filtre de noms de conférence* indique un ensemble d'intervalles alphabétiques devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seules les conférences dont les noms sont compris dans les intervalles indiqués doivent être retournées. Ce paramètre peut prendre plusieurs formes. Le paragraphe 9 indique les possibilités prévues par le protocole.

Le paramètre facultatif *Filtre d'états de conférence* indique un ensemble d'états de conférence devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seules les conférences dont l'état correspond à ceux qui sont indiqués doivent être retournées. Les valeurs possibles de ce paramètre sont "*Conférence réservée*", "*Conférence partiellement réservée*", "*Conférence en liste d'attente*" et "*Conférence en cours*".

Le paramètre facultatif *Filtre de dates* spécifie un ensemble de jours d'intervalle pour la recherche. Ce paramètre peut prendre plusieurs formes. Le paragraphe 9 indique les possibilités prévues par le protocole.

Le paramètre *Liste des résumés de conférence* énumère les conférences avec le résultat de la transaction de consultation. Chaque conférence est décrite par un résumé dont la structure est indiquée au Tableau 8-21 du 8.2.8.1. L'absence de ce paramètre signifie qu'aucune conférence réservée ou en cours n'est visible par l'utilisateur connecté.

Le paramètre *Code de résultat* peut prendre la valeur normalisée et supplémentaire "*Absence de conférence correspondant aux critères*".

Le paramètre *Message de résultat* est une chaîne de caractères facultative qui peut être utilisée pour compléter le code de résultat.

8.2.8.1 Paramètre *Liste des résumés de conférence*

Chaque élément du paramètre *Liste des résumés de conférence* possède la structure définie dans le Tableau 8-21. La question de savoir si le demandeur obtient la liste des résumés de la seule conférence qu'il a réservée ou une liste concernant les conférences réservées par d'autres abonnés relève du système de réservation.

Tableau 8-21/T.135 – Paramètre *Liste des résumés de conférence* – Structure élémentaire

Paramètre	Rsp	Cnf
référence de conférence/réservation	M	M(=)
nom de conférence	O	O(=)
description de conférence	O	O(=)
identificateur du détenteur de la conférence	O	O(=)
nom du détenteur de la conférence	O	O(=)
date	M	M(=)
heure	M	M(=)
durée	M	M(=)
état	M	M(=)
autorisation de modification	M	M(=)

Le paramètre facultatif *Identificateur du détenteur de la conférence* indique l'identificateur de l'abonné qui a réservé la conférence (ou au nom duquel la conférence a été réservée).

Le paramètre facultatif *Nom du détenteur de la conférence* donne le nom de l'abonné qui a réservé la conférence (ou au nom duquel la conférence a été réservée).

Le paramètre *Etat* indique l'état de la conférence. Les valeurs possibles de ce paramètre sont "*Conférence réservée*", "*Conférence partiellement réservée*", "*Conférence en liste d'attente*" et "*Conférence en cours*".

Le paramètre *Autorisation de modification* est un fanion booléen qui indique si l'utilisateur connecté a le droit de modifier ou d'annuler la conférence.

Les autres paramètres ont été décrits dans les sous-paragraphes précédents.

8.2.9 Primitive *RS-Conference-Detail-Inquire*

Cette primitive permet au demandeur d'obtenir la forme détaillée d'une conférence programmée ou engagée particulière. Le demandeur fournit le paramètre *Référence de conférence/réservation* et le système renvoie tous les paramètres décrivant une conférence réservée. Le Tableau 8-22 énumère les paramètres décrivant une conférence réservée.

Tableau 8-22/T.135 – Primitive *RS-Conference-Detail-Inquire*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
référence de conférence/réservation	M	M(=)	M(=)	M(=)
nom de conférence			M	M(=)
description de conférence			O	O(=)
identificateur de détenteur de conférence			O	O(=)
nom du détenteur de conférence			O	O(=)
mode de mise en cascade des ponts MCU			O	O(=)
commande de présidence audiovisuelle			O	O(=)
mot de passe de président audiovisuel			O	O(=)
organisateur externe T.120			O	O(=)
mode présidé T.120			O	O(=)
liste de privilèges de président T.120			O	O(=)
liste de privilèges communs T.120			O	O(=)
mot de passe de président T.120			O	O(=)
mot de passe commun			O	O(=)
débit utile global			O	O(=)
méthode de commutation vidéo			O	O(=)
format vidéo initial			O	O(=)
algorithme vidéo initial			O	O(=)
algorithme audio initial			O	O(=)
mode de facturation			O	O(=)
compte de facturation de l'organisateur			O	O(=)
date			M	M(=)

Tableau 8-22/T.135 – Primitive *RS-Conference-Detail-Inquire* (fin)

Paramètre	Req	Ind	Rsp	Cnf
heure			M	M(=)
durée			M	M(=)
liste des sites de réservation			M	M(=)
méthode par défaut de raccordement des sites			O	O(=)
mot de passe du modificateur de conférence			O	O(=)
autorisation de modification			M	M(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

Chaque élément du paramètre *Liste des sites de réservation* contient les mêmes sous-paramètres qu'au Tableau 8-18, à l'exception du paramètre *Type d'opération*. Tous ces sous-paramètres sont contraints par les colonnes "Rsp" et "Cnf" de ce tableau. Ils sont décrits dans les sous-paragraphes précédents.

Le paramètre *Autorisation de modification* est un fanion booléen qui indique si l'utilisateur connecté possède le droit de modifier ou d'annuler la conférence.

Le paramètre *Code de résultat* peut prendre la valeur normalisée supplémentaire "*Conférence absente*".

Les autres paramètres de la primitive sont décrits dans des sous-paragraphes précédents.

8.2.10 Primitive *RS-Conference-Check-Availability*

Tableau 8-23/T.135 – Primitive *RS-Conference-Check-Availability*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
filtre de dates	M	M(=)	O(=)	O(=)
liste des sites demandeurs	M	M(=)		
débit utile minimal accepté	M	M(=)		
liste des résultats			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)

Cette primitive permet au demandeur d'obtenir une liste d'intervalles temporels libres pour une conférence donnée, définis en termes de tous les paramètres associés à l'attribution des ressources.

Le paramètre *Pointeur de connexion* est le pointeur local sur la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* identifie la transaction. Voir 7.2.3.

Le paramètre facultatif *Filtre de dates* spécifie un ensemble d'intervalles journaliers pour la recherche. Ce paramètre peut prendre plusieurs formes. Voir le paragraphe 9 les possibilités prévues par le protocole.

Le paramètre *Liste des sites demandeurs* donne une liste des sites qui doivent être considérés comme participant à la conférence. Chaque site de cette liste possède la structure décrite au 8.2.10.1.

Le paramètre *Débit utile minimal accepté* indique un débit utile minimal qui doit être appliqué globalement à la conférence. Ce paramètre est défini de la même façon que le paramètre *Débit utile global* décrit au 8.2.5.

Le paramètre *Liste des résultats* donne une liste des intervalles temporels pour lesquels la conférence peut être réservée. Chaque élément de cette liste se compose d'une liste de dates ou d'intervalles datés. A chacune de ces dates ou de ces intervalles datés est associée une liste d'intervalles temporels à l'intérieur desquels la conférence peut être réservée. A chacun de ces intervalles temporels est associé un débit utile maximal qui peut être autorisé en tant que débit utile global pour la conférence. Facultativement, pour chacun des sites énumérés dans le paramètre *Liste des sites demandeurs*, la liste des interfaces avec le réseau que le site sera ou, en variante, ne sera pas, autorisé à utiliser sera associée à chacun des intervalles temporels. Ce paramètre peut prendre plusieurs formes. Voir au paragraphe 9 les possibilités offertes par le protocole.

Le paramètre *Code de résultat* peut prendre la valeur normalisée et supplémentaire "*Absence de ressources disponibles*".

Le paramètre *Message de résultat* peut être utilisé pour fournir un complément rédactionnel au code de résultat.

8.2.10.1 Paramètre *Liste des sites demandeurs*

Chaque élément du paramètre *Liste des sites demandeurs* possède la structure définie dans le Tableau 8-24.

Tableau 8-24/T.135 – Paramètre *Liste des sites demandeurs* – Structure élémentaire

Paramètre	Req	Ind
identificateur de site	M	M(=)
temps d'entrée d'un site	O	O(=)
durée d'un site	O	O(=)
débit utile de site	O	O(=)
interfaces réseau utilisables	O	O(=)

Le paramètre *Interfaces réseau utilisables* donne une liste des interfaces avec le réseau que le site sera en mesure d'exploiter pour la conférence. L'absence de ce paramètre doit être interprétée comme indiquant la liste de toutes les interfaces enregistrées pour ce site.

Pour la description des autres paramètres, voir les sous-paragraphe précédents.

8.2.11 Primitive *RS-Site-Record*

Cette primitive sert à créer des enregistrements temporaires ou permanents de sites dans la base de données du système de réservation. Elle doit être utilisée pour enregistrer un site (temporairement ou non) avant l'émission d'une primitive *RS-Conference-Reserve* afin de réserver une conférence avec ce site.

Tableau 8-25/T.135 – Primitive *RS-Site-Record*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
indicateur permanent (RC)	O	O(=)	O	O(=)
type de site	M	M(=)	O(=)	O(=)
classe de site	C	C(=)	O(=)	O(=)
identificateur du détenteur de site	O	O(=)	O(=)	O(=)
nom du site	O	O(=)	O(=)	O(=)
géographie du site	O	O(=)	O(=)	O(=)
interfaces avec le réseau	M	M(=)	O(=)	O(=)
description d'équipement	O(=)	O(=)	O(=)	O(=)
contact	O	O(=)	O(=)	O(=)
date de suppression			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
identificateur de site			C	C(=)
données d'utilisateur	O	O(=)	O	O(=)

Le paramètre *Pointeur de connexion* est le pointeur local de la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* identifie la transaction. Voir 7.2.3.

Le paramètre *Indicateur permanent* est un fanion facultatif qui permet à l'application de réservation cliente de demander l'enregistrement permanent du site à l'intérieur de la base de données du système de réservation (le type d'enregistrement par défaut est: *enregistrement temporaire*). L'acceptation ou le rejet de ce service, ainsi que l'interprétation exacte du qualificatif *permanent* sont laissées à la discrétion du système de réservation. Par exemple, un système de réservation peut accepter l'enregistrement permanent d'un site mais supprimer automatiquement l'enregistrement correspondant à l'issue d'une longue période de non-utilisation. Dans les formes réponse/confirmation de la primitive, ce paramètre permet au système de réservation d'indiquer le type d'enregistrement qui a été adopté. Le système de réservation obtient ensuite la possibilité de transformer une demande d'enregistrement permanent en un enregistrement temporaire, et de renvoyer un code de résultat indiquant une réussite partielle ainsi que, facultativement, d'indiquer une date de suppression pour l'enregistrement de site (voir ci-dessous les paramètres *Date de suppression* et *Code de résultat*).

