

Union internationale des télécommunications

# UIT-T

SECTEUR DE LA NORMALISATION  
DES TÉLÉCOMMUNICATIONS  
DE L'UIT

# T.124

(01/2007)

SÉRIE T: TERMINAUX DES SERVICES  
TÉLÉMATIQUES

Protocoles de transmission de données pour les  
conférences multimédias

---

## Commande générique de conférence

Recommandation UIT-T T.124

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

Télécopie – Cadre général	T.0–T.19
Compression des images fixes – Mires	T.20–T.29
Télécopie – Protocoles du groupe 3	T.30–T.39
Représentation des couleurs	T.40–T.49
Codage des caractères	T.50–T.59
Télécopie – Protocoles du groupe 4	T.60–T.69
Services télématiques – Cadre général	T.70–T.79
Compression des images fixes – JPEG-1, deux tons et JBIG	T.80–T.89
Services télématiques – Terminaux et protocoles du RNIS	T.90–T.99
Vidéotexte – Cadre général	T.100–T.109
<b>Protocoles de transmission de données pour les conférences multimédias</b>	<b>T.120–T.149</b>
Téléécriture	T.150–T.159
Cadre général multimédia et hypermédia	T.170–T.189
Traitement coopératif des documents	T.190–T.199
Services télématiques – Interfonctionnement	T.300–T.399
Architecture de document ouverte	T.400–T.429
Transfert et manipulation des documents	T.430–T.449
Profil d'application de document	T.500–T.509
Profil d'application de communication	T.510–T.559
Services télématiques – Caractéristiques des équipements	T.560–T.649
Compression des images fixes – JPEG 2000	T.800–T.849
Compression des images fixes – Extensions du système JPEG-1	T.850–T.899

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

# Recommandation UIT-T T.124

## Commande générique de conférence

### Résumé

La présente Recommandation fournit un cadre général à haut niveau pour la gestion et la commande de conférences de terminaux audiographiques et audiovisuels et d'unités de contrôle multipoint (MCU, *multipoint control unit*). Elle comprend des fonctions génériques de commande conférence (GCC, *generic conference control*) telles que l'établissement et la fin de conférences, la gestion du répertoire des nœuds participant à une conférence, la gestion du répertoire des entités de protocole d'application et des capacités d'application présentes au sein d'une conférence, des services de référentiel disponibles pour les entités de protocole d'application, la coordination de la présidence de la conférence ainsi que d'autres fonctions diverses. Elle dépend des Recommandations associées T.122 et T.125 et de la Recommandation T.123 faisant partie de l'infrastructure T.120.

La présente version révisée de la Recommandation UIT-T T.124 apporte un certain nombre de clarifications à la version précédente.

### Source

La Recommandation UIT-T T.124 a été approuvée le 13 janvier 2007 par la Commission d'études 16 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8.

## AVANT-PROPOS

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

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

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

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

## NOTE

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

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

## DROITS DE PROPRIÉTÉ INTELLECTUELLE

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

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux développeurs de consulter la base de données des brevets du TSB sous <http://www.itu.int/ITU-T/ipr/>.

© UIT 2008

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

## TABLE DES MATIÈRES

	<b>Page</b>
1	Domaine d'application ..... 1
2	Références normatives..... 2
3	Définitions ..... 3
4	Abréviations..... 7
5	Conventions ..... 8
6	Aperçu général..... 8
6.1	Modèle systémique de nœud de conférence ..... 10
6.2	Etablissement et fin de la conférence ..... 11
6.3	Le répertoire de conférence ..... 12
6.4	Le répertoire d'application ..... 12
6.5	Le référentiel d'application ..... 12
6.6	La présidence de la conférence..... 12
6.7	Fonctions diverses ..... 13
6.8	Conférences échelonnables ..... 13
6.9	Résumé des services abstraits GCC ..... 15
7	Définition du service GCC ..... 18
7.1	Etablissement et fin de conférence ..... 18
7.2	Le répertoire de conférence ..... 57
7.3	Le répertoire d'application ..... 63
7.4	Le référentiel d'application ..... 79
7.5	Présidence de conférence ..... 90
7.6	Fonctions diverses ..... 100
8	Spécification du protocole GCC..... 105
8.1	Fonctionnement général ..... 105
8.2	Etablissement et fin de conférence ..... 106
8.3	Les répertoires d'application et de conférence..... 138
8.4	Le référentiel d'application ..... 163
8.5	Présidence de conférence ..... 173
8.6	Fonctions diverses ..... 180
8.7	Définitions des GCCPDU..... 182
9	Utilisation du service de communication multipoint..... 204
9.1	Services MCS ..... 204
9.2	Affectation de canaux..... 205
9.3	Attribution de jetons ..... 206
9.4	Utilisation des services de transmission de données MCS..... 206
9.5	Codage des PDU dans les primitives MCS ..... 209

	<b>Page</b>	
9.6	Format de paramètre "données utilisateur" pour la primitive MCS-Connect-Provider.....	210
9.7	Interprétation du sélecteur de domaine MCS .....	210
Annexe A – Affectations statiques d'identificateur de canal et de jeton.....		212
A.1	Affectations statiques d'identificateur de canal .....	212
A.2	Affectations statiques d'identificateur de jeton.....	212
Annexe B – Affectations d'identificateurs d'objet .....		213
Annexe C – Paramètre Adresse réseau – Description et utilisation.....		214
Appendice I – Relation entre systèmes T.120 et H.243 dans des conférences H.320 .....		215
I.1	Introduction .....	215
I.2	Sélection des conférences et protection par mot de passe.....	215
I.3	Variante d'identificateur nodal .....	216

# **Recommandation UIT-T T.124**

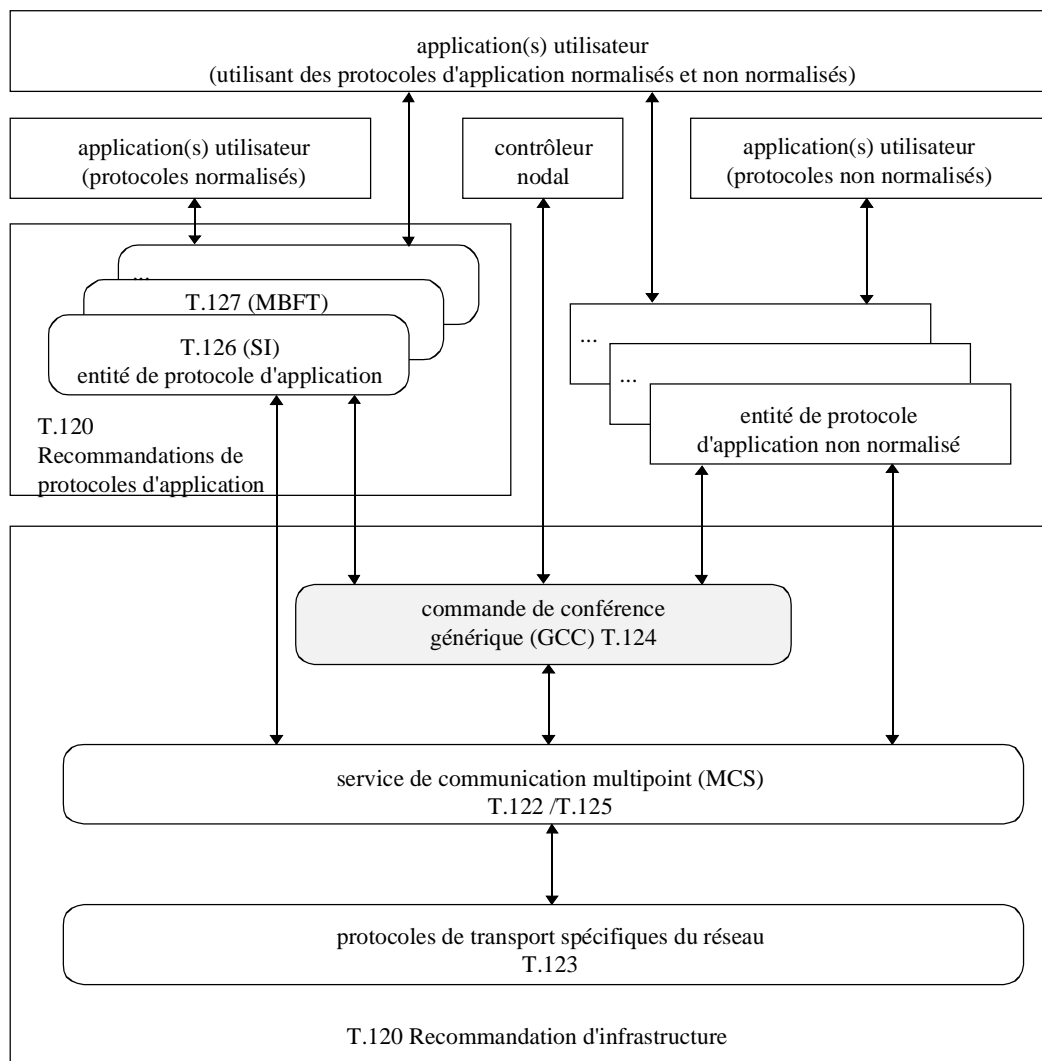
## **Commande générique de conférence**

### **1 Domaine d'application**

La présente Recommandation fournit un cadre général à haut niveau pour la gestion et la commande de conférences de terminaux audiographiques et audiovisuels et d'unités de commande multipoint (MCU). Elle comprend des fonctions de la commande générique de conférence (GCC) telles que l'établissement et la fin de conférences, la gestion du répertoire des nœuds participant à une conférence, la gestion du répertoire des entités de protocole d'application et des capacités d'application présentes au sein d'une conférence, des services de référentiel disponibles pour les entités de protocole d'application, la coordination de la présidence de la conférence ainsi que d'autres fonctions diverses.

La présente Recommandation est définie dans le cadre général de [UIT-T T.120] incluant les Recommandations associées T.122 et T.125 qui définissent le mécanisme de livraison multipoint utilisé par la présente Recommandation ainsi que [UIT-T T.123] qui spécifie les piles de protocole audiovisuel pour chacun des réseaux de communication pris en charge.