NOTE – Le mécanisme d'enregistrement des sites implique les règles suivantes: lorsqu'un site est enregistré temporairement, le système de réservation concerné doit garantir à l'enregistrement du site une durée de vie au moins égale à celle de la connexion de réservation parente, si le site n'est pas référencé dans une réservation de conférence acceptée et émise par cette connexion; lorsqu'un site est enregistré temporairement et référencé dans une réservation de conférence acceptée, le système de réservation doit conserver l'enregistrement du site au moins jusqu'à ce que la conférence soit terminée ou annulée.

Le paramètre *Type de site* désigne le type de dispositif de conférence. Les valeurs normalisées sont: "*Pont MCU*", "*Terminal/pont MCU combinés*", "*Terminal*", "*Téléphone*" ou "*Dispositif spécial*".

Le paramètre *Classe de site* indique les types de support que le site utilise. Ce paramètre est obligatoire lorsque le paramètre *Type de site* est mis à l'une des valeurs suivantes: "Pont MCU", "Terminal/pont MCU combinés", "Terminal" ou "Dispositif spécial" et n'est pas applicable dans les autres cas. Les valeurs normalisées de ce paramètre sont: "Conférence multimédia", "Conférence audiovisuelle", "Conférence audiographique", "Conférence visiographique", "Données seulement".

Le paramètre *Identificateur du détenteur du site* est un identificateur d'abonné en variante qui permet au demandeur d'enregistrer le site au nom d'un tiers abonné. Par défaut (c'est-à-dire lorsque ce paramètre n'est pas fourni), le détenteur de l'enregistrement du site est implicitement identifié par le paramètre *Identificateur d'abonné* fourni au moment de la connexion (c'est-à-dire dans la primitive *RS-Connect*). Il est du type *identificateur général* défini au 8.2.1.

Le paramètre *Nom du site* est une chaîne de caractères facultativement utilisée pour nommer le site.

Le paramètre *Géographie du site* est une table de sous-paramètres facultatifs qui donnent des renseignements sur l'emplacement du site. La structure de ce paramètre est décrite au 8.2.11.1.

Le paramètre *Interfaces avec le réseau* donne la liste des interfaces avec le réseau disponibles dans les arrangements relatifs aux interfaces avec le réseau dont le site est équipé. La structure élémentaire de ce paramètre est décrite au 8.2.11.2.

Le paramètre *Description d'équipement* est une table d'informations sur l'équipement de conférence de ce site. Ce paramètre est décrit par le Tableau 8-28 au 8.2.11.3.

Le paramètre *Contact* est une table facultative qui donne des informations sur un utilisateur associé au site. Sa structure est semblable à celle qui est décrite au Tableau 8-16.

Le paramètre *Date de suppression* est un paramètre facultatif que le système de réservation peut renvoyer pour indiquer la date à laquelle l'enregistrement du site sera supprimé.

Le paramètre *Code de résultat* peut prendre les valeurs normalisées et supplémentaires "Mode temporaire forcé" et "Espace d'utilisateur rempli".

Le paramètre *Message de résultat* est une chaîne de caractères facultative qui peut être utilisée pour fournir un complément rédactionnel au code de résultat.

Le paramètre *Identificateur de site* est un identificateur qui permettra de faire référence au site lors d'ultérieures transactions (par exemple pour des réservations de conférence). Sa présence est obligatoire si l'enregistrement réussit et son attribution est du ressort du système de réservation. Il est du type *identificateur général* qui est défini au 8.2.1. Ce paramètre n'est pas applicable si l'enregistrement échoue.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions hors du domaine d'application de la présente Recommandation.

8.2.11.1 Paramètre *Géographie du site*

Tableau 8-26/T.135 – Paramètre *Géographie du site*

Paramètre	Req	Ind	Rsp	Cnf
pays	O	O(=)	O(=)	O(=)
région	O	O(=)	O(=)	O(=)
ville	O	O(=)	O(=)	O(=)
adresse postale	O	O(=)	O(=)	O(=)

Les paramètres *Pays*, *Région* et *Ville* sont des chaînes de caractères alphabétiques qui désignent, respectivement, le pays, la région ou l'état ..., et la ville du site.

Le paramètre *Adresse postale* est une chaîne de caractères qui peut être utilisée pour fournir l'adresse postale du site en clair.

NOTE – L'extension à l'utilisation des services d'annuaire X.500 fera l'objet d'un complément d'étude.

8.2.11.2 Paramètre *Interfaces avec le réseau*

Ce paramètre possède la structure indiquée dans le Tableau 8-27.

Tableau 8-27/T.135 – Paramètre *Interfaces avec le réseau* – Structure élémentaire

Paramètre	Req	Ind	Rsp	Cnf
référence d'interface	M	M(=)	M(=)	M(=)
description d'interface	M	M(=)	M(=)	M(=)
profils pris en charge	M	M(=)	M(=)	M(=)

Le paramètre *Référence d'interface* est fourni par l'utilisateur connecté. Il désignera le site dans les transactions ultérieures. Il est du type *identificateur général* défini au 8.2.1.

Le paramètre *Description d'interface* possède une structure qui dépend du type du réseau associé. Les types de réseau possibles sont les suivants: RNIS, RDCC, RTGC, RDCP, ATM, réseaux mobiles et LAN.

Pour les réseaux de type RNIS et RDCC, la description d'interface porte sur l'arrangement relatif aux interfaces avec le réseau, c'est-à-dire sur une liste d'une ou de plusieurs interfaces réseau pouvant être combinées ensemble (par exemple, un ensemble de trois interfaces au débit de base RNIS dont les canaux B peuvent être agrégés). Chaque élément de l'ensemble est une liste qui indique les débits utiles qui sont pris en charge par l'interface composite, ainsi que le réseau auquel l'interface est connectée. Les débits utiles possibles sont ceux qui sont énumérés pour le paramètre *Descripteur de connexion* décrit au 8.2.5.2.1. En association avec chaque débit utile, le paramètre *Adresse réseau* donne une adresse réseau de type E.164 étendu qui doit être utilisée pour appeler l'interface composite (voir 8.2.5.2.1 la description d'une adresse réseau de type E.164 étendu). En association avec la liste des interfaces élémentaires, le paramètre *Agrégation de canaux* spécifie la liste des algorithmes d'agrégation de canaux que le site peut mettre en œuvre pour combiner des canaux individuels. Les valeurs normalisées d'un élément de cette liste sont: "*H.221*", "*H.244*" et "*ISO/CEI 13871*".

Pour les réseaux de type RTGC, la description d'interface se compose d'une adresse réseau E.164 étendue qui doit être utilisée pour appeler l'interface, dans laquelle les paramètres facultatifs *Sous-adresse* et *Compatibilité de couche supérieure* ne sont pas applicables.

Pour les réseaux de type RDCP, la description d'interface se compose d'une adresse réseau générale (voir 8.2.5.2.1 la description d'une adresse réseau générale).

Pour les réseaux en mode ATM, la description d'interface se compose d'une adresse réseau générale (voir 8.2.5.2.1) et, facultativement, d'une valeur d'entier indiquant le débit utile maximale autorisé, en cellules par seconde.

Pour les réseaux mobiles, la description d'interface se compose d'une adresse réseau de type E.164 étendu qui peut être utilisée pour appeler l'interface.

Pour les réseaux locaux, la description d'interface se compose d'une adresse réseau générale (voir 8.2.5.2.1) et, facultativement, d'une valeur d'entier indiquant en kbit/s la largeur de bande maximale qui sera mise à la disposition de l'interface en cours de conférence.

Le paramètre *Profils pris en charge* énumère les configurations de base des profils plus complexes que le site est capable de mettre en œuvre à l'interface. Les valeurs normalisées sont celles qui sont énumérées pour le paramètre *Profil à appliquer* décrit au 8.2.5.2.1. Pour les profils multimédias (H.3xx, asvd, dsvd), le fanion booléen associé indique si le site est en mesure d'exploiter la suite de protocoles T.120 pour la partie données du multiplex.

8.2.11.3 Paramètre *Description d'équipement*

Tableau 8-28/T.135 – Paramètre *Description d'équipement*

Paramètre	Req	Ind	Rsp	Cnf
référence d'équipement	O	O(=)	O(=)	O(=)
référence d'IMUX	O	O(=)	O(=)	O(=)
référence de codec audio	O	O(=)	O(=)	O(=)
algorithmes audio pris en charge	C	C(=)	O(=)	O(=)
référence de codec vidéo	O	O(=)	O(=)	O(=)
algorithmes vidéo pris en charge	C	C(=)	O(=)	O(=)
formats d'image pris en charge	C	C(=)	O(=)	O(=)
référence du processeur de données	O	O(=)	O(=)	O(=)

Le paramètre facultatif *Référence d'équipement* donne des informations générales sur le terminal, sur le pont MCU, etc., du site. Ce paramètre correspond à un paramètre du protocole du type défini en ASN.1 *VendorModelVersion* qui est soit un identificateur H.221 non normalisé (voir dans le Tableau 8-1 bis du 8.2.2 le paramètre de capacité non condensable *Identificateur du système de réservation*) ou un ensemble de trois chaînes de caractères, l'une pour le vendeur, l'autre pour le modèle et la dernière pour la version du dispositif.

Les paramètres *Vendeur d'IMUX*, *Modèle d'IMUX* et *Version d'IMUX* sont des chaînes alphanumériques qui identifient le processeur d'IMUX s'il est présent. Tous ces paramètres sont facultatifs mais il convient de noter que les paramètres *Vendeur d'IMUX* et *Modèle d'IMUX* sont inutilisables l'un sans l'autre.

Le paramètre facultatif *Référence de codec audio* donne des informations sur le codec audio du site. Sa structure est la même que celle du paramètre *Référence d'équipement*.

Le paramètre *Algorithmes audio pris en charge* se compose d'une liste d'un ou de plusieurs identificateurs d'algorithme qui indique la capacité audio du site. Ce paramètre n'est pas applicable si le site est un poste téléphonique ou n'exploite pas de dispositif audio; il est obligatoire dans les autres cas. Ses valeurs normales sont: "G.711 – loi A", "G.711 – loi μ ", "G.722", "G.723", "G.728", "G.729" et "Audio MPEG".

Le paramètre facultatif *Référence de codec vidéo* donne des informations sur le codec vidéo du site. Sa structure est la même que celle du paramètre *Référence d'équipement*.

Le paramètre *Algorithmes vidéo pris en charge* se compose d'une liste d'un ou de plusieurs identificateurs d'algorithme qui indique la capacité vidéo du site. Ce paramètre n'est pas applicable si le site est un poste téléphonique ou n'exploite pas de dispositif vidéo; il est obligatoire dans les autres cas. Ses valeurs normales sont, pour chaque élément de la liste: "H.261", "H.262" ou "H.263".

Le paramètre *Formats d'image pris en charge* se compose d'une liste d'un ou de plusieurs identificateurs de format d'image qui indique la capacité vidéo du site. Ce paramètre n'est pas applicable si le site est un poste téléphonique ou n'exploite pas de dispositif vidéo; il est obligatoire dans les autres cas. Ses valeurs normales sont, pour chaque élément de la liste: "CIF", "QCIF", "SIF", "SQCIF".

Le paramètre facultatif *Référence du processeur de données* donne des informations sur le codec audio du site. Sa structure est la même que celle du paramètre *Référence d'équipement*.

8.2.12 Primitive RS-Site-Modify

Cette primitive sert à modifier un enregistrement de site dans la base de données du système de réservation. Sa structure est donnée dans le Tableau 8-29.

Tableau 8-29/T.135 – Primitive RS-Site-Modify

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
identificateur de site	M	M(=)	O(=)	O(=)
indicateur permanent (RC)	O	O(=)	O	O(=)
type de site	O	O(=)	O(=)	O(=)
classe de site	O	O(=)	O(=)	O(=)
identificateur du détenteur de site	O	O(=)	O(=)	O(=)
nom du site	O	O(=)	O(=)	O(=)
géographie du site	O	O(=)	O(=)	O(=)
interfaces avec le réseau	O	O(=)	O(=)	O(=)
description d'équipement	O	O(=)	O(=)	O(=)
contact	O	O(=)	O(=)	O(=)
date de suppression			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

En général, les paramètres de cette primitive sont conformes aux règles suivantes:

- un paramètre est inclus dans cette primitive afin de remplacer la valeur transmise lors de l'enregistrement du site ou lors de sa dernière modification;
- la non-fourniture d'un paramètre équivaut à sa non-modification.

NOTE – Pour chaque type de paramètre (chaînes, nombres, etc.) les mises en œuvres d'entités APE conformes à la présente Recommandation peuvent fournir des valeurs spéciales ou des paramètres supplémentaires, non définis dans la présente Recommandation, afin d'indiquer la désactivation d'un paramètre (c'est-à-dire pour annuler un précédent positionnement du paramètre).

Les paramètres énumérés dans le Tableau 8-29 et non décrits ci-dessous suivent ces règles.

Le paramètre *Identificateur de site* désigne l'enregistrement de site à modifier. Il est obligatoire dans les primitives de demande et d'indication mais facultatif dans celles de confirmation et de réponse. Sa valeur est celle qui est renvoyée par le système de réservation lors de l'enregistrement du site au

moyen de la primitive *RS-Record-Site* ou par d'autres moyens hors du domaine d'application de la présente Recommandation.

Le paramètre *Interfaces avec le réseau* est semblable au paramètre *Interfaces avec le réseau* du 8.2.11.2, avec les différences présentées au 8.2.12.1.

Le paramètre *Code de résultat* peut prendre les valeurs normalisées et supplémentaires suivantes: "*Mode temporaire forcé*", "*Site absent*" et "*Espace d'utilisateur plein*" ou "*Code de défaillance par défaut*".

Tous les autres paramètres et sous-paramètres décrivant le site deviennent facultatifs.

8.2.12.1 Paramètres *Interfaces avec le réseau*

Dans la primitive *RS-Site-Modify*, ce paramètre possède la structure indiquée dans le Tableau 8-30.

Tableau 8-30/T.135 – Paramètre *Interfaces avec le réseau* – Structure élémentaire

Paramètre	Req	Ind	Rsp	Cnf
type d'opération	M	M(=)		
référence d'interface	M	M(=)	M	M(=)
description d'interface	O	O(=)	M	M(=)
profils pris en charge	O	O(=)	M	M(=)

Dans les formes demande/indication de la primitive, la liste des interfaces avec le réseau contient les interfaces qui sont affectées par la transaction de modification. Le paramètre *Type d'opération* désigne l'opération à effectuer sur chaque interface. Les valeurs possibles de ce paramètre sont: "*Suppression*", "*Adjonction*" et "*Modification*". La valeur du paramètre "*Type d'opération*" détermine comme suit les contraintes applicables à d'autres paramètres. Si le paramètre *Type d'opération* indique une suppression, seul le paramètre *Référence d'interface* est applicable et obligatoire. Si le paramètre *Type d'opération* indique une modification ou une adjonction, d'autres paramètres sont contraints conformément au Tableau 8-27, conformément aux règles indiquées dans le 8.2.11.2.

Dans les formes réponse/confirmation de la primitive, la liste des interfaces avec le réseau doit contenir l'ensemble des interfaces résultant de la transaction de modification, c'est-à-dire la réunion de l'ensemble des interfaces qui étaient définies avant la transaction de modification avec l'ensemble des interfaces qui étaient présentes dans les primitives *RS-Site-Modify-request/indication* et dont le paramètre *Type d'opération* était à la valeur "*Modification*" ou "*Adjonction*". Les paramètres sont contraints conformément au Tableau 8-27 et suivent les règles indiquées au 8.2.11.2.

8.2.13 Primitive *RS-Site-Delete*

Cette primitive est utilisée pour supprimer un enregistrement de site dans la base de données du système de réservation. Ses paramètres sont énumérés dans le Tableau 8-31.

Tableau 8-31/T.135 – Primitive *RS-Site-Delete*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=)	M
identificateur de site	M	M(=)	O(=)	O(=)

Tableau 8-31/T.135 – Primitive *RS-Site-Delete* (fin)

Paramètre	Req	Ind	Rsp	Cnf
Ccode de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

Le paramètre *Code de résultat* peut prendre la valeur normalisée supplémentaire "*Site absent*".

Les autres paramètres sont décrits dans les sous-paragraphes précédents.

8.2.14 Primitive *RS-Site-Directory-Inquire*

Cette primitive est utilisée pour consulter une liste de sites enregistrés dans la base de données du système de réservation. Les sites accessibles par ce service peuvent être d'autres sites que ceux qui ont été enregistrés par l'utilisateur, comme des salles de vidéoconférence louées par le fournisseur de services, des sites enregistrés par d'autres utilisateurs, etc. La structure de cette primitive est indiquée dans le Tableau 8-32.

Tableau 8-32/T.135 – Primitive *RS-Site-Directory-Inquire*

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=IN)	M(=RQ)
pointeur de transaction		M	M(=IN)	M
filtre d'identificateurs de détenteur de site	O	O(=)		
filtre de noms de détenteur de site	O	O(=)		
filtre de noms de site	O	O(=)		
filtre d'emplacements géographiques	O	O(=)		
liste de sites de résultat			O	O(=)
code de résultat			M	M(=)
message de résultat			O	O(=)
données d'utilisateur	O	O(=)	O	O(=)

Le paramètre *Pointeur de connexion* est le pointeur local de la connexion de réservation. Voir 7.2.2.

Le paramètre *Pointeur de transaction* identifie la transaction. Voir 7.2.3.

Le paramètre facultatif *Filtre d'identificateurs de détenteur de site* indique une liste d'identificateurs d'abonné devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seuls les sites réservés par des abonnés identifiés par ce paramètre (ou réservés en leur nom) doivent être retournés.

Le paramètre *Filtre de noms de détenteur de site* indique un ensemble d'intervalles alphabétiques devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seuls les sites réservés par des abonnés dont les noms sont compris dans les intervalles indiqués (ou réservés pour le compte de ces abonnés) doivent être retournés. Ce paramètre peut prendre plusieurs formes. Le paragraphe 9 indique les possibilités prévues par le protocole.

Le paramètre facultatif *Filtre de noms de site* indique un ensemble d'intervalles alphabétiques devant servir de critères de sélection pour la consultation. Si ce paramètre est fourni, seuls les sites dont les

noms sont compris dans les intervalles indiqués doivent être retournés. Ce paramètre peut prendre plusieurs formes. Le paragraphe 9 indique les possibilités prévues par le protocole.

Le paramètre facultatif *Filtre d'emplacements géographiques* indique un ensemble de trois intervalles alphabétiques devant servir de critères de sélection par pays, par région et par ville. Ce paramètre peut prendre plusieurs formes. Le paragraphe 9 indique les possibilités prévues par le protocole.

Le paramètre *Liste de sites de résultat* est la liste des sites fournissant le résultat de la transaction de consultation. Ce paramètre est décrit au 8.2.14.1. L'absence de ce paramètre dans les primitives de réponse/confirmation, tandis que le paramètre *Code de résultat* indique un succès de la transaction, doit être interprétée comme une absence de visibilité de sites par l'utilisateur connecté.

Le paramètre *Code de résultat* peut prendre la valeur normalisée supplémentaire "*Absence de critères de correspondance*".

Le paramètre *Message de résultat* est une chaîne de caractères facultative qui peut être utilisée pour fournir un complément rédactionnel au paramètre *Code de résultat*.

Le paramètre *Données d'utilisateur* contient des données facultatives qui peuvent être utilisées pour des fonctions hors du domaine d'application de la présente Recommandation.

8.2.14.1 Paramètre *Liste de sites de résultat*

Ce paramètre est une liste de descriptions de sites renvoyée par le système de réservation. Chaque élément de la liste possède la structure définie par le Tableau 8-33.

Tableau 8-33/T.135 – Paramètre *Liste de sites de résultat* – Structure élémentaire

Paramètre	Rsp	Cnf
identificateur de site	M	M(=)
type de site	M	M(=)
classe de site	C	C(=)
identificateur de détenteur de site	O	O(=)
nom de détenteur de site	O	O(=)
nom de site	M	M(=)
géographie du site	O	O(=)
interfaces avec le réseau	M	M(=)
description d'équipement	O	O(=)
informations supplémentaires sur le site	O	O(=)
date de suppression	O	O(=)
autorisation de modification	M	M(=)

Le paramètre *Identificateur de site* est l'identificateur du site dans la base de données du système de réservation.