La Figure 1-1 donne un aperçu général du domaine d'application de la présente Recommandation et de ses relations avec les autres éléments du cadre général T.120 au sein d'un nœud donné.



T0819400-94

**Figure 1-1 – Domaine d'application de la Recommandation T.124**

## 2 Références normatives

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

- [UIT-T F.702]      Recommandation UIT-T F.702 (1996), *Services de conférence multimédia*.
- [UIT-T H.221]      Recommandation UIT-T H.221 (2004), *Structure de trame pour un canal d'un débit de 64 à 1920 kbit/s pour les téléservices audiovisuels*.
- [UIT-T T.35]        Recommandation UIT-T T.35 (2001), *Procédure d'attribution des codes définis par l'UIT-T pour les facilités non normalisées*.
- [UIT-T T.120]      Recommandation UIT-T T.120 (2007), *Protocoles de données pour conférence multimédia*.



- [UIT-T T.122]      Recommandation UIT-T T.122 (1998), *Service de communication multipoint – Définition du service.*
- [UIT-T T.123]      Recommandation UIT-T T.123 (2007), *Piles de protocole de données propres au réseau pour conférences multimédias.*
- [UIT-T T.125]      Recommandation UIT-T T.125 (1998), *Spécification du protocole du service de communication multipoint.*
- [UIT-T T.126]      Recommandation UIT-T T.126 (1997), *Protocole du service multipoint d'imagerie fixe et d'annotation.*
- [UIT-T T.127]      Recommandation UIT-T T.127 (1995), *Protocole de transfert multipoint de fichiers binaires.*
- [UIT-T X.680]      Recommandation UIT-T X.680 (2002) | ISO/CEI 8824-1:2002, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- [UIT-T X.691]      Recommandation UIT-T X.691 (2002) | ISO/CEI 8825-2:2002, *Technologies de l'information – Règles de codage de la notation de syntaxe abstraite numéro un: spécification des règles de codage compact.*
- [ISO/CEI 10646]      ISO/CEI 10646:2003, *Technologies de l'information – Jeu universel de caractères codés à plusieurs octets.*

### 3 Définitions

La présente Recommandation définit les termes suivants:

**3.1 nœud anonyme:** catégorie de nœud T.120 qui peut participer à une conférence tout en restant invisible à tous les autres nœuds de cette conférence. La présence de nœuds anonymes est nécessaire afin de faciliter les conférences T.120 à grande échelle.

**3.2 protocole d'application:** tout protocole, normalisé ou non, conçu pour utiliser des services T.120.

**3.3 entité de protocole d'application:** instanciation d'un protocole d'application au sein d'un terminal ou d'une MCU. Les entités de protocole d'application sont mises en œuvre par des applications utilisateur mais ne sont pas des applications utilisateur. Une entité de protocole d'application communique avec le service GCC au moyen d'un point d'accès GCCSAP qui lui permet de se mettre en relation avec le fournisseur GCC présent sur son terminal local ou sur son MCU. Les communications avec des fournisseurs GCC sont assurées uniquement par les entités de protocole d'application et non par les applications utilisateur. De multiples entités de protocole d'application basées sur un même protocole d'application peuvent s'inscrire sur un nœud donné. Elles peuvent appartenir à une seule ou à plusieurs sessions de protocole d'application. Une entité de protocole d'application donnée communique avec le fournisseur GCC local au moyen d'un seul fournisseur GCCSAP et possède un seul identificateur d'utilisateur MCS si elle s'est inscrite dans l'état actif. Une entité de protocole d'application qui s'est inscrite dans l'état actif fait partie d'une seule session de protocole d'application identifiée par son identificateur de session. L'absence d'identificateur de session indique l'appartenance à la session par défaut. Une entité de protocole d'application qui s'est inscrite dans l'état inactif est considérée comme ne faisant partie d'aucune session de protocole d'application. Une telle entité peut toutefois utiliser les services GCC et l'information qui lui est associée figure dans le répertoire d'application.

**3.4 clé de protocole d'application:** la clé de protocole d'application identifie le type de protocole d'application pour une session de protocole d'application. De multiples sessions de protocole d'application du même type seront identifiées en utilisant la même clé de protocole d'application mais d'identificateurs de session différents. Une clé de protocole d'application est soit

un identificateur d'objet ASN.1 appartenant à une Recommandation, une Norme ou un protocole non normalisé, soit un identificateur non normalisé utilisant les conventions de codage de [UIT-T H.221].

**3.5 session de protocole d'application:** ensemble d'entités de protocole d'application homologues.

**3.6 enregistrement d'application:** ensemble d'informations situées sur un nœud donné et concernant une entité de protocole d'application donnée. Cet ensemble comprend l'identificateur d'utilisateur de l'application, l'indicateur d'activité ainsi que d'autres paramètres.

**3.7 référentiel d'application:** référentiel central, localisé sur le fournisseur GCC sommital. Une entité de protocole d'application peut y enregistrer son utilisation de jetons, de canaux et d'autres paramètres. Des entités de protocole d'application homologues peuvent par la suite accéder au référentiel pour retrouver ces informations.

**3.8 répertoire d'application:** ensemble de tous les enregistrements de toutes les entités de protocole d'application inscrites sur tous les nœuds d'une conférence, ainsi que la liste des capacités d'application de chaque session de protocole d'application.

**3.9 identificateur d'utilisateur d'application:** identificateur d'utilisateur MCS attribué par le service MCS à une entité de protocole d'application.

**3.10 mode présidé:** le mode présidé permet à un nœud président de piloter les entités de protocole d'application dans tous les nœuds d'une conférence et de limiter l'utilisation des entités de protocole d'application faite par d'autres nœuds. Le mode présidé est instauré quand un président a été affecté à une conférence. Ceci a lieu lorsqu'un nœud a réussi à s'emparer du jeton de président.

**3.11 président:** le président, s'il existe, est un nœud d'une conférence pilotant certains aspects de la conférence comme la supervision de la communication entre entités de protocole d'application, la supervision des participants d'une conférence et la fin de la conférence. Une conférence possède un seul président ou aucun. Un nœud devient président soit en s'emparant du jeton de président, soit en demandant ou en acceptant la présidence de la conférence du président en exercice.

**3.12 conférence:** un ensemble de nœuds qui sont liés et capables d'échanger des informations audiographiques et audiovisuelles à travers divers réseaux de communication.

**3.13 répertoire d'application de conférence:** base de données gérée par chaque fournisseur GCC qui comprend un ensemble d'enregistrements d'application pour chaque entité de protocole d'application de chaque nœud de la conférence, ainsi que d'autres informations telles que la liste des capacités d'application de chaque session de protocole d'application.

**3.14 mode de conférence:** lorsqu'une conférence est créée, l'organisateur spécifie un mode de conférence qui définit les catégories nodales qui seront autorisées à participer à la conférence. Les modes de conférence sont les suivants: *conventionnels-seulement*, qui n'autorise que des nœuds conventionnels à participer à la conférence; *comptés-contrôlés*, qui n'autorise que les nœuds comptés et contrôlés à participer à la conférence; *anonymes-contrôlés*, qui n'autorise que les nœuds comptés et conventionnels à participer à la conférence; et le *mode sans restriction*, qui autorise les trois catégories nodales à participer à la conférence. Le contrôleur nodal doit utiliser le mot de passe T.124 pour déterminer les nœuds qui peuvent participer à titre conventionnel lorsque le mode de conférence est soit *compté-contrôlé* soit *anonyme-contrôlé*.

**3.15 profil de conférence:** base de données gérée par chaque fournisseur GCC constituée de l'information relative à la conférence dans son ensemble, telle que le nom de la conférence, le mot de passe éventuel, etc.

**3.16 répertoire de conférence:** base de données gérée par chaque fournisseur GCC constituée d'une liste des nœuds de la conférence. Cette liste contient pour chaque nœud l'identificateur du

nœud, le type de nœud, le nom du nœud et peut contenir une liste de participants du nœud ainsi que d'autres informations optionnelles.

**3.17 point d'accès de contrôle au service GCC:** interface de communication entre un fournisseur GCC et le contrôleur nodal au sein d'un nœud donné.

**3.18 organisateur:** nœud qui a créé la conférence en émettant la primitive de demande GCC-Conference-Create (création d'une conférence).

**3.19 mot de passe de l'organisateur:** chaîne numérique, associée à une chaîne de caractères optionnelle, pouvant être utilisée lors de la création d'une conférence afin de permettre à l'organisateur de quitter la conférence et d'y revenir par la suite en utilisant le mot de passe pour retrouver les privilèges d'organisateur. Pour que ceci soit possible, il est nécessaire qu'un mot de passe d'organisateur soit fourni, sous l'une des deux formes possibles, lors de la création de la conférence. L'utilisation du mot de passe d'organisateur correct permet d'entrer dans une conférence verrouillée, mais ne dispense pas de fournir le mot de passe correct pour entrer dans une conférence protégée par mot de passe.

**3.20 nœud conventionnel:** catégorie de nœud T.120 la plus élémentaire, qui définit un nœud à capacités entières qui est inclus dans tous les échanges de répertoire.

**3.21 nœud compté:** catégorie nodale ayant les caractéristiques suivantes: le nœud apparaît dans les répertoires de nœuds conventionnels, il n'a pas d'incidence sur les capacités de conférence et il peut participer à des sessions d'entités de protocole d'application (APE) en mode acquitté. Un nœud compté permet aux autres nœuds de participer à des conférences assez grandes, composées d'un petit nombre de nœuds conventionnels (qui fournissent le contenu) et d'un grand nombre de nœuds comptés (qui peuvent accuser réception du contenu extrait de la source).

**3.22 session par défaut:** des entités de protocole d'application actives n'ayant pas d'identificateur de session dans leur clé de session sont considérées comme faisant partie d'une seule session appelée session par défaut.

**3.23 identificateur d'entité:** identificateur numérique de 16 bits utilisé pour identifier chaque entité de protocole d'application inscrite dans un nœud. La valeur de l'identificateur d'entité est unique pour toutes les entités de protocole d'application d'un nœud donné. Elle n'a pas besoin d'être unique entre nœuds. Une entité de protocole d'application particulière peut être identifiée sans ambiguïté par la combinaison de l'identificateur d'entité et de l'identificateur de nœud sur lequel est inscrite l'entité de protocole d'application.

**3.24 fournisseur GCC:** agent fournissant des services GCC au contrôleur nodal local et aux entités de protocole d'application locales au niveau d'un terminal ou d'une MCU.

**3.25 point d'accès au service GCC:** interface de communication entre un fournisseur GCC et une entité de protocole d'application au sein d'un nœud donné.

**3.26 poignée:** nombre entier de 32 bits attribué par le fournisseur GCC sommital lors de l'utilisation de la primitive de demande GCC-Registry-Allocate-Handle (allocation de poignée). Les valeurs de chaque poignée sont uniques au sein d'une conférence.

**3.27 répertoire d'application local:** base de données gérée par tout fournisseur GCC comportant, pour toute entité de protocole d'application inscrite localement auprès de ce fournisseur GCC, un enregistrement d'application ainsi que d'autres informations. Cette information est utilisée pour constituer l'information échangée avec les autres nœuds en vue de créer le répertoire d'application de la conférence.

**3.28 domaine MCS:** hiérarchie de connexions MCS entre nœuds. Les nœuds peuvent utiliser les services MCS pour communiquer à l'intérieur d'un domaine donné, mais non entre domaines distincts. Il existe une correspondance un pour un entre une conférence GCC et un domaine MCS donné.

- 3.29 sélecteur de domaine MCS:** identificateur local non ambigu d'un domaine MCS.
- 3.30 identificateur d'utilisateur MCS:** numéro d'identification non ambigu attribué par le service MCS à un utilisateur MCS. Les fournisseurs GCC ainsi que les entités de protocole d'application sont des utilisateurs MCS. Un identificateur d'utilisateur MCS attribué à un fournisseur de service GCC est appelé identificateur de nœud. Un identificateur d'utilisateur MCS attribué à une entité de protocole d'application est appelé identificateur d'utilisateur d'application. Un identificateur d'utilisateur MCS est valide uniquement au sein d'un domaine MCS.
- 3.31 multipoint:** faculté d'échanger simultanément des données entre des nœuds multiples, par opposition avec l'échange point à point entre deux nœuds connectés directement.
- 3.32 pont de conférence; unité de commande multipoint:** dispositif à accès multiple, servant à interconnecter des terminaux ou d'autres MCU en mode multipoint. Une MCU ayant des capacités GCC réalise les services GCC et MCS. Une MCU n'est pas conçue en premier lieu comme un point de terminaison pour une communication utilisateur.
- 3.33 terminal multiaccès:** équipement de point de terminaison audiographique ou audiovisuel qui a également la capacité de relayer une information T.120. Le comportement d'un terminal multiaccès est caractérisé, comme celui d'un terminal, par l'établissement automatique d'une conférence unique. Toutefois, comme pour une MCU, il peut y avoir pour une conférence donnée dans un terminal multiaccès plus d'une connexion de service MCS.
- 3.34 nœud:** terminal, terminal multiaccès ou MCU. Un nœud donné contient un fournisseur GCC unique. Un nœud unique peut être constitué d'un ou de plusieurs dispositifs physiques. De même un dispositif physique unique peut héberger plusieurs nœuds logiques.
- 3.35 catégorie nodale:** chaque nœud T.120 entre dans l'une des trois catégories suivantes: conventionnels, comptés ou anonymes. Ces catégories nodales permettent différents degrés d'échelonnabilité à l'intérieur d'une conférence T.120.
- 3.36 contrôleur nodal:** entité fonctionnelle, associée à chaque terminal ou MCU, servant de contrôleur pour un nœud.
- 3.37 identificateur de nœud:** identificateur d'utilisateur de service MCS attribué par le service MCS au fournisseur GCC d'un nœud.
- 3.38 mode non présidé:** mode dans lequel une conférence n'a pas de président.
- 3.39 participant:** personne participant à une conférence au niveau d'un nœud.
- 3.40 mot de passe:** chaîne numérique, associée à une chaîne de caractères optionnelle, pouvant être spécifiée au moment de la création d'une conférence. Dans ce cas, un nœud qui tente d'entrer dans une conférence fournira, sous l'une des deux formes possibles, ce mot de passe dans la primitive GCC-Conference-Join afin d'être accepté dans la conférence.
- 3.41 entité de protocole d'application homologue:** entités de protocole d'application qui se sont inscrites dans l'état actif en utilisant des clés de session identiques, y compris la partie identificateur de session de la clé de session. Des protocoles d'application homologues sont ceux qui peuvent communiquer entre eux au cours d'une conférence.
- 3.42 ressource:** objet pouvant être utilisé et partagé par des nœuds d'une conférence. Un canal ou un jeton est une ressource.
- 3.43 identificateur de session:** paramètre optionnel pouvant faire partie d'une clé de session et utilisé pour faire une distinction entre des ensembles multiples d'entités de protocole d'application basées sur le même protocole d'application. Les membres de chaque ensemble d'entités de protocole d'application, définis par l'utilisation d'un protocole d'application commun et d'identificateurs de session identiques, communiquent entre eux d'une manière séparée. Les entités de protocole d'application actives n'ayant pas d'identificateur de session présent dans leur clé de session font

partie d'une session unique appelée session par défaut. Un ensemble d'entités de protocole d'application homologues est appelé session de protocole d'application. Les identificateurs de session se présentent sous la forme d'identificateurs de canal MCS.

**3.44 clé de session:** identificateur commun à des entités de protocole d'application homologues. Une clé de session est constituée de deux composants. Le premier, la clé de protocole d'application identifie le type de protocole d'application. Le second composant est une partie optionnelle de la clé de session, l'identificateur de session qui identifie une session particulière de ce protocole d'application. L'absence de la clé de session indique la session par défaut. Les entités de protocole d'application qui ont en commun la totalité de la clé de session, comprenant aussi bien la clé de protocole d'application que l'identificateur éventuel de session, sont considérées comme des entités de protocole d'application homologues.

**3.45 terminal:** équipement audiographique ou audiovisuel de point de terminaison. Un terminal fournissant les capacités GCC exploite les services GCC et MCS. Un terminal ne peut posséder, pour une conférence donnée, qu'une seule connexion de service MCS.

**3.46 fournisseur GCC sommital:** fournisseur GCC corésident avec le fournisseur MCS sommital dans une conférence. Le fournisseur GCC sommital a des responsabilités que les autres fournisseurs GCC n'ont pas besoin de prendre en charge. La localisation du fournisseur GCC sommital ne change pas au cours d'une conférence.

**3.47 Unicode:** format multilingue de chaîne de texte défini par [ISO/CEI 10646] – plan de base multilingue.

**3.48 colonne 00 Unicode:** sous-ensemble du répertoire Unicode constitué de 256 positions de code contenant les jeux de caractères Basic Latin et Latin-1 supplémentaire ainsi que des caractères de contrôle et des codes réservés.

**3.49 application utilisateur:** entité utilisant une ou plusieurs entités de protocole d'application. Le domaine d'une application utilisateur se limite aux tâches qui n'influencent pas l'interprétation des informations transportées entre des entités de protocole d'application homologues, par exemple la présentation à l'utilisateur final. En conséquence, l'interopérabilité entre les entités de protocole d'application mise en œuvre par les applications utilisateur et leurs entités de protocole d'application homologues ne nécessite, du point de vue des applications utilisateur, aucune spécification par une Recommandation de norme ou par tout autre moyen.

## 4 Abréviations

La présente Recommandation utilise les abréviations suivantes:

GCC	commande générique de conférence ( <i>generic conference control</i> )
GCCSAP	point d'accès au service commande générique de conférence ( <i>generic conference control service access point</i> )
MCS	service de communication multipoint ( <i>multipoint communication service</i> )
MCU	unité de commande multipoint ( <i>multipoint control unit</i> )
PDU	unité de données protocolaire ( <i>protocol data unit</i> )
RDCC	réseau de données à commutation de circuits
RDCP	réseau de données à commutation par paquets
RNIS	réseau numérique à intégration de services
RTPC	réseau téléphonique public commuté

La présente Recommandation utilise les abréviations suivantes, définies dans [UIT-T T.122]:

MCS	service de communication multipoint
-----	-------------------------------------

## 5 Conventions

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

M le paramètre est obligatoire.

C le paramètre est conditionnel.

O le paramètre est optionnel.

Blanc le paramètre est absent.

(=) la valeur du paramètre est identique à la valeur du paramètre correspondant de la primitive précédente, l'ordre de précedence étant: demande, indication, réponse et confirmation.

(=RQ) la valeur du paramètre est identique à la valeur du paramètre correspondant dans une primitive précédente, avec RQ = demande, IN = indication, RS = réponse et CF = confirmation.

Les primitives peuvent posséder jusqu'à quatre types de catégories: 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:

- primitive de demande: émise par un contrôleur nodal ou une entité de protocole d'application en vue de lancer une action donnée;
- primitive d'indication: émise par un fournisseur GCC soit comme résultat d'une primitive de demande, soit comme résultat d'une action lancée par la fonction GCC;
- primitive de réponse: émise par un contrôleur nodal ou une entité de protocole d'application en réponse à une primitive d'indication qui est définie comme nécessitant une réponse;
- primitive de confirmation: émise par un fournisseur GCC comme résultat d'une primitive de réponse.

Les PDU se divisent en trois catégories. Les noms de PDU contiennent tous les symboles Request, Indication ou Response (demande, indication ou réponse) indiquant l'utilisation prévue pour la PDU. Elles sont définies comme suit:

- PDU de demande: ces unités exigent en retour une PDU de réponse. Si la demande est faite pour une fonction non prise en charge par le nœud récepteur, par exemple une PDU optionnelle ou non normalisée, une unité de réponse PDU générique *FunctionNotSupportedResponse* (réponse de fonction non prise en charge) sera utilisée pour fournir une réponse au demandeur, c'est-à-dire envoyée au canal correspondant à l'identificateur du nœud demandeur.
- PDU d'indication: ces unités ne nécessitent pas de réponse. Ce sont, par exemple, celles qui ont un but d'information.
- PDU de réponse: ces unités fournissent des réponses à des unités de demande PDU particulières, envoyées sur le canal correspondant à l'identificateur du nœud demandeur.