Le paramètre *Type de site* indique le type du site. Voir 8.2.11.

Le paramètre *Classe de site* indique la classe du site. Voir 8.2.11.

Le paramètre *Identificateur de détenteur de site* identifie l'abonné qui a créé l'enregistrement du site (ou au nom duquel l'enregistrement du site a été créé).

Le paramètre *Nom de détenteur de site* indique le nom de l'abonné qui a créé l'enregistrement du site (ou au nom duquel l'enregistrement du site a été créé).

Le paramètre *Nom de site* est une chaîne de caractères nommant le site.

Le paramètre facultatif *Géographie du site* donne des informations sur l'emplacement géographique du site. Ce paramètre possède la structure définie par le Tableau 8-26 du 8.2.11.1.

Le paramètre *Interfaces avec le réseau* énumère toutes les interfaces réseau du site. Voir 8.2.11.2.

Le paramètre *Description d'équipement* donne des informations sur l'équipement de ce site comme défini au 8.2.11.

Le paramètre *Informations supplémentaires sur le site* peut être utilisé pour décrire d'autres caractéristiques du site. Ce paramètre est une chaîne de caractères de longueur indéterminée. Il peut par exemple servir à indiquer la capacité d'accueil d'une salle de vidéoconférence louée, la présence d'un tableau blanc électronique, etc.

Le paramètre facultatif *Date de suppression* indique à quelle date l'enregistrement du site sera supprimé de la base de données de réservation (voir 8.2.11).

Le paramètre *Autorisation de modification* est un fanion booléen indiquant si l'utilisateur connecté possède le droit de modifier ou de supprimer l'enregistrement du site.

8.2.15 Primitive RS-Non-Standard-Request

Cette primitive permet au demandeur d'effectuer une transaction ou sous-transaction non normalisée. Comme décrit au 7.2.3, une transaction ou sous-transaction non normalisée peut être, selon son type, non acquittée, acquittée ou explicitement acquittée. Les paramètres de cette primitive sont énumérés dans le Tableau 8-34.

Tableau 8-34/T.135 – Primitive RS-Non-Standard-Transaction

Paramètre	Req	Ind	Rsp	Cnf
pointeur de connexion	M	M	M(=)	M(=RQ)
pointeur de transaction		M	M(=)	M
pointeur de transaction parente	O	O	O(=)	O(=RQ)
type d'acquiescement			O	
acquiescement explicite requis				O
clé de protocole	O	O(=)	O	O(=)
données	M	M(=)	M	M(=)

Le paramètre *Pointeur de connexion* identifie la connexion de réservation. Voir 7.2.2.1.

Le paramètre *Pointeur de transaction* identifie la transaction non normalisée.

Le paramètre facultatif *Pointeur de transaction parente* désigne une transaction parente. Voir 7.2.3.4.

Le paramètre *Type d'acquiescement* peut prendre une des valeurs suivantes: "*Acquiescement explicite*", "*Acquiescement*" ou "*Pas d'acquiescement*", selon le type de transaction ou de sous-transaction pour lequel le service est utilisé. Ce paramètre n'est applicable que dans la forme réponse de la primitive et son absence équivaut à la valeur "*Pas d'acquiescement*" (voir 7.2.3.4).

Le paramètre *Acquittement explicite requis* indique que l'application de réservation doit acquitter explicitement la transaction au moyen du service *RS-Transaction-Acknowledge-request*, comme indiqué au 7.2.3.4. Ce paramètre n'est applicable que pour la forme confirmation de la primitive.

Le paramètre facultatif *Clé de protocole* indique le protocole ou le format de données utilisé pour les données non normalisées. Il s'agit soit d'un identificateur de type ASN.1 OBJECT IDENTIFIER faisant partie d'un protocole normalisé ou non normalisé par une Recommandation, soit d'un identificateur non normalisé utilisant les conventions de codage de la Recommandation H.221.

Le paramètre *Données* contient les informations émises ou reçues.

8.2.16 Primitive *RS-Non-Standard-Data*

Cette primitive permet au demandeur d'envoyer à l'extrémité homologue des données non normalisées. Comme décrit au 7.2.3, les données envoyées peuvent être corrélées avec une transaction ou sous-transaction particulière. Les paramètres de cette primitive sont énumérés dans le Tableau 8-35.

Tableau 8-35/T.135 – Primitive *RS-Non-Standard-Data*

Paramètre	Req	Ind
pointeur de connexion	M	M
pointeur de transaction associée	O	O
clé de protocole	O	O(=)
données	M	M(=)

Le paramètre *Pointeur de connexion* identifie la connexion de réservation. Voir 7.2.2.1.

Le paramètre facultatif *Pointeur de transaction associée* identifie une (sous-)transaction à laquelle les données envoyées sont associées.

Le paramètre facultatif *Clé de protocole* indique le protocole ou le format de données utilisé pour les données non normalisées (voir 8.2.15).

Le paramètre *Data* contient les informations émises ou reçues.

8.2.17 Primitive *RS-Transaction-Allocation*

Cette primitive n'existe que sous la forme indication. Elle est utilisée par une entité URST APE pour indiquer à l'application de réservation l'identificateur attribué pour la transaction engagée. L'entité URST APE doit propager cette primitive dès que l'unité PDU *rsXXXRequest* ou *rsNonStandardRequest* est envoyée à l'entité homologue. Afin d'éviter des divergences, lorsqu'une application de réservation lance plusieurs transactions simultanément (c'est-à-dire qu'elle n'attend pas la primitive *RS-Transaction-Allocation-indication* correspondante avant de lancer la transaction suivante), une entité URST APE doit remettre en séquence les primitives *RS-Transaction-Allocation-indication* dans le même ordre que les demandes de lancement de transaction reçues. Il est toutefois recommandé que, si possible, une application de réservation qui a lancé une transaction attende la primitive *RS-Transaction-Allocation-indication* avant de lancer une autre transaction.

Tableau 8-36/T.135 – Primitive *RS-Transaction-Allocation*

Paramètre	Ind
pointeur de connexion	M
pointeur de transaction	M
pointeur de transaction parente	O

Le paramètre *Pointeur de connexion* identifie la connexion de réservation.

Le paramètre *Pointeur de transaction* identifie la transaction.

Le paramètre *Pointeur de transaction parente* identifie la transaction parente.

8.2.18 Primitive *RS-Transaction-Acknowledge*

La forme demande de cette primitive est utilisée par une application de réservation située du côté de l'origine d'une transaction afin d'acquitter explicitement la transaction, lorsque la réponse est reçue du côté répondeur avec le paramètre *Acquittement explicite requis* activé dans la primitive de service correspondante. Dès réception d'une primitive *RS-Transaction-Acknowledge-request* en provenance de l'application de réservation, l'entité URST APE doit envoyer à son homologue une unité *rsTransactionAcknowledgeIndication* et doit propager en amont vers l'application une primitive *RS-Transaction-Acknowledge-confirm*. Du côté homologue, l'entité URST APE propage une forme indication de la primitive pour indiquer que la transaction a été explicitement acquittée.

Tableau 8-37/T.135 – Primitive *RS-Transaction-Acknowledge*

Paramètre	Req	Ind
pointeur de connexion	M	M
pointeur de transaction	M	M(=)

Le paramètre *Pointeur de connexion* identifie la connexion de réservation.

Le paramètre *Pointeur de transaction* identifie la transaction acquittée.

8.2.19 Primitive *RS-Transaction-Cancel*

La forme demande de cette primitive est utilisée par une application de réservation pour annuler une transaction. Elle ne doit être utilisée que par le côté d'origine de la transaction. Dès réception d'une demande *RS-Transaction-Cancel-request* en provenance de l'application de réservation, l'entité URST APE vérifie le numéro de transaction correspondant afin de déterminer si l'application est autorisée à annuler la transaction; elle doit ensuite propager immédiatement en amont une primitive *RS-Transaction-Cancel-confirm* avec le paramètre *Code de résultat* mis à la valeur "Non émetteur" si ce n'est pas le cas. Du côté homologue, l'entité URST APE propage une forme indication de la primitive pour signaler l'annulation.

Tableau 8-38/T.135 – Primitive *RS-Transaction-Cancel*

Paramètre	Req	Ind
pointeur de connexion	M	M
pointeur de transaction	M	M(=)

Le paramètre *Pointeur de connexion* identifie la connexion de réservation.

Le paramètre *Pointeur de transaction* identifie la transaction acquittée.

8.2.20 Primitive *RS-Transaction-Error*

Cette primitive sert à indiquer à l'application de réservation homologue qu'une erreur a été détectée dans un précédent message entrant, associé à une transaction. Dès réception d'une demande *RS-Transaction-Error-request* en provenance de l'application de réservation, l'entité URST APE doit envoyer une unité PDU *rsTransactionErrorIndication* à son homologue, en mettant le paramètre *errorCode* de cette unité PDU à la valeur acheminée dans le paramètre *Code d'erreur* de la primitive de demande. Dès réception d'une unité PDU *rsTransactionErrorIndication*, une entité URST APE doit propager une forme indication de la primitive vers l'application de réservation locale.

Tableau 8-39/T.135 – Primitive *RS-Transaction-Error*

Paramètre	Req	Ind
pointeur de transaction	M	M(=)
code d'erreur	M	M(=)

Le paramètre *Pointeur de connexion* identifie la connexion de réservation.

Le paramètre *Pointeur de transaction* identifie la transaction acquittée.

Le paramètre *Code d'erreur* indique le code de l'erreur détectée. Ses valeurs normales sont: "*Protocole non normalisé et non reconnu*" et "*Code d'erreur par défaut*".

9 Spécification du protocole

9.1 Codage des unités PDU des transactions URST

La structure des unités PDU des transactions URST est spécifiée au 9.2 au moyen de la notation ASN.1 selon la Recommandation X.680. Toutes les unités PDU des transactions URST doivent être codées avant d'être transmises par application des règles de codage condensé de la Recommandation X.691, avec utilisation de la variante d'alignement de base.

NOTE – L'utilisation des étiquettes automatiques dans la définition du protocole URST implique que l'ordre des structures SEQUENCE et CHOICE, contenues dans le cadre de cette définition, se répercute sur les valeurs réellement codées.

9.2 Module ASN.1 des transactions URST

URST-PROTOCOL DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

-- Exporte tous les symboles

-- Importe les définitions de T.124

IMPORTS

**TextString, SimpleTextString, DialingString, ExtraDialingString, SubAddressString,
H221NonStandardIdentifier, Key, NonStandardParameter, ChallengeResponse, ChallengeRequest,
Privilege**

FROM GCC-PROTOCOL {itu-t(0) recommendation(0) t(20) 124};

-- Définition des types

UnsignedInteger ::= INTEGER (0..MAX)

LatinCapitalLettersString ::= BMPString (FROM(latinCapitalA..latinCapitalZ))

LatinSmallLettersString ::= BMPString (FROM(latinSmallA..latinSmallZ))

LatinLettersString ::= BMPString (FROM(LatinCapitalLettersString | LatinSmallLettersString))

DigitalString ::= BMPString (FROM (digit0..digit9))

LatinAlphanumericString ::= BMPString (FROM (LatinLettersString | DigitalString))

GenericID ::= CHOICE {
 numberID UnsignedInteger,
 stringID LatinAlphanumericString
}

ConnectionID ::= SEQUENCE {
 owningFlag BOOLEAN,
 connectionNumber INTEGER (0..65535)
}

TransactionID ::= SEQUENCE {
 owningFlag BOOLEAN,
 transactionNumber INTEGER (0..65535)
}

SimpleTextStringInterval ::= CHOICE {
 thisString SimpleTextString,
 anyFromThisStringSimpleTextString,
 anyToThisString SimpleTextString,
 inBetween SEQUENCE {
 inferiorString SimpleTextString,
 superiorString SimpleTextString
 }
}

Time ::= SEQUENCE {
 hour INTEGER (0..23),
 minute INTEGER (0..59)
}

TimeInterval ::= CHOICE {
 whenEver NULL,
 never NULL,
 fromThisTimeUntilWhenever Time,
 fromNowUntilThisTime Time,
 inBetween SEQUENCE {
 startTime Time,
 endTime Time
 }
}

Day ::= SEQUENCE {
 day INTEGER (0..31),

```

    month      INTEGER (1..12),
    year       UnsignedInteger OPTIONAL -- valeur par défaut (implicite = l'année actuelle)
}

```

```

DaysInterval ::= CHOICE {
    whenEver      NULL,
    never          NULL,
    todayOnly     NULL,
    thisDayOnly   Day,
    fromThisDayUntilWhenever Day,
    fromNowUntilThisDay Day,
    inBetween     SEQUENCE {
        firstDay   Day,
        lastDay    Day
    }
}

```

```

Duration ::= SEQUENCE {
    hours   UnsignedInteger,
    minutes INTEGER (0..59)
}

```

```

ConferenceDuration ::= CHOICE {
    permanent      NULL,
    untilTermination NULL,
    duration        Duration
}

```

```

SiteParticipationDuration ::= CHOICE {
    allConference  NULL,
    untilExit      NULL,
    duration        Duration
}

```

```

AuthenticationData ::= SEQUENCE {
    suppliedElement CHOICE {
        passwordInTheClear LatinAlphanumericString, -- mot de passe fourni directement
        challengeResponse    ChallengeResponse
    } OPTIONAL,
    requiredElement ChallengeRequest OPTIONAL,
    ...
}

```

```

AcknowledgmentType ::= CHOICE {
    noAcknowledgment NULL,
    simple            NULL,
    explicit          NULL,
    ...
}

```

```

MCUCascadingMode ::= CHOICE {
    simple          NULL,
    masterExternal NULL,
    masterInternal NULL,
    unspecified     NULL,
    ...
}

```

```

VideoSwitchingMethod ::= CHOICE {
    voiceActivityDetection NULL,
    periodicRotation       UnsignedInteger, -- période en secondes
}

```

```

    fixedBroadcaster      GenericID,                -- Identificateur obtenu à
                                                                -- l'enregistrement du site
    lastSpeakersMosaic    UnsignedInteger,
    fixedListMosaic       SET (SIZE(1..MAX)) OF GenericID, -- nombre de sources
                                                                -- Identificateurs d'enregistrement de
                                                                -- sites
    ...
}

VideoFormat ::= CHOICE {
    cif          NULL,
    qcif         NULL,
    sif          NULL,
    sqcif        NULL,
    runtimeChoice NULL,
    ...
}

VideoCoding ::= CHOICE {
    h261        NULL,
    h262        NULL,
    h263        NULL,
    runtimeChoice NULL,
    ...
}

AudioCoding ::= CHOICE {
    g711-A      NULL,
    g711-mu     NULL,
    g722        NULL,
    g723        NULL,
    g728        NULL,
    g729        NULL,
    mpegAudio   NULL,
    runtimeChoice NULL,
    ...
}

WaitingListPolicy ::= CHOICE {
    allWaitingList  NULL,
    bestEffort      NULL,
    noWaitingList   NULL,
    ...
}

BillingMode ::= CHOICE {
    organizer      NULL,
    participants   NULL,
    mixed          NULL,
    implicitBillingMode NULL,
    ...
}

X400Address ::= SEQUENCE {
    administrationDomain SimpleTextString,           -- champ A
    country               SimpleTextString,           -- champ C
    privateDomain         SimpleTextString OPTIONAL,  -- champ P
    organization          SimpleTextString OPTIONAL,  -- champ O
    organizationalUnits   SET (SIZE(1..4)) OF SimpleTextString OPTIONAL, -- champ OUX
    givenName            SimpleTextString OPTIONAL,   -- champ G
    initials             SimpleTextString OPTIONAL,   -- champ I
    surname              SimpleTextString OPTIONAL,   -- champ S
}

```

```

    generationQualifier    SimpleTextString OPTIONAL,
    commonName             SimpleTextString OPTIONAL
}
-- champ Q
-- champ C

ElectronicAddress ::= CHOICE {
    x400          X400Address,
    e-mail SimpleTextString,
    ...
}

PersonDescription ::= SEQUENCE {
    name           SimpleTextString,
    nameComplement SimpleTextString OPTIONAL,
    postalAddress  SimpleTextString OPTIONAL,
    telephoneNumbers SET (SIZE(1..MAX)) OF DialingString OPTIONAL,
    facsimileNumbers SET (SIZE(1..MAX)) OF DialingString OPTIONAL,
    electronicAddresses SET (SIZE(1..MAX)) OF ElectronicAddress OPTIONAL,
    optionalStrings  SET (SIZE(1..MAX)) OF SimpleTextString OPTIONAL,
    ...
}

JoiningMethod ::= CHOICE {
    called      NULL,
    calling     NULL,
    ...
}

ExtendedE164NetworkAddress ::= SEQUENCE {
    internationalNumber DialingString,
    subAddress           SubAddressString OPTIONAL,
    extraDialing         ExtraDialingString OPTIONAL
}

TransportAddress ::= SEQUENCE {
    nsapAddress      OCTET STRING (SIZE (1..20)),
    transportSelector OCTET STRING OPTIONAL
}

NetworkAddress ::= CHOICE {
    extendedE164      ExtendedE164NetworkAddress,
    transportAddress TransportAddress,
    nonStandard       NonStandardParameter,
    ...
}

ChannelAggregationMethod ::= CHOICE {
    h221          NULL,
    h244          NULL,
    iso-iec-13871 NULL,
    nonStandard   NonStandardParameter,
    ...
}
-- Le mode d'agrégation réel est sélectionné dynamiquement
-- selon les procédures décrites dans l'ISO/CEI 13871.

Profile ::= CHOICE {
    simpleProfile CHOICE {
        -- Mode de transfert de base :
        speech           NULL,
        telephony-3kHz   NULL,
        telephony-7kHz   NULL,
        ...
    }
    ...
}
-- Téléphonie simple
-- G.711 sur RNIS ou RDCC
-- G.722 sur RNIS ou RDCC

```

```

        voice-band          NULL,      -- Modems
        frameRelay,        NULL,
-- Profils de données T.120 seulement (T.123):
        t123-pstn-basic    NULL,
        t123-psdn-basic    NULL,
        t123-b-isdn-basic  NULL
    },
    multimediaProfile      SEQUENCE {
        profile            CHOICE {
            h310           NULL,
            h320           NULL,
            h321           NULL,
            h322           NULL,
            h323           NULL,
            h324           NULL,
            h324m          NULL,
            v61            NULL,      -- ASVD
            v70            NULL      -- DSVD
        },
        t120Indicator      BOOLEAN -- Ce fanion indique si les protocoles T.120
                                -- sont mis en œuvre pour les données
    },
    dsmccDownloadProfile  NULL,
    nonStandard           NonStandardParameter,
    ...
}

AllocatedNetworkAddress ::= SEQUENCE {
    connectionDescriptor CHOICE {
        combinedCircuits SEQUENCE {
            circuitsList SET (SIZE(1..MAX)) OF SEQUENCE {
                circuitMode CHOICE {
                    -- circuit nominaux pour RNIS et RDCC
                    digital-56k    NULL,
                    digital-64k    NULL,
                    digital-2x64k  NULL,
                    digital-384k   NULL,
                    digital-1472k  NULL,
                    digital-1536k  NULL,
                    digital-1920k  NULL,
                    multirate-base-64k INTEGER (1..30) -- nombre de canaux B
                },
                networkAddress SEQUENCE {
                    extendedE164 ExtendedE164NetworkAddress,
                    highLayerCompatibility HLCSupportedCodePoints OPTIONAL
                },
                channelAggregations SET OF ChannelAggregationMethod OPTIONAL, -- indique
                    -- les méthodes d'agrégation qui peuvent être utilisées pour combiner les circuits
                    -- L'absence de ce paramètre indique qu'aucune agrégation de canaux ne doit être
                    -- effectuée.
            },
            singleAddress SEQUENCE {
                address NetworkAddress,
                highLayerCompatibility HLCSupportedCodePoints OPTIONAL
            }
        },
        networkInterfaceToUse GenericID OPTIONAL,
        profileToOperate Profile OPTIONAL,
        ...
}

```

```

BookingSite ::= SEQUENCE {
    siteID                GenericID,
    entryDelay            Duration DEFAULT { hours 0, minutes 0 },
    duration              SiteParticipationDuration DEFAULT allConference : NULL,
    transferRate          INTEGER (1..30) OPTIONAL,      -- N x canaux atomiques
    joiningMethod         JoiningMethod OPTIONAL,
    waitingListIndicator  BOOLEAN DEFAULT FALSE,
    allocatedNetworkAddress AllocatedNetworkAddress OPTIONAL,
    participantBillingRatio INTEGER (0..100) OPTIONAL,
    conferees             SET (SIZE(1..MAX)) OF PersonDescription OPTIONAL,
    ...
}