## 6 Aperçu général

Dans le contexte du service audiovisuel de conférence de l'UIT-T, une conférence fait référence à un groupe de nœuds dispersés géographiquement qui sont reliés entre eux et peuvent échanger des

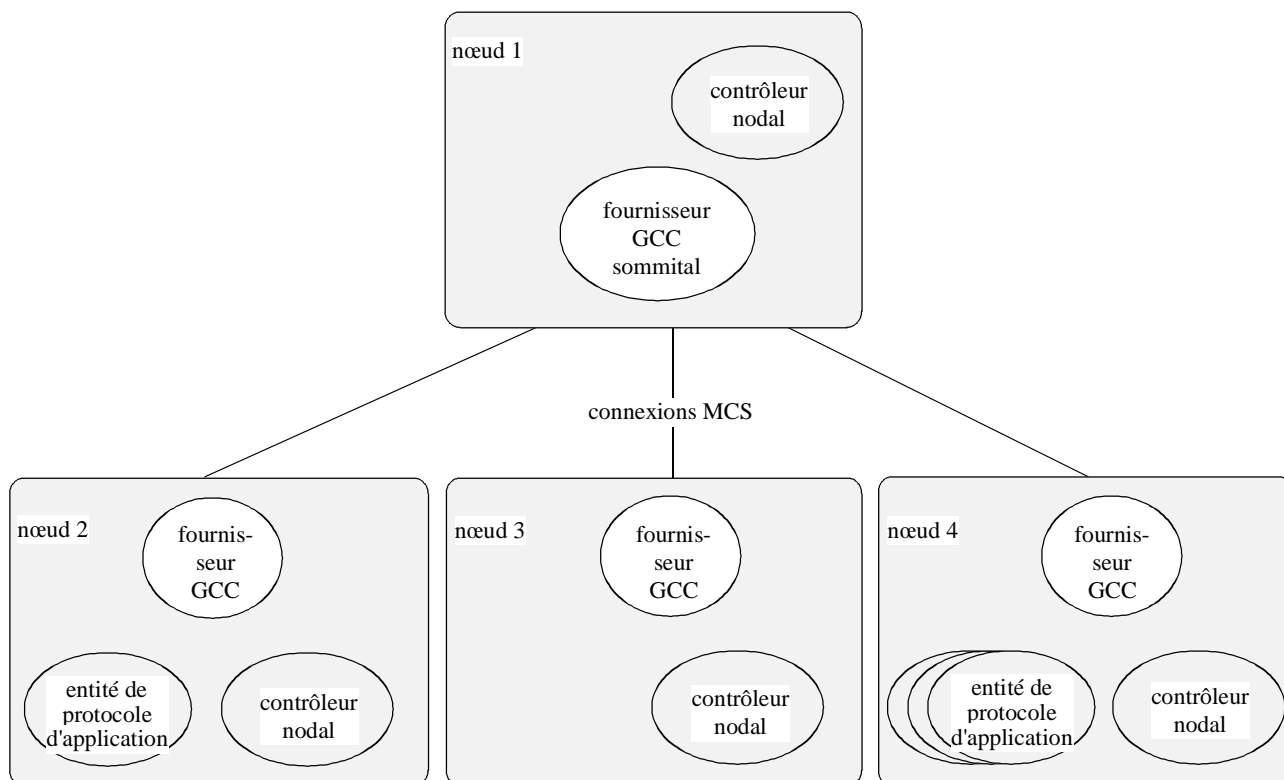
informations audiographiques et audiovisuelles à travers des réseaux de communication divers. Les participants à une conférence peuvent avoir accès à divers types de capacités de maniement de média tels que les moyens audio uniquement (téléphonie), les moyens audio et données (audiographiques), les moyens audio et vidéo (audiovisuels) et les moyens audio, vidéo et données (multimédia).

Les Recommandations des séries F, G, H et T fournissent un cadre général pour l'interfonctionnement de terminaux audio, vidéo et graphiques en mode point à point à travers des réseaux de télécommunication existants. Elles fournissent également la capacité d'interconnexion de trois terminaux ou plus au sein d'une même conférence à l'aide d'une MCU.

La présente Recommandation fournit un cadre général à haut niveau pour la gestion et la commande de conférences de terminaux audiographiques, audiovisuels et de MCU. Elle coexiste avec les Recommandations associées [UIT-T T.122] et [UIT-T T.125] (MCS) et [UIT-T T.123] (AVPS) afin de fournir un mécanisme pour l'établissement et le pilotage des conférences. Les Recommandations [UIT-T T.122], [UIT-T T.123], [UIT-T T.124] et [UIT-T T.125] constituent un ensemble minimal de Recommandations permettant de réaliser un terminal ou une MCU de fonctionnalité complète.

La présente Recommandation comprend les composants fonctionnels de la commande générique de conférence (GCC) suivants: établissement et fin de la conférence, gestion du répertoire de conférence, gestion du répertoire d'application, fourniture des services de référentiel d'application et de présidence de la conférence. Les définitions de services pour les primitives associées à ces composants fonctionnels sont données au paragraphe 7. Les définitions de protocoles correspondantes sont données au paragraphe 8.

La Figure 6-1 présente un exemple de répartition des composants du service GCC à l'intérieur d'un domaine MCS. Les composants du service GCC sont en blanc. Tout terminal ou MCU contient un fournisseur GCC fournissant les services GCC pour le contrôleur nodal local et les entités de protocole d'application locales.



T0819410-94

**Figure 6-1 – Exemple de composants GCC distribués au sein d'un domaine MCS**

## 6.1 Modèle systémique de nœud de conférence

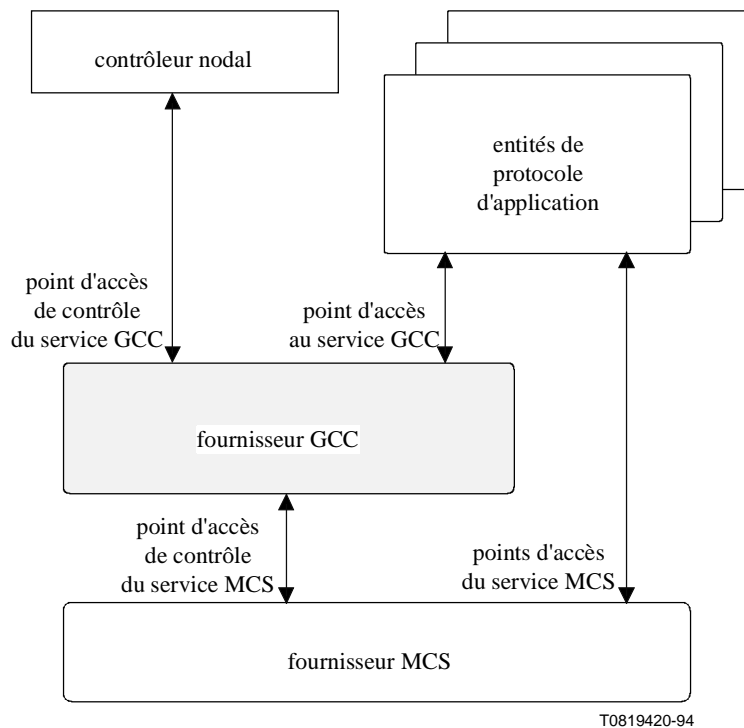
Tout nœud participant à une conférence GCC est constitué d'une couche MCS, d'une couche GCC, d'un contrôleur nodal et peut également comporter une ou plusieurs entités de protocole d'application. La Figure 6-2 illustre les relations entre ces composants au sein d'un nœud donné. Le contrôleur nodal est l'entité de commande du nœud qui traite des aspects de la conférence s'appliquant à l'ensemble du nœud. Le contrôleur nodal interagit avec le service GCC, mais n'interagit pas directement avec le service MCS. Les entités de protocole d'application interagissent directement avec le service GCC et le service. Les services fournis aux entités de protocole d'application par le service GCC consistent en premier lieu à permettre aux entités de protocole d'application homologues de communiquer directement via le service MCS. Il peut exister une communication locale entre entités de protocole d'application ou entre des entités de protocole d'application et le contrôleur nodal, mais ceci est un sujet de mise en œuvre locale non couvert par la présente Recommandation. Il peut y avoir au sein d'un nœud plusieurs entités de protocole d'application basées sur le même protocole d'application. Elles peuvent dans ce cas, soit faire partie de la même session de protocole d'application, ce qui les met en mesure de communiquer à l'intérieur du nœud aussi bien qu'avec d'autres entités de protocole d'application homologues, soit faire partie de sessions distinctes de protocole d'application, ce qui leur permet de communiquer séparément avec leurs homologues tout en utilisant le même protocole.

Les primitives de service décrites au paragraphe 7 concernent le point d'accès au service GCC et le point d'accès de contrôle au service GCC indiqués dans la Figure 6-2. Les PDU décrites au paragraphe 8 sont transmises au moyen de primitives de service MCS disponibles au point d'accès de contrôle MCS (*MCSAP*, *MCS access point*).

NOTE 1 – L'intention normative de la présente Recommandation est de spécifier les procédures et les contenus de la communication externe, c'est-à-dire les séquences d'exécution de primitives et d'échanges de données agissant, à des fins de commande de la conférence, à travers le point de contrôle *MCSAP*. La structure interne d'un nœud suggérée par la Figure 6-2 présente l'objet des fonctions du protocole GCC mais n'a pas de caractère normatif. Les primitives du service GCC dont l'action est strictement locale n'ont pas besoin d'exister dans tous les nœuds sous la forme où elles sont présentées ici. Les déclarations concernant ce qu'un contrôleur nodal ou une entité de protocole d'application fait dans certaines circonstances doivent être interprétées librement si les mêmes résultats de communication externe peuvent être obtenus par des mécanismes internes différents.