```

```

ApplyingSite ::= SEQUENCE {
    siteID                GenericID,
    entryDelay            Duration DEFAULT { hours 0, minutes 0 },
    duration              SiteParticipationDuration DEFAULT allConference : NULL,
    transferRate          INTEGER (1..30) OPTIONAL,      -- N x canaux atomiques
    possibleNetworkInterfaces SET (SIZE(1..MAX)) OF GenericID OPTIONAL,
    ...
}

```

```

ConferenceModificationPolicy ::= CHOICE {
    keepPrevious          NULL,
    cancelPrevious        NULL,
    allWaitingList        NULL,
    bestEffort            NULL,
    ...
}

```

```

ConferenceStatus ::= CHOICE {
    reserved              NULL,
    partiallyReserved     NULL,
    inWaitingList         NULL,
    ongoing               NULL,
    ...
}

```

```

ConferenceSummary ::= SEQUENCE {
    conferenceReference   GenericID,
    conferenceName        SimpleTextString,
    conferenceDescription SimpleTextString DEFAULT "",
    conferenceOwnerID     GenericID OPTIONAL,
    conferenceOwnerName   SimpleTextString OPTIONAL,
    date                  Day,
    time                  Time,
    duration              ConferenceDuration,
    status                ConferenceStatus,
    modifyRight           BOOLEAN DEFAULT TRUE,
    ...
}

```

```

ISDNInterface ::= SEQUENCE {
    combinedInterfaces SET (SIZE(1..MAX)) OF SEQUENCE {
        transferRatesSupported SET (SIZE(1..MAX)) OF CHOICE {
            digital-64k        NULL,
            digital-2x64k      NULL,
            digital-384k       NULL,
            digital-1536k      NULL,

```

```

        digital-1920k    NULL,
        multirate-base-64k CHOICE {
            specifiedMultiples SET OF INTEGER (1..30),
            allMultiples      NULL
        },
    },
    networkAddress      SEQUENCE {
        extendedE164     ExtendedE164NetworkAddress,
        highLayerCompatibility HLCSupportedCodePoints OPTIONAL
    },
    ...
},
channelAggregations   SET (SIZE(1..MAX)) OF ChannelAggregationMethod OPTIONAL,
...
}
CSDNInterface ::= SEQUENCE {
    combinedInterfaces SET (SIZE(1..MAX)) OF SEQUENCE {
        baseRate        CHOICE {
            digital-56k    NULL,
            digital-64k    NULL,
            ...
        },
        networkAddress   ExtendedE164NetworkAddress,
        ...
    },
    channelAggregations SET (SIZE(1..MAX)) OF ChannelAggregationMethod OPTIONAL,
    ...
}

ATMInterface ::= SEQUENCE {
    networkAddress      NetworkAddress,
    -- L'utilisation de plusieurs adresses pour l'établissement de communications multidestinataires
    -- fera l'objet d'un complément d'étude.
    maxTransferRate     INTEGER (0..MAX) OPTIONAL,          -- débit exprimé en cellules par
                                                            -- secondes
    ...
}

GSTNInterface ::= SEQUENCE {
    networkAddress      ExtendedE164NetworkAddress,
    ...
}

MobileInterface ::= SEQUENCE {
    networkAddress      ExtendedE164NetworkAddress,
    ...
}

PSDNInterface ::= SEQUENCE {
    networkAddress      NetworkAddress,
    ...
}

LANInterface ::= SEQUENCE {
    networkAddress      NetworkAddress,
    maxTransferRate     INTEGER (0..MAX) OPTIONAL,          -- débit exprimé en kbit/s
    ...
}

```

```

NetworkInterface ::= SEQUENCE {
    interfaceID          GenericID,
    interfaceDescription CHOICE {
        isdnInterface      ISDNInterface,
        csdnInterface      CSDNInterface,
        atmInterface       ATMInterface,
        gsnInterface       GSTNInterface,
        mobileInterface    MobileInterface,
        psdnInterface      PSDNInterface,
        lanInterface       LANInterface,
        nonStandard        NonStandardParameter,
        ...
    },
    profilesSupported    Profiles,
    ...
}

```

```

SiteType ::= CHOICE {
    mcu                NULL,
    combinedTerminalMCU NULL,
    terminal            NULL,
    telephone          NULL,
    specialDevice       NULL,
    nonStandard         NonStandardParameter,
    ...
}

```

```

SiteClass ::= CHOICE {
    multimedia        NULL,
    audiovisual        NULL,
    audiographic       NULL,
    visiographic       NULL,
    dataOnly           NULL,
    nonStandard        NonStandardParameter,
    ...
}

```

```

VendorModelVersion ::= CHOICE {
    h221NonStandard H221NonStandardIdentifier,
    textForm        SEQUENCE {
        vendor    SimpleTextString,
        model     SimpleTextString,
        version   SimpleTextString
    }
}

```

```

EquipmentDescription ::= SEQUENCE {
    generalInformation    VendorModelVersion OPTIONAL,
    imuxInformation       VendorModelVersion OPTIONAL,
    audioCodec            VendorModelVersion OPTIONAL,
    audioAlgorithms       SET (SIZE(1..MAX)) OF AudioCoding OPTIONAL,
    videoCodec            VendorModelVersion OPTIONAL,
    videoAlgorithms       SET (SIZE(1..MAX)) OF VideoCoding OPTIONAL,
    t120StackInformation  VendorModelVersion OPTIONAL,
    ...
}

```

```

UserData ::= CHOICE {
    rawData          OCTET STRING,
    identifiedData   SET (SIZE(1..MAX)) OF NonStandardParameter
}

```

```

ResultCode ::= CHOICE {
    success          NULL,
    badParameters   NULL,
    badValues       NULL,
    transactionRefused NULL,
    defaultFailureCode NULL,
    authenticationInProgress NULL,
    authenticationFailed NULL,
    waitingList     NULL,
    noResourcesAvailable NULL,
    noSuchConference NULL,
    temporaryModeForced NULL,
    usersSpaceFull  NULL,
    noSuchSite      NULL,
    noneMatchingCriteria NULL,
    nonStandard     NonStandardParameter,
    ...
}

```

```

-----
-- Définition des constantes
-----

```

```

latinCapitalA      BMPString ::= {0, 0, 0, 65}
latinCapitalZ      BMPString ::= {0, 0, 0, 90}
latinSmallA        BMPString ::= {0, 0, 0, 97}
latinSmallZ        BMPString ::= {0, 0, 0, 122}
digit0             BMPString ::= {0, 0, 0, 48}
digit9             BMPString ::= {0, 0, 0, 57}
implicitAccount    GenericID ::= stringID : ""
connectedSubscriber GenericID ::= stringID : ""
noPassword         LatinAlphanumericString ::= ""

```

```

-----
-- Définition des unités PDU
-----

```

```

RSConnectRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    subscriberID      GenericID OPTIONAL, -- permet l'accès anonyme
    authenticationData AuthenticationData OPTIONAL,
    higherProtocolsSupported SET (SIZE(1..MAX)) OF Key OPTIONAL,
    userData          UserData OPTIONAL,
    ...
}

```

```

RSConnectResponse ::= SEQUENCE {
    connectionID      ConnectionID,
    subscriberID      GenericID OPTIONAL,
    authenticationData AuthenticationData OPTIONAL,
    higherProtocolsSupported SET (SIZE(1..MAX)) OF Key OPTIONAL,
}

```

```

    userData          UserData OPTIONAL,
    resultCode        ResultCode,
    resultMessage     SimpleTextString OPTIONAL,
    ...
}

```

```

RSDisconnectIndication ::= SEQUENCE {
    connectionID      ConnectionID,
    reasonCode        CHOICE {
        normalTermination          NULL,
        disconnectingFromConference NULL,
        defaultDisconnectionReason NULL,
        nonStandard                 NonStandardParameter
        ...
    } OPTIONAL,
    reasonMessage     SimpleTextString OPTIONAL,
    userData          UserData OPTIONAL,
    ...
}

```

```

RSConferenceReserveRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    conferenceName    SimpleTextString,
    conferenceDescription TextString DEFAULT "", -- aucune description
    mcuCascadingMode  MCUCascadingMode DEFAULT unspecified,
    audiovisualChairControl BOOLEAN DEFAULT FALSE,
    audiovisualChairmanPassword LatinAlphanumericString DEFAULT noPassword,
    t120ExternalConvener BOOLEAN DEFAULT FALSE,
    t120ConductedMode BOOLEAN DEFAULT FALSE,
    t120ConductorPrivileges SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
    t120CommonPrivileges SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
    t120ConductorPassword LatinAlphanumericString DEFAULT noPassword,
    commonPassword    LatinAlphanumericString DEFAULT noPassword,
    globalTransferRate INTEGER (1..30), -- N x canaux atomiques
    videoSwitchingMethod VideoSwitchingMethod DEFAULT voiceActivityDetection:NULL,
    initialVideoFormat VideoFormat DEFAULT runtimeChoice,
    initialVideoAlgorithm VideoCoding DEFAULT runtimeChoice,
    initialAudioAlgorithm AudioCoding DEFAULT runtimeChoice,
    waitingListPolicy  WaitingListPolicy DEFAULT noWaitingList,
    billingMode        BillingMode DEFAULT implicitBillingMode,
    organizerBillingAccount GenericID DEFAULT implicitAccount,
    conferenceOwnerID  GenericID DEFAULT connectedSubscriber,
    date               Day,
    time               Time,
    duration            ConferenceDuration DEFAULT untilTermination : NULL,
    bookingSites        SET (SIZE(2..MAX)) OF BookingSite,
    sitesDefaultJoiningMethod JoiningMethod,
    conferenceModifierPassword LatinAlphanumericString DEFAULT noPassword,
    userData           UserData OPTIONAL,
    ...
}

```

```

RSConferenceReserveResponse ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    conferenceName    SimpleTextString OPTIONAL,
    conferenceDescription TextString OPTIONAL,

```

mcuCascadingMode	MCUCascadingMode OPTIONAL,
audiovisualChairControl	BOOLEAN OPTIONAL,
audiovisualChairmanPassword	LatinAlphanumericString OPTIONAL,
t120ExternalConvener	BOOLEAN OPTIONAL,
t120ConductedMode	BOOLEAN OPTIONAL,
t120ConductorPrivileges	SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
t120CommonPrivileges	SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
t120ConductorPassword	LatinAlphanumericString OPTIONAL,
commonPassword	LatinAlphanumericString OPTIONAL,
globalTransferRate	INTEGER (1..30) OPTIONAL, -- $N \times \text{canaux atomiques}$
videoSwitchingMethod	VideoSwitchingMethod OPTIONAL,
initialVideoFormat	VideoFormat OPTIONAL,
initialVideoAlgorithm	VideoCoding OPTIONAL,
initialAudioAlgorithm	AudioCoding OPTIONAL,
billingMode	BillingMode OPTIONAL,
organizerBillingAccount	GenericID OPTIONAL,
conferenceOwnerID	GenericID OPTIONAL,
date	Day OPTIONAL,
time	Time OPTIONAL,
duration	ConferenceDuration OPTIONAL,
bookingSites	SET (SIZE(2..MAX)) OF BookingSite OPTIONAL,
siteDefaultJoiningMethod	JoiningMethod OPTIONAL,
conferenceModifierPassword	LatinAlphanumericString OPTIONAL,
conferenceReference	GenericID OPTIONAL,
acknowledgmentType	AcknowledgmentType (simple explicit) DEFAULT simple,
resultCode	ResultCode ,
resultMessage	SimpleTextString OPTIONAL,
userData	UserData OPTIONAL,
...	

}

RSConferenceModifyRequest ::= SEQUENCE {

connectionID	ConnectionID ,
transactionID	TransactionID ,
conferenceReference	GenericID ,
conferenceModifierPassword	LatinAlphanumericString OPTIONAL,
defaultPolicy	ConferenceModificationPolicy DEFAULT keepPrevious,
conferenceName	SimpleTextString OPTIONAL,
conferenceDescription	TextString OPTIONAL,
mcuCascadingMode	MCUCascadingMode OPTIONAL,
audiovisualChairControl	BOOLEAN OPTIONAL,
audiovisualChairmanPassword	LatinAlphanumericString OPTIONAL,
t120ExternalConvener	BOOLEAN OPTIONAL,
t120ConductedMode	BOOLEAN OPTIONAL,
t120ConductorPrivileges	SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
t120CommonPrivileges	SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
t120ConductorPassword	LatinAlphanumericString OPTIONAL,
commonPassword	LatinAlphanumericString OPTIONAL,
globalTransferRate	INTEGER (1..30) OPTIONAL -- $N \times \text{canaux atomiques}$
videoSwitchingMethod	VideoSwitchingMethod OPTIONAL,
initialVideoFormat	VideoFormat OPTIONAL,
initialVideoAlgorithm	VideoCoding OPTIONAL,
initialAudioAlgorithm	AudioCoding OPTIONAL,
billingMode	BillingMode OPTIONAL,
organizerBillingAccount	GenericID OPTIONAL,
conferenceOwnerID	GenericID OPTIONAL,
date	Day OPTIONAL,
time	Time OPTIONAL,
duration	ConferenceDuration OPTIONAL,

```

    bookingSites          CHOICE {
        fullRefresh      SET (SIZE(2..MAX)) OF BookingSite,
        partialRefresh   SET (SIZE(1..MAX)) OF CHOICE {
            deleteSite   GenericID,
            modifySite   BookingSite,
            addSite      BookingSite
        }
    } OPTIONAL,
    siteDefaultJoiningMethod JoiningMethod OPTIONAL,
    newModifierPassword    LatinAlphanumericString OPTIONAL,
    userData               UserData OPTIONAL,
    ...
}

RSConferenceModifyResponse ::= SEQUENCE {
    connectionID          ConnectionID,
    transactionID         TransactionID,
    conferenceReference   GenericID,
    conferenceName        SimpleTextString OPTIONAL,
    conferenceDescription TextString OPTIONAL,
    mcuCascadingMode      MCUCascadingMode OPTIONAL,
    audiovisualChairControl BOOLEAN OPTIONAL,
    audiovisualChairmanPassword LatinAlphanumericString OPTIONAL,
    t120ExternalConvener  BOOLEAN OPTIONAL,
    t120ConductedMode     BOOLEAN OPTIONAL,
    t120ConductorPrivileges SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
    t120CommonPrivileges SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
    t120ConductorPassword LatinAlphanumericString OPTIONAL,
    commonPassword        LatinAlphanumericString OPTIONAL,
    globalTransferRate    INTEGER (1..30) OPTIONAL, -- N x canaux atomiques
    videoSwitchingMethod VideoSwitchingMethod OPTIONAL,
    initialVideoFormat    VideoFormat OPTIONAL,
    initialVideoAlgorithm VideoCoding OPTIONAL,
    initialAudioAlgorithm AudioCoding OPTIONAL,
    billingMode            BillingMode OPTIONAL,
    organizerBillingAccount GenericID OPTIONAL,
    conferenceOwnerID     GenericID OPTIONAL,
    date                  Day OPTIONAL,
    time                  Time OPTIONAL,
    duration              ConferenceDuration OPTIONAL,
    bookingSites          SET (SIZE(2..MAX)) OF BookingSite OPTIONAL,
    siteDefaultJoiningMethod JoiningMethod OPTIONAL,
    newModifierPassword    LatinAlphanumericString OPTIONAL,
    acknowledgmentType    AcknowledgmentType (simple | explicit) DEFAULT simple,
    resultCode            ResultCode,
    resultMessage         SimpleTextString OPTIONAL,
    userData              UserData OPTIONAL,
    ...
}

RSConferenceCancelRequest ::= SEQUENCE {
    connectionID          ConnectionID,
    transactionID         TransactionID,
    conferenceReference   GenericID,
    conferenceModifierPassword LatinAlphanumericString OPTIONAL,
    userData              UserData OPTIONAL,
    ...
}

```

```

RSConferenceCancelResponse ::= SEQUENCE {
    connectionID           ConnectionID,
    transactionID        TransactionID,
    conferenceReference   GenericID,
    acknowledgmentType   AcknowledgmentType (simple) OPTIONAL,
    resultCode           ResultCode,
    resultMessage        SimpleTextString OPTIONAL,
    userData             UserData OPTIONAL,
    ...
}

RSConferenceListInquireRequest ::= SEQUENCE {
    connectionID           ConnectionID,
    transactionID        TransactionID,
    conferenceOwnerIDFilter SET (SIZE(1..MAX)) OF GenericID OPTIONAL,
    conferenceOwnerNameFilter SET (SIZE(1..MAX)) OF SimpleTextStringInterval OPTIONAL,
    conferenceNameFilter    SET (SIZE(1..MAX)) OF SimpleTextStringInterval OPTIONAL,
    conferenceStatusFilter  SET (SIZE(1..4)) OF ConferenceStatus,
    dateFilter           SET (SIZE(1..MAX)) OF DaysInterval OPTIONAL,
    userData             UserData OPTIONAL,
    ...
}

RSConferenceListInquireResponse ::= SEQUENCE {
    connectionID           ConnectionID,
    transactionID        TransactionID,
    conferenceSummaries   SET (SIZE(1..MAX)) OF ConferenceSummary OPTIONAL,
    acknowledgmentType   AcknowledgmentType (noAcknowledgment) OPTIONAL,
    resultCode           ResultCode,
    resultMessage        SimpleTextString OPTIONAL,
    userData             UserData OPTIONAL,
    ...
}

RSConferenceDetailInquireRequest ::= SEQUENCE {
    connectionID           ConnectionID,
    transactionID        TransactionID,
    conferenceReference   GenericID,
    userData             UserData OPTIONAL,
    ...
}

RSConferenceDetailInquireResponse ::= SEQUENCE {
    connectionID           ConnectionID,
    transactionID        TransactionID,
    conferenceReference   GenericID,
    conferenceName       SimpleTextString,
    conferenceDescription TextString DEFAULT "",
    conferenceOwnerID    GenericID DEFAULT connectedSubscriber,
    conferenceOwnerName  SimpleTextString OPTIONAL,
    mcuCascadingMode     MCUCascadingMode DEFAULT unspecified,
    audiovisualChairControl BOOLEAN DEFAULT FALSE,
    audiovisualChairmanPassword LatinAlphanumericString DEFAULT noPassword,
    t120ExternalConvener BOOLEAN DEFAULT FALSE,
    t120ConductedMode    BOOLEAN DEFAULT FALSE,
    t120ConductorPrivileges SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
    t120CommonPrivileges SET (SIZE(1..MAX)) OF Privilege OPTIONAL,
    t120ConductorPassword LatinAlphanumericString DEFAULT noPassword,
    commonPassword      LatinAlphanumericString DEFAULT noPassword,

```

```

globalTransferRate      INTEGER (1..30), -- N × canaux atomiques
videoSwitchingMethod    VideoSwitchingMethod DEFAULT voiceActivityDetection : NULL,
initialVideoFormat      VideoFormat DEFAULT runtimeChoice,
initialVideoAlgorithm    VideoCoding DEFAULT runtimeChoice,
initialAudioAlgorithm    AudioCoding DEFAULT runtimeChoice,
billingMode             BillingMode DEFAULT implicitBillingMode,
organizerBillingAccount GenericID DEFAULT implicitAccount,
date                   Day,
time                   Time,
duration               ConferenceDuration,
bookingSites           SET (SIZE(2..MAX)) OF BookingSite,
siteDefaultJoiningMethod JoiningMethod,
conferenceModifierPassword LatinAlphanumericString DEFAULT noPassword,
modifyRight            BOOLEAN DEFAULT TRUE,
acknowledgmentType     AcknowledgmentType (noAcknowledgment) OPTIONAL,
resultCode              ResultCode,
resultMessage           SimpleTextString OPTIONAL,
userData                UserData OPTIONAL,
...
}

```

```

RSConferenceCheckAvailabilityRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    dateIntervals     SET (SIZE(1..MAX)) OF DaysInterval,
    applyingSites     SET (SIZE(2..MAX)) OF ApplyingSite,
    minimumTransferRate INTEGER (1..30) , -- N × canaux atomiques
    userData          UserData OPTIONAL,
    ...
}

```

```

RSConferenceCheckAvailabilityResponse ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    queryResultSET (SIZE(1..MAX)) OF SEQUENCE {
        daysInterval DaysInterval,
        possibilities SET (SIZE(1..MAX)) OF SEQUENCE {
            timeInterval SET (SIZE(1..MAX)) OF TimeInterval,
            maximumTransferRate INTEGER (1..30) , -- N × canaux atomiques
            interfaceRestrictions SET (SIZE(1..MAX)) OF SEQUENCE {
                siteID      GenericID,
                interfaces   CHOICE {
                    allowedInterfaces SET (SIZE(1..MAX)) OF GenericID,
                    forbiddenInterfaces SET (SIZE(1..MAX)) OF GenericID
                }
            } OPTIONAL -- toutes les surfaces sont autorisées par défaut
        }
    } OPTIONAL,
    resultCode        ResultCode,
    resultMessage      SimpleTextString OPTIONAL,
    userData           UserData OPTIONAL,
    ...
}