NOTE 2 – Le modèle systémique suppose que les demandes de service peuvent être émises à tout instant. Un contrôleur nodal ou une entité de protocole d'application n'a pas besoin d'attendre une confirmation du fournisseur GCC concernant la demande précédente avant d'émettre une nouvelle demande. Une réalisation particulière peut toutefois imposer des exigences plus strictes.





**Figure 6-2 – Modèle systémique indiquant le point d'accès au service GCC et la relation avec le service MCS**

## 6.2 Etablissement et fin de la conférence

Le service GCC fournit un ensemble de services pour l'établissement et la fin de conférences. Une conférence peut être envisagée comme une salle de réunion dans laquelle un nombre quelconque de participants peuvent se réunir afin d'échanger des informations audiographiques et audiovisuelles. A l'instar de salles de réunion physiques, des services tels que la recherche des conférences en cours, l'entrée dans une conférence, le départ d'une conférence, la restriction à l'accès d'une conférence ont également un sens dans le cas de conférences audiographiques et audiovisuelles.

Avant l'entrée dans une conférence, les participants d'un nœud peuvent ne pas connaître toute l'information nécessaire pour y entrer. Le service GCC fournit aux participants un moyen de consulter un répertoire de conférence et de sélectionner la conférence dans laquelle ils souhaitent entrer. Ce service est comparable à un programme de conférence, généralement affiché dans un hall d'entrée, permettant de trouver une salle de réunion dans laquelle se tient une réunion donnée.

Le service GCC fournit un moyen pour la création de nouvelles conférences. La création peut être faite par un participant de la conférence ou par un administrateur de conférence. Quand une nouvelle conférence est créée, ses caractéristiques, appelées profil de conférence, sont spécifiées par son créateur. Le profil de la conférence contient des informations telles que le nom de la conférence, l'indication si son accès est limité au moyen d'un mot de passe et s'il est possible d'y entrer librement ou uniquement sur invitation – selon que la conférence est déverrouillée ou verrouillée.

L'extension d'une conférence existante peut être déclenchée par le nœud entrant, par un organisateur de conférence (ou par un nœud désigné par l'organisateur). Si une conférence est protégée par un mot de passe, un nœud entrant a l'obligation de fournir le mot de passe correct pour être autorisé à y entrer. Si une conférence est verrouillée, l'entrée n'est possible que sur demande de l'organisateur ou d'un nœud désigné par l'organisateur. Les nœuds peuvent entrer ou être ajoutés pendant toute la durée d'une conférence. Il est possible d'entrer en même temps dans plus d'une conférence. Le service GCC fournit également un moyen de transférer des participants d'une conférence à une

autre. Cette fonction peut être utilisée pour fusionner deux conférences ou pour scinder une conférence en plusieurs parties.

Un nœud peut se déconnecter à tout instant en laissant les autres nœuds poursuivre la conférence. Selon la méthode de fin choisie, une conférence peut se terminer automatiquement lorsque tous les nœuds se sont déconnectés ou être terminée manuellement par une demande explicite de fin de conférence. L'organisateur ou un nœud désigné par l'organisateur peut également forcer explicitement la fin de l'ensemble de la conférence ou en éjecter un nœud donné.

### **6.3 Le répertoire de conférence**

Lorsqu'un nœud a fait son entrée dans une conférence, il annonce sa présence à cette conférence. Le service GCC fournit une primitive permettant à tout nœud d'une conférence d'annoncer sa présence et une autre permettant à tout nœud d'obtenir soit des mises à jour du répertoire de conférence soit la totalité du répertoire de conférence. Celui-ci est une liste des nœuds conventionnels et éventuellement comptés de la conférence. Le répertoire comprend, pour chaque nœud, des informations telles que le nom du nœud, une liste des participants de ce nœud ainsi que d'autres informations nécessaires à une communication correcte entre nœuds. Un nœud conventionnel ou compté n'est pas considéré comme faisant partie de la conférence tant qu'il ne figure pas dans le répertoire de conférence.

### **6.4 Le répertoire d'application**

Le service GCC fournit un moyen d'identifier quelles sont les entités de protocole d'application disponibles sur chaque nœud et un moyen permettant de fournir l'information nécessaire à la communication entre entités de protocole d'application homologues. Lors de son entrée dans la conférence, chaque nœud conventionnel ou compté envoie à tous les autres nœuds sa liste locale d'entités de protocole d'application – appelée *répertoire d'application local* – qu'il pourra mettre à jour à tout instant par la suite. Le répertoire d'application de la conférence est constitué à partir de cette information et diffusé à tous les nœuds intéressés. Les parties pertinentes de ce répertoire sont ensuite communiquées localement à chaque entité de protocole d'application et au contrôleur nodal. En plus d'un simple répertoire, le service GCC fournit également aux entités de protocole d'application des nœuds conventionnels un service permettant d'inclure dans l'information échangée une liste de capacités spécifiques du protocole d'application. Le service GCC applique à cette information issue de tous les nœuds un ensemble de règles fixées afin de déterminer un ensemble commun de capacités d'application. Cette information est également communiquée localement à chaque entité de protocole d'application.

### **6.5 Le référentiel d'application**

Le référentiel d'application est une base de données active résidant sur le fournisseur GCC sommital et pouvant être utilisée pour gérer les canaux, les jetons et d'autres ressources partagées utilisées dans une conférence. Le référentiel d'application peut fournir une aide lors de l'établissement d'une communication entre entités de protocole d'application homologues.

### **6.6 La présidence de la conférence**

Le service GCC fournit une méthode permettant à un nœud de devenir le président d'une conférence. Le service GCC utilise un jeton pour définir si une conférence est présidée ou non. Le nœud qui s'empare du jeton de président devient le président de la conférence. Un nœud peut également demander la présidence ou accepter la présidence transmise par le président en exercice. Le service GCC fournit à la demande l'identité du président de conférence en exercice. Il est possible d'indiquer lors de la création d'une conférence que le mode présidé n'est pas autorisé pour la durée de la conférence.

Le mode présidé est disponible pour assurer l'ordre dans le déroulement d'une conférence. Le moyen effectif par lequel cet ordre est assuré est déterminé par les protocoles d'application. Les Recommandations spécifiant des protocoles d'application peuvent en particulier définir des procédures différentes selon que la conférence est présidée ou non. Le service GCC fournit un procédé pour une exploitation de base en mode présidé utilisable par des protocoles d'application. Ce procédé permet à un nœud de demander une autorisation au nœud président. Si l'autorisation est accordée, toutes les entités de protocole d'application de ce nœud peuvent agir en accord avec la spécification du protocole d'application. Par exemple, les protocoles d'application peuvent spécifier des limitations strictes pour les opérations permises aux nœuds qui n'ont pas d'autorisation du président, alors que les limitations sont levées en partie ou en totalité pour les nœuds qui possèdent cette autorisation. Les protocoles d'application peuvent, par exemple, spécifier également qu'une fois qu'une autorisation générale a été accordée par l'entité de protocole d'application homologue, des autorisations supplémentaires doivent être accordées par l'entité de protocole d'application homologue du nœud président, si elle existe, avant que l'exécution de certaines opérations soit permise.

## **6.7 Fonctions diverses**

Une méthode est fournie pour la coordination de conférences limitées dans le temps. Des procédés sont fournis pour permettre à un nœud de trouver le temps restant dans une conférence limitée dans le temps, d'annoncer à tous les nœuds la quantité de temps restante (utilisé normalement pour annoncer que le temps est pratiquement écoulé) et de demander l'addition de temps supplémentaire.

Une méthode est également fournie pour demander l'assistance d'une opératrice non spécifiée. Une autre méthode est aussi prévue pour permettre la transmission de messages textuels simples.

## **6.8 Conférences échelonnables**

Le protocole T.124 est conçu pour permettre un certain nombre de niveaux différents d'échelonnabilité, y compris de petites conférences "à couplages serrés", des conférences "acquittées" de taille moyenne et de très grandes conférences "à couplages lâches". Ces échelons sont obtenus au moyen de trois catégories nodales, dont chacune offre un degré variable d'échelonnabilité. Ces catégories, dont un aperçu général est donné ci-dessous, sont les suivantes: "nœuds conventionnels", "nœuds comptés" et "nœuds anonymes".

Les nœuds conventionnels fournissent à chaque nœud d'une conférence une grande quantité d'informations nodales. Ce sont les nœuds de "poids fort" en termes d'incidence sur les autres membres de la conférence. L'utilisation de cette catégorie convient aux principaux membres d'une conférence à couplages lâches et à tous les nœuds participant à une petite conférence à couplages serrés. Les nœuds conventionnels sont chargés de créer toutes les sessions d'entités APE à l'intérieur d'une conférence et indiquent les capacités négociées dans ces sessions. Chaque conférence T.124 doit comporter au moins un nœud conventionnel. De même, chaque session d'entité APE doit, pendant toute sa durée, comporter au moins un nœud conventionnel. Une fois que tous les nœuds conventionnels ont quitté une session d'entité APE, cette session cesse d'exister.

Les nœuds comptés sont de poids moins fort que les nœuds conventionnels. Les nœuds de ce type n'apparaissent que dans les répertoires reçus par des nœuds conventionnels. Les nœuds comptés ne créent pas de sessions d'entité APE et n'ont pas d'incidence sur les capacités de session. Les capacités pour chaque session d'entité APE sont établies par les nœuds conventionnels participant à la conférence. Les nœuds comptés sont normalement utilisés dans les situations où il est nécessaire d'accuser réception de données extraites par un nœud conventionnel. Les conférences à nœuds comptés participants s'échelonnent mieux que les conférences ne comportant que des nœuds conventionnels mais elles restent limitées en taille en raison de l'échange d'informations de répertoire. Les nœuds comptés peuvent extraire un contenu pour la conférence mais celui-ci ne doit jamais être acquitté.