```

```

RSSiteRecordRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    permanentRecord   BOOLEAN DEFAULT FALSE,
}

```

```

siteType           SiteType,
siteClass          SiteClass OPTIONAL,
siteOwnerID       GenericID DEFAULT connectedSubscriber,
siteName          SimpleTextString OPTIONAL,
geographicInformation SEQUENCE {
    country        SimpleTextString OPTIONAL,
    region         SimpleTextString OPTIONAL,
    city           SimpleTextString OPTIONAL,
    postalAddress  SimpleTextString OPTIONAL
} OPTIONAL,
networkInterfaces SET (SIZE(1..MAX)) OF NetworkInterface,
equipmentDescription EquipmentDescription OPTIONAL,
contactPerson     PersonDescription OPTIONAL,
userData          UserData OPTIONAL,
...
}

```

```

RSSiteRecordResponse ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    permanentRecord   BOOLEAN OPTIONAL,
    siteType          SiteType OPTIONAL,
    siteClass         SiteClass OPTIONAL,
    siteOwnerID       GenericID OPTIONAL,
    siteName          SimpleTextString OPTIONAL,
    geographicInformation SEQUENCE {
        country        SimpleTextString OPTIONAL,
        region         SimpleTextString OPTIONAL,
        city           SimpleTextString OPTIONAL,
        postalAddress  SimpleTextString OPTIONAL
    } OPTIONAL,
    networkInterfaces SET (SIZE(1..MAX)) OF NetworkInterface,
    equipmentDescription EquipmentDescription OPTIONAL,
    contactPerson     PersonDescription OPTIONAL,
    siteID            GenericID OPTIONAL,
    deletionDate      Day OPTIONAL,
    acknowledgmentType AcknowledgmentType (simple) OPTIONAL,
    resultCode        ResultCode,
    resultMessage     SimpleTextString OPTIONAL,
    userData          UserData OPTIONAL,
    ...
}

```

```

RSSiteModifyRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    siteID            GenericID,
    permanentRecord   BOOLEAN OPTIONAL,
    siteType          SiteType OPTIONAL,
    siteClass         SiteClass OPTIONAL,
    siteOwnerID       GenericID OPTIONAL,
    siteName          SimpleTextString OPTIONAL,
    geographicInformation SEQUENCE {
        country        SimpleTextString OPTIONAL,
        region         SimpleTextString OPTIONAL,
        city           SimpleTextString OPTIONAL,
        postalAddress  SimpleTextString OPTIONAL
    } OPTIONAL,
    networkInterfaces CHOICE {
        fullRefresh   SET (SIZE(1..MAX)) OF NetworkInterface,
        partialRefresh SET (SIZE(1..MAX)) OF CHOICE {

```

```

        deleteInterface    GenericID,
        modifyInterface    SEQUENCE {
            interfaceID    GenericID,
            interface      NetworkInterface
        },
        addInterface      NetworkInterface
    }
} OPTIONAL,
equipmentDescription    EquipmentDescription OPTIONAL,
contactPerson           PersonDescription OPTIONAL,
userData                UserData OPTIONAL,
...
}

RSSiteModifyResponse ::= SEQUENCE {
    connectionID          ConnectionID,
    transactionID         TransactionID,
    siteID                GenericID OPTIONAL,
    permanentRecord      BOOLEAN OPTIONAL,
    siteType              SiteType OPTIONAL,
    siteClass             SiteClass OPTIONAL,
    siteOwnerID          GenericID OPTIONAL,
    siteName              SimpleTextString,
    geographicInformation SEQUENCE {
        country           SimpleTextString OPTIONAL,
        region            SimpleTextString OPTIONAL,
        city              SimpleTextString OPTIONAL,
        postalAddress     SimpleTextString OPTIONAL
    } OPTIONAL,
    networkInterfaces     SET (SIZE(1..MAX)) OF NetworkInterface OPTIONAL,
    equipmentDescription  EquipmentDescription OPTIONAL,
    contactPerson         PersonDescription OPTIONAL,
    deletionDate          Day OPTIONAL,
    acknowledgmentType    AcknowledgmentType (simple) OPTIONAL,
    resultCode            ResultCode,
    resultMessage         SimpleTextString OPTIONAL,
    userData              UserData OPTIONAL,
    ...
}

RSSiteDeleteRequest ::= SEQUENCE {
    connectionID          ConnectionID,
    transactionID         TransactionID,
    siteID                GenericID,
    userData              UserData OPTIONAL,
    ...
}

RSSiteDeleteResponse ::= SEQUENCE {
    connectionID          ConnectionID,
    transactionID         TransactionID,
    siteID                GenericID OPTIONAL,
    acknowledgmentType    AcknowledgmentType (simple) OPTIONAL,
    resultCode            ResultCode,
    resultMessage         SimpleTextString OPTIONAL,
    userData              UserData OPTIONAL,
    ...
}

```

```

RSSiteDirectoryInquireRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    siteOwnerIDFilter SET (SIZE(1..MAX)) OF GenericID OPTIONAL,
    siteOwnerNameFilter SET (SIZE(1..MAX)) OF SimpleTextStringInterval OPTIONAL,
    siteNameFilter    SET (SIZE(1..MAX)) OF SimpleTextStringInterval OPTIONAL,
    geographicLocationFilter SET (SIZE(1..MAX)) OF SEQUENCE {
        country      SimpleTextString OPTIONAL,
        region       SimpleTextString OPTIONAL,
        city         SimpleTextString OPTIONAL,
        postalAddress SimpleTextString OPTIONAL
    } OPTIONAL,
    userData          UserData OPTIONAL,
    ...
}

```

```

RSSiteDirectoryInquireResponse ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    queryResultSET (SIZE(1..MAX)) OF SEQUENCE {
        siteType      SiteType,
        siteClass     SiteClass OPTIONAL,
        siteOwnerID   GenericID OPTIONAL,
        siteOwnerName SimpleTextString OPTIONAL,
        siteName      SimpleTextString,
        geographicInformation SEQUENCE {
            country      SimpleTextString OPTIONAL,
            region       SimpleTextString OPTIONAL,
            city         SimpleTextString OPTIONAL,
            postalAddress SimpleTextString OPTIONAL
        } OPTIONAL,
        networkInterfaces SET (SIZE(1..MAX)) OF NetworkInterface,
        equipmentDescription EquipmentDescription OPTIONAL,
        contactPerson     PersonDescription OPTIONAL,
        siteExtraInformation TextString OPTIONAL,
        deletionDate      Day OPTIONAL,
        modifyRight       BOOLEAN DEFAULT TRUE,
        ...
    } OPTIONAL,
    resultCode          ResultCode,
    resultMessage       SimpleTextString OPTIONAL,
    userData            UserData OPTIONAL,
    ...
}

```

```

RSNonStandardRequest ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    parentTransactionID TransactionID OPTIONAL,
    protocolKey       Key OPTIONAL,
    data              OCTET STRING,
    ...
}

```

```

RSNonStandardResponse ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    acknowledgmentType AcknowledgmentType,

```

```

    protocolKey      Key OPTIONAL,
    data             OCTET STRING,
    ...
}

```

```

RSNonStandardIndication ::= SEQUENCE {
    connectionID      ConnectionID,
    relatedTransactionID TransactionID OPTIONAL,
    protocolKey      Key OPTIONAL,
    data             OCTET STRING,
    ...
}

```

```

RSTransactionAcknowledgeIndication ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    ...
}

```

```

RSTransactionCancelIndication ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    ...
}

```

```

RSTransactionErrorIndication ::= SEQUENCE {
    connectionID      ConnectionID,
    transactionID     TransactionID,
    errorCode         CHOICE {
        unrecognizedNonStandardProtocol NULL,
        defaultErrorCode                NULL,
        nonStandard                     NonStandardParameter,
        ...
    },
    ...
}

```

```

-----
-- Liste de toutes les unités PDU de transactions URST
-----

```

```

URSTPDU ::= CHOICE {
    rsConnectRequest           RSConnectRequest,
    rsConnectResponse         RSConnectResponse,
    rsDisconnectIndication    RSDisconnectIndication,
    rsConferenceReserveRequest RSConferenceReserveRequest,
    rsConferenceReserveResponse RSConferenceReserveResponse,
    rsConferenceModifyRequest  RSConferenceModifyRequest,
    rsConferenceModifyResponse RSConferenceModifyResponse,
    rsConferenceCancelRequest  RSConferenceCancelRequest,
    rsConferenceCancelResponse RSConferenceCancelResponse,
    rsConferenceListInquireRequest RSConferenceListInquireRequest,
    rsConferenceListInquireResponse RSConferenceListInquireResponse,
    rsConferenceDetailInquireRequest RSConferenceDetailInquireRequest,
    rsConferenceDetailInquireResponse RSConferenceDetailInquireResponse,
    rsConferenceCheckAvailabilityRequest RSConferenceCheckAvailabilityRequest,
    rsConferenceCheckAvailabilityResponse RSConferenceCheckAvailabilityResponse,
    rsSiteRecordRequest       RSSiteRecordRequest,
    rsSiteRecordResponse      RSSiteRecordResponse,
}

```

rsSiteModifyRequest
rsSiteModifyResponse
rsSiteDeleteRequest
rsSiteDeleteResponse
rsSiteDisrectoryInquireRequest
rsSiteDisrectoryInquireResponse
rsNonStandardRequest
rsNonStandardResponse
rsNonStandardIndication
rsTransactionAcknowledgeIndication
rsTransactionCancelIndication
rsTransactionErrorIndication

...

}

END

RSSiteModifyRequest,
RSSiteModifyResponse,
RSSiteDeleteRequest,
RSSiteDeleteResponse,
RSSiteDirectoryInquireRequest,
RSSiteDirectoryInquireResponse,
RSNonStandardRequest,
RSNonStandardResponse,
RSNonStandardIndication,
RSTransactionAcknowledgeIndication,
RSTransactionCancelIndication,
RSTransactionErrorIndication,

SERIES DES RECOMMANDATIONS UIT-T

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