Des trois catégories, celle des nœuds anonymes est celle qui a le poids le plus faible. Les nœuds de cette catégorie n'ont pas d'incidence sur les répertoires de conférence et d'application, ne peuvent pas créer de sessions d'entité APE et n'ont pas d'incidence sur les capacités de conférence. Leur objet est de permettre des conférences à très grande échelle dans des environnements où il existe un petit nombre de présentateurs et un grand nombre d'observateurs. Un scénario typique de conférence pourrait comprendre un petit nombre de nœuds conventionnels, qui seraient les principaux contributeurs à la conférence et un grand nombre de nœuds anonymes qui seraient observateurs.

Avant l'introduction des catégories nodales, le nombre de nœuds pouvant participer à une conférence à commande GCC était sévèrement limité. Les catégories nodales permettent de prendre en charge un certain nombre de modèles d'échelle de conférence différents. Etant donné que les nœuds successoraux qui ne sont pas informés des catégories nodales doivent continuer à être pris en charge dans le cadre de la Recommandation T.120, quelques restrictions sont nécessaires pour assurer la rétrocompatibilité. Par ailleurs, une caractéristique de la commande GCC qui facilite les conférences échelonnables est l'utilisation de mises à jour de répertoire delta par le fournisseur sommital, lorsque celui-ci informe les participants à une conférence du fait qu'un nœud a joint, laissé ou modifié un répertoire enregistré. Cette caractéristique n'était pas non plus prise en charge dans la version originale de la commande GCC. Les règles suivantes doivent être respectées pour assurer la rétrocompatibilité avec les nœuds successoraux:

- un nœud ne peut être considéré comme anonyme ou compté que s'il se joint à la conférence par l'intermédiaire d'un nœud qui est informé des catégories nodales. De même, tous les nœuds issus du nœud se joignant au fournisseur sommital (compris) doivent aussi être informés des catégories nodales;
- les nœuds qui ne sont pas informés des catégories nodales ou qui ne signalent pas leur compréhension des catégories nodales lorsqu'ils se joignent à une conférence sont toujours traités comme étant conventionnels;
- le fournisseur sommital, l'organisateur et tous les nœuds de gestion sont toujours rangés dans la catégorie des nœuds conventionnels;
- tous les nœuds T.124, à l'exception de ceux qui ne participeront à une conférence T.120 qu'à titre de nœuds terminaux anonymes ou comptés, doivent continuer à prendre en charge le canal de diffusion GCC de nœud successoral, ainsi que le canal de diffusion GCC de nœud conventionnel et, si nécessaire, le canal de diffusion GCC de nœud compté (voir l'explication de tous les canaux de diffusion dans la définition du protocole donnée dans la présente Recommandation);
- seuls les nœuds conventionnels sont autorisés à ajouter, à supprimer ou à modifier les paramètres dans le référentiel. Pour prendre en charge des nœuds comptés ou anonymes, le référentiel peut être lu par tout nœud qui en fait la demande, y compris les nœuds qui ne sont pas énumérés dans les répertoires de conférence et d'application;
- il n'existe pas de restrictions topologiques quant aux catégories nodales qui peuvent être connectées à d'autres catégories nodales, à condition que tous les nœuds supérieurs à un nœud envisageant d'utiliser le protocole associé aux catégories nodales soient informés de ces catégories.

En résumé, lorsqu'une conférence est créée, on spécifie un mode de conférence qui définit les catégories nodales qui seront autorisées à participer à la conférence. Il appartient à la fois au fournisseur de commande GCC et au contrôleur nodal situé chez le fournisseur sommital de déterminer les catégories dans lesquelles un nœud arrivant doit s'inscrire. Ces modes de conférence sont les suivants: *conventionnels-seulement*, qui n'autorise que des nœuds conventionnels à participer à la conférence; *comptés-contrôlés*, qui n'autorise que les nœuds comptés et contrôlés à participer à la conférence; *anonymes-contrôlés*, qui n'autorise que les nœuds comptés et conventionnels à participer à la conférence; et le *mode sans restriction*, qui autorise les trois

catégories nodales à participer à la conférence. Le contrôleur nodal doit utiliser le mot de passe T.124 pour déterminer les nœuds qui peuvent participer à titre conventionnel lorsque le mode de conférence est soit *compté-contrôlé* soit *anonyme-contrôlé*.

## 6.9 Résumé des services abstraits GCC

Le Tableau 6-1 donne une liste de toutes les primitives GCC et des PDU associées. Le tableau indique également si la primitive est obligatoire (M) ou exigée sous condition (C) ou optionnelle (O) pour un terminal ou une MCU. Pour un terminal multiaccès et pour toute primitive, l'exigence prise en considération pour cette primitive sera la plus restrictive de celle qui concerne le terminal ou la MCU. Une primitive est exigée sous condition si son utilisation est spécifiée comme obligatoire par le protocole d'application d'une ou de plusieurs entités de protocole d'application situées sur le nœud. Le tableau mentionne également si les PDU correspondantes sont obligatoires (M) ou exigées sous condition (C) dans les directions d'émission (T) et de réception (R). Pour un terminal multiaccès, l'exigence prise en considération sera de nouveau celle du terminal ou de la MCU qui est la plus restrictive. Une PDU est exigée sous condition si elle n'est exigée que si la primitive correspondante est fournie par le nœud. Dans le cas où une primitive n'est pas obligatoire mais où la PDU est obligatoire, ceci implique qu'il existe une certaine partie du protocole, sans relation avec la primitive, qui utilise cette PDU et exige qu'elle soit prise en charge.

**Tableau 6-1 – Primitives GCC et PDU**

Unité fonctionnelle	Primitives	Terme	MCU	PDU associées	Direction	Terme	MCU
Etablissement et fin de conférence	demande GCC-Conference-Create	M	O	ConferenceCreateRequest	T	M	C
	indication GCC-Conference-Create	M	M	ConferenceCreateRequest	R	M	M
	réponse GCC-Conference-Create	M	M	ConferenceCreateResponse	T	M	M
	confirmation GCC-Conference-Create	M	O	ConferenceCreateResponse UserIDIndication	R T, R	M M	C M
	demande GCC-Conference-Query	M	M	ConferenceQueryRequest	T	M	M
	indication GCC-Conference-Query	M	M	ConferenceQueryRequest	R	M	M
	réponse GCC-Conference-Query	M	M	ConferenceQueryResponse	T	M	M
	confirmation GCC-Conference-Query	M	M	ConferenceQueryResponse	R	M	M
	demande GCC-Conference-Join	M	O	ConferenceJoinRequest	T	M	C
	indication GCC-Conference-Join	O	M	ConferenceJoinRequest	R	C	M
	réponse GCC-Conference-Join	O	M	ConferenceJoinResponse	T	C	M
	confirmation GCC-Conference-Join	M	O	ConferenceJoinResponse UserIDIndication	R T, R	M M	C M
	demande GCC-Conference-Invite	O	M	ConferenceInviteRequest	T	C	M
	indication GCC-Conference-Invite	M	M	ConferenceInviteRequest	R	M	M
	réponse GCC-Conference-Invite	M	M	ConferenceInviteResponse	T	M	M
	confirmation GCC-Conference-Invite	O	M	ConferenceInviteResponse UserIDIndication	R T, R	C M	M M
	demande GCC-Conference-Add	O	O	ConferenceAddRequest	T	C	C
	indication GCC-Conference-Add	O	O	ConferenceAddRequest	R	C	C
	réponse GCC-Conference-Add	O	O	ConferenceAddResponse	T	C	C
	confirmation GCC-Conference-Add	O	O	ConferenceAddResponse	R	C	C
	demande GCC-Conference-Lock	O	O	ConferenceLockRequest	T	C	C
	indication GCC-Conference-Lock	O	O	ConferenceLockRequest	R	C	C
	réponse GCC-Conference-Lock	O	O	ConferenceLockResponse	T	C	C
	confirmation GCC-Conference-Lock	O	O	ConferenceLockResponse	R	C	C
	demande GCC-Conference-Unlock	O	O	ConferenceUnlockRequest	T	C	C
	indication GCC-Conference-Unlock	O	O	ConferenceUnlockRequest	R	C	C
	réponse GCC-Conference-Unlock	O	O	ConferenceUnlockResponse	T	C	C

**Tableau 6-1 – Primitives GCC et PDU**

Unité fonctionnelle	Primitives	Terme	MCU	PDU associées	Direction	Terme	MCU	
	confirmation GCC-Conference-Unlock	O	O	ConferenceUnlockResponse	R	C	C	
	indication GCC-Conference-Lock-Report	O	O	ConferenceLockIndication ConferenceUnlockIndication	T, R T, R	C C	C C	
	demande GCC-Conference-Disconnect	M	M	–	–	–	–	
	indication GCC-Conference-Disconnect	M	M	–	–	–	–	
	confirmation GCC-Conference-Disconnect	M	M	–	–	–	–	
	demande GCC-Conference-Terminate	O	O	ConferenceTerminateRequest	T	C	C	
	indication GCC-Conference-Terminate		M	M	ConferenceTerminateRequest	R	M	M
					ConferenceTerminateIndication	T	M	M
	confirmation GCC-Conference-Terminate		O	O	ConferenceTerminateIndication	R	M	M
					ConferenceTerminateResponse	T	M	M
	confirmation GCC-Conference-Terminate		O	O	ConferenceTerminateResponse	R	C	C
					ConferenceTerminateResponse	R	C	C
	demande GCC-Conference-Eject-User	O	O	ConferenceEjectUserRequest	T	C	C	
	indication GCC-Conference-Eject-User		M	M	ConferenceEjectUserRequest	R	M	M
					ConferenceEjectUserIndication	T	M	M
	confirmation GCC-Conference-Eject-User		O	O	ConferenceEjectUserIndication	R	M	M
					ConferenceEjectUserResponse	T	M	M
	confirmation GCC-Conference-Eject-User		O	O	ConferenceEjectUserResponse	R	C	C
					ConferenceEjectUserResponse	R	C	C
	demande GCC-Conference-Transfer	O	M	ConferenceTransferRequest	T	C	M	
indication GCC-Conference-Transfer		M	M	ConferenceTransferRequest	R	M	M	
				ConferenceTransferIndication	T	M	M	
confirmation GCC-Conference-Transfer		O	M	ConferenceTransferIndication	R	M	M	
				ConferenceTransferResponse	T	M	M	
confirmation GCC-Conference-Transfer		O	M	ConferenceTransferResponse	R	C	M	
				ConferenceTransferResponse	R	C	M	
Répertoire de conférence	demande GCC-Conference-Announce-Presence	M	M	RosterUpdateIndication	T, R	M	M	
	confirmation GCC-Conference-Announce-Presence	M	M	–	–	–	–	
	indication GCC-Conference-Roster-Report	M	M	RosterUpdateIndication	T, R	M	M	
	demande GCC-Conference-Roster-Inquire	O	O	–	–	–	–	
Répertoire d'application	confirmation GCC-Conference-Roster-Inquire	O	O	–	–	–	–	
	indication GCC-Application-Permission-To-Enroll	M	C	–	–	–	–	
	demande GCC-Application-Enroll	M	C	RosterUpdateIndication	T, R	M	M	
				RosterRefreshRequest	T, R	C	C	
	confirmation GCC-Application-Enroll	M	C	–	–	–	–	
	indication GCC-Application-Roster-Report	M	C	RosterUpdateIndication	T, R	M	M	
	demande GCC-Application-Roster-Inquire	O	O	–	–	–	–	
	confirmation GCC-Application-Roster-Inquire	O	O	–	–	–	–	
demande GCC-Application-Invoke	O	O	ApplicationInvokeIndication	T	C	C		
indication GCC-Application-Invoke	O	O	ApplicationInvokeIndication	R	C	C		
			–	–	–	–		
Référentiel d'application	demande GCC-Registry-Register-Channel	C	C	RegistryRegisterChannelRequest	T	C	C	
				RegistryRegisterChannelRequest	R	M	M	
	confirmation GCC-Registry-Register-Channel	C	C	RegistryResponse	T	M	M	
				RegistryResponse	R	C	C	
	demande GCC-Registry-Assign-Token	C	C	RegistryAssignTokenRequest	T	C	C	
confirmation GCC-Registry-Assign-Token	C	C	RegistryAssignTokenRequest	R	M	M		
			RegistryResponse	T	M	M		
confirmation GCC-Registry-Assign-Token	C	C	RegistryResponse	R	C	C		
demande GCC-Registry-Set-Parameter	C	C	RegistrySetParameterRequest	T	C	C		

**Tableau 6-1 – Primitives GCC et PDU**

Unité fonctionnelle	Primitives	Terme	MCU	PDU associées	Direction	Terme	MCU
	confirmation GCC-Registry-Set-Parameter	C	C	RegistrySetParameterRequest RegistryResponse RegistryResponse	R T R	M M C	M M C
	demande GCC-Registry-Retrieve-Entry	C	C	RegistryRetrieveEntryRequest	T	C	C
	confirmation GCC-Registry-Retrieve-Entry	C	C	RegistryRetrieveEntryRequest RegistryResponse RegistryResponse	R T R	M M C	M M C
	demande GCC-Registry-Delete-Entry	C	C	RegistryDeleteEntryRequest	T	C	C
	confirmation GCC-Registry-Delete-Entry	C	C	RegistryDeleteEntryRequest RegistryResponse RegistryResponse	R T R	M M C	M M C
	demande GCC-Registry-Monitor	C	C	RegistryMonitorEntryRequest	T	C	C
	indication GCC-Registry-Monitor	C	C	RegistryMonitorEntryRequest RegistryMonitorEntryIndication RegistryMonitorEntryIndication	R T R	M M C	M M C
	confirmation GCC-Registry-Monitor	C	C	RegistryResponse RegistryResponse	T R	M C	M C
	demande GCC-Registry-Allocate-Handle	C	C	RegistryAllocateHandleRequest	T	C	C
	confirmation GCC-Registry-Allocate-Handle	C	C	RegistryAllocateHandleRequest RegistryAllocateHandleResponse RegistryAllocateHandleResponse	R T R	M M C	M M C
Conduite de conférence	demande GCC-Conductor-Assign	O	O	-	-	-	-
	indication GCC-Conductor-Assign	C	C	ConductorAssignIndication	T	M (Note)	M (Note)
	confirmation GCC-Conductor-Assign	O	O	ConductorAssignIndication -	R -	C -	C -
	demande GCC-Conductor-Release	O	O	ConductorReleaseIndication	T	C	C
	indication GCC-Conductor-Release	C	C	ConductorReleaseIndication ConductorReleaseIndication	R T	M (Note) M (Note)	M (Note) M (Note)
	confirmation GCC-Conductor-Release	O	O	-	-	-	-
	demande GCC-Conductor-Please	O	O	-	-	-	-
	indication GCC-Conductor-Please	O	O	-	-	-	-
	confirmation GCC-Conductor-Please	O	O	-	-	-	-
	demande GCC-Conductor-Give	O	O	-	-	-	-
	indication GCC-Conductor-Give	O	O	-	-	-	-
	réponse GCC-Conductor-Give	O	O	ConductorAssignIndication	T	C	C
	confirmation GCC-Conductor-Give	O	O	-	-	-	-
	demande GCC-Conductor-Inquire	C	C	-	-	-	-
	confirmation GCC-Conductor-Inquire	C	C	-	-	-	-
	demande GCC-Conductor-Permission-Ask	O	O	ConductorPermissionAskIndication	T	C	C
	indication GCC-Conductor-Permission-Ask	O	O	ConductorPermissionAskIndication	R	C	C
confirmation GCC-Conductor-Permission-Ask	O	O	-	-	-	-	
demande GCC-Conductor-Permission-Grant	O	O	ConductorPermissionGrantIndication	T	C	C	
indication GCC-Conductor-Permission-Grant	O	O	ConductorPermissionGrantIndication	R	C	C	
confirmation GCC-Conductor-Permission-Grant	O	O	-	-	-	-	
Divers	demande GCC-Conference-Time-Remaining	O	O	ConferenceTimeRemainingIndication	T	C	C
	indication GCC-Conference-Time-Remaining	O	O	ConferenceTimeRemainingIndication	R	C	C

**Tableau 6-1 – Primitives GCC et PDU**

Unité fonctionnelle	Primitives	Terme	MCU	PDU associées	Direction	Terme	MCU	
	confirmation GCC-Conference-Time-Remaining	O	O	–	–			
	demande GCC-Conference-Time-Inquire	O	O	ConferenceTimeInquireIndication	T	C	C	
	indication GCC-Conference-Time-Inquire	O	O	ConferenceTimeInquireIndication	R	C	C	
	confirmation GCC-Conference-Time-Inquire	O	O	–	–			
	demande GCC-Conference-Extend	O	O	ConferenceTimeExtendIndication	T	C	C	
	indication GCC-Conference-Extend	O	O	ConferenceTimeExtendIndication	R	C	C	
	confirmation GCC-Conference-Extend	O	O	–	–			
	demande GCC-Conference-Assistance	O	O	ConferenceAssistanceIndication	T	C	C	
	indication GCC-Conference-Assistance	O	O	ConferenceAssistanceIndication	R	C	C	
	confirmation GCC-Conference-Assistance	O	O	–	–			
	demande GCC-Text-Message	O	O	TextMessageIndication	T	C	C	
	indication GCC-Text-Message	O	O	TextMessageIndication	R	C	C	
	confirmation GCC-Text-Message	O	O	–	–			
	–	–	–	–	FunctionNotSupported	T, R	M	M

NOTE – Le fournisseur GCC sommital doit obligatoirement pouvoir recevoir les unités PDU ConductorReleaseIndication afin de traiter correctement les requêtes nécessitant des privilèges. Ces privilèges dépendent du mode présidé ou non présidé de la conférence. Le fournisseur GCC sommital doit obligatoirement pouvoir émettre des unités PDU ConductorAssignIndication et ConductorReleaseIndication pour pouvoir traiter correctement le rattachement de nouveaux nœuds à la conférence.

## 7 Définition du service GCC

### 7.1 Etablissement et fin de conférence

Le présent paragraphe décrit les primitives nécessaires à l'établissement et à l'achèvement des conférences. Il est prévu que ces primitives sont réservées à l'usage du seul contrôleur nodal d'un terminal ou d'une MCU.

#### 7.1.1 Le profil de conférence

Les conférences possèdent les caractéristiques suivantes définies lors de leur création et communiquées à tout nœud lors de son entrée dans la conférence. Ces caractéristiques ne changent pas pendant toute la conférence. Cette information est désignée dans son ensemble comme profil de conférence:

- *nom de conférence* – Chaîne numérique associée à une chaîne de texte unicode rangée 00 optionnelle identifiant la conférence. Si les deux formes du nom de conférence sont utilisées au moment de la création de la conférence, l'une ou l'autre peut être utilisée pour désigner la conférence en question;
- *description de conférence* – Chaîne de caractères optionnelle décrivant la conférence. Si une conférence est listée, cette chaîne figure dans la liste de descripteurs de conférence de la réponse à une primitive de demande GCC-Conference-Query (interrogation de conférence);
- *protégée par mot de passe ou non protégée par mot de passe* – Sélection de la protection de la conférence par mot de passe;
- *listée ou non listée* – Définit si une conférence figure ou non dans la liste de conférences obtenue par la primitive de demande GCC-Conference-Query;
- *pouvant être présidée ou non* – Définit si la conférence peut être placée en mode présidé ou si la conférence est en permanence en mode non présidé;



- *méthode de fin* – Définit si une conférence doit se prolonger jusqu'à ce qu'elle soit terminée explicitement (cas de fin manuelle), ou si elle doit durer jusqu'à ce que tous les participants se soient déconnectés (cas de fin automatique);
- *listes de privilèges* – Ensemble de listes indiquant quels sont les privilèges normalement réservés à l'organisateur qui sont également disponibles pour le président, pour un nœud quelconque d'une conférence présidée ou pour un nœud quelconque d'une conférence non présidée;
- *mode de conférence* – Option qui spécifie les catégories nodales qui seront autorisées à participer à la conférence après la création de celle-ci. Si le mode de conférence n'est pas spécifié, sa valeur par défaut sera *conventionnels-seulement*.

### 7.1.2 Description des services abstraits

La liste qui suit récapitule les primitives définies dans le présent sous-paragraphe et en résume les fonctions:

- GCC-Conference-Create (*création d'une conférence*) – Utilisée par le contrôleur nodal pour créer une nouvelle conférence et en spécifier les caractéristiques.
- GCC-Conference-Query (*interrogation de conférence*) – Utilisée par le contrôleur nodal pour demander quelles sont les conférences en cours et quelle est l'information nécessaire pour tenter d'y entrer.
- GCC-Conference-Join (*entrée dans une conférence*) – Utilisée par le contrôleur nodal pour entrer dans une conférence.
- GCC-Conference-Invite (*invitation à une conférence*) – Utilisée par le contrôleur nodal pour inviter un nœud à une conférence existante.
- GCC-Conference-Add (*addition à une conférence*) – Autorisée uniquement pour l'organisateur de la conférence ou pour un nœud désigné par l'organisateur, cette primitive est utilisée par le contrôleur nodal pour demander qu'un nœud soit ajouté à la conférence par une numérotation de départ à partir d'une MCU.
- GCC-Conference-Lock (*verrouillage de conférence*) – Autorisée uniquement pour l'organisateur de la conférence ou pour un nœud désigné par l'organisateur, cette primitive est utilisée par le contrôleur nodal pour interdire à de nouveaux participants d'entrer dans une conférence sans y être ajoutés d'une manière explicite.
- GCC-Conference-Unlock (*déverrouillage de conférence*) – Autorisée uniquement pour l'organisateur de la conférence ou pour un nœud désigné par l'organisateur, cette primitive est utilisée par le contrôleur nodal pour permettre à de nouveaux participants d'entrer dans une conférence.
- GCC-Conference-Lock-Report (*compte rendu de verrouillage de conférence*) – Fournit au contrôleur nodal l'indication que la conférence est passée du mode verrouillé au mode non verrouillé ou vice versa.
- GCC-Conference-Disconnect (*déconnexion de conférence*) – Utilisée par le contrôleur nodal pour déconnecter le nœud local d'une conférence en cours.
- GCC-Conference-Terminate (*fin de conférence*) – Autorisée uniquement pour l'organisateur de la conférence ou pour un nœud désigné par l'organisateur, cette primitive est utilisée par le contrôleur nodal pour mettre fin à l'ensemble d'une conférence en déconnectant tous les nœuds.
- GCC-Conference-Eject-User (*éjection d'utilisateur de conférence*) – Autorisée uniquement pour l'organisateur de la conférence, pour un nœud désigné par l'organisateur (ou pour un nœud qui est supérieur direct du nœud à éjecter dans la hiérarchie de connexion), cette

primitive est utilisée par le contrôleur nodal pour déconnecter un nœud donné d'une conférence en cours.

- GCC-Conference-Transfer (*transfert de conférence*) – Autorisée uniquement pour l'organisateur de la conférence ou pour un nœud désigné par l'organisateur, cette primitive est utilisée par le contrôleur nodal pour transférer des nœuds d'une conférence vers une autre conférence. Elle peut être employée dans une procédure de fusion ou d'éclatement de conférence.

### 7.1.2.1 GCC-Conference-Create (création d'une conférence)

La primitive de demande GCC-Conference-Create est utilisée par un contrôleur nodal pour créer une nouvelle conférence sur un nœud distant. Le nœud local entre automatiquement dans la conférence. Cette primitive peut être émise à tout instant. Lorsqu'une conférence est créée, le nœud auquel la primitive de demande de création est adressée (le nœud qui reçoit la primitive d'indication GCC-Conference-Create) entre également dans la conférence et en devient le fournisseur GCC sommital. Ce nœud reste fournisseur GCC sommital pour la durée de la conférence. Dans certaines mises en œuvre, il peut être possible de créer localement une conférence sans utiliser de primitives GCC. Dans ce cas, le nœud sur lequel la conférence est créée devient le fournisseur GCC sommital. Le Tableau 7-1 donne les paramètres et les types de primitives de ce service. La Figure 7-1 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-1 – Types de primitives et paramètres – GCC-Conference-Create**

Paramètre	Demande	Indication	Réponse	Confirmation
Nom de conférence	M	M(=)	M(=)	M(=)
Modificateur de nom de conférence	C		C	
Identificateur de conférence		M	M(=)	M
Mot de passe de l'organisateur	O	O(=)		
Mot de passe	O	O(=)		
Conférence verrouillée	M	M(=)		
Conférence listée	M	M(=)		
Conférence pouvant être présidée	M	M(=)		
Méthode de fin	M	M(=)		
Liste de privilèges du président	O	O(=)		
Liste de privilèges de conférence en mode présidé	O	O(=)		
Liste de privilèges de conférence en mode non présidé	O	O(=)		
Description de conférence	O	O(=)		
Identificateur de l'appelant	O	O(=)		
Adresse appelante	O	O(=)		
Adresse appelée	O	O		
Paramètres de domaine	M	M	M	M(=)
Qualité de service	M	M	M	M(=)
Adresse réseau locale	O		O	
Priorité de conférence	O	O(=)		

**Tableau 7-1 – Types de primitives et paramètres – GCC-Conference-Create**

Paramètre	Demande	Indication	Réponse	Confirmation
Mode de conférence	O	O(=)		
Données utilisateur	O	O(=)	O	O(=)
Résultat			M	M(=)

*Nom de conférence:* nom identifiant la conférence à créer. Il se compose d'une chaîne numérique ainsi que d'une chaîne de caractères unicode rangée 00 facultative, chacune de longueur maximale de 255 caractères. Si les deux formes du nom de conférence sont utilisées, un nœud qui souhaite entrer dans cette conférence peut donner l'une ou l'autre forme du nom dans la primitive de demande d'entrée. Une valeur numérique devra nécessairement être incluse dans la demande d'entrée, sous forme d'une variante numérique du nom. En conséquence, un nom de conférence en mode texte ne comportant que des caractères numériques ne fera jamais l'objet d'une comparaison et ne devra donc pas être utilisé – en d'autres termes, la variante en mode texte du nom de conférence doit comporter au moins un caractère non numérique.

*Modificateur de nom de conférence:* si le nœud demandeur ou le nœud répondeur est déjà engagé dans une conférence dont le nom de conférence, dans sa partie numérique ou sa partie texte, est identique au nom figurant dans la primitive de demande, ce paramètre sera également présent dans la primitive correspondante de demande ou de réponse. La valeur de ce paramètre sera unique pour toutes les conférences du nœud en question qui portent ce nom de conférence. Si ce modificateur est présent, il sera utilisé comme paramètre "modificateur du nom de conférence du nœud appelé" dans une primitive de demande GCC-Conference-Join faite par un autre nœud tentant d'entrer dans la conférence au moyen d'une connexion directe avec le nœud concerné. Ce modificateur est également présent dans la réponse à une primitive GCC-Conference-Query (interrogation de conférence) adressée à ce nœud. Ce paramètre est une chaîne numérique de longueur maximale 255 chiffres.

*Identificateur de conférence:* identificateur attribué localement à la conférence qui vient d'être créée. Toutes les références ultérieures à la conférence sont faites en utilisant l'identificateur de conférence comme un identificateur non ambigu. L'identificateur de conférence sera identique au sélecteur de domaine MCS utilisé localement pour identifier le domaine MCS associé à la conférence.

*Mot de passe de l'organisateur:* paramètre facultatif contenant une chaîne numérique et une chaîne de caractères unicode rangée 00 facultative utilisées par l'organisateur pour s'identifier dans des opérations ultérieures. L'organisateur peut se déconnecter puis entrer de nouveau dans la conférence en retrouvant ses privilèges d'organisateur si l'entrée se fait par une connexion directe avec le fournisseur GCC sommital. Il s'agit du mot de passe privé, d'une longueur maximale de 255 chiffres et 255 caractères, permettant à l'organisateur d'exécuter des opérations qui lui sont réservées. Si ce mot de passe a la valeur NULL (néant), l'organisateur n'aura pas la possibilité de retrouver ses privilèges lorsqu'il se reconnectera. Une valeur numérique devra nécessairement être incluse dans la demande d'entrée, sous forme d'une variante numérique du mot de passe d'organisateur. En conséquence, un mot de passe d'organisateur en mode texte ne comportant que des caractères numériques ne fera jamais l'objet d'une comparaison et ne devra donc pas être utilisé – en d'autres termes, la variante en mode texte du mot de passe de l'organisateur doit comporter au moins un caractère non numérique.

*Mot de passe:* ce paramètre est constitué d'une chaîne numérique et d'une chaîne de caractères unicode rangée 00 facultative, toutes deux de longueur maximale 255 chiffres et 255 caractères. Si aucun mot de passe est présent, la conférence n'est pas protégée par un mot de passe. Une valeur numérique devra nécessairement être incluse dans la demande d'entrée, sous forme d'une variante

numérique du mot de passe. En conséquence, un mot de passe en mode texte ne comportant que des caractères numériques ne fera jamais l'objet d'une comparaison et ne devra donc pas être utilisé – en d'autres termes, la variante en mode texte du mot de passe doit comporter au moins un caractère non numérique.

NOTE – Si la conférence est protégée par un mot de passe, le contrôleur nodal doit fournir un mot de passe numérique et peut également fournir un mot de passe alphabétique. Le mot de passe numérique est exigé afin de prendre en compte les nœuds qui ne possèdent pas de mécanisme adéquat d'entrée de texte. Dans le cas où un mot de passe alphabétique est utilisé, le mot de passe numérique qui doit également être fourni n'est pas nécessairement produit par l'utilisateur. Il peut être plus commode et plus sûr d'utiliser un mot de passe numérique produit par une machine.

*Conférence verrouillée*: le positionnement de cet indicateur verrouille une conférence dès sa création, interdisant à tout nœud d'entrer dans cette conférence s'il n'est pas ajouté d'une manière explicite par la primitive GCC-Conference-Invite (ou indirectement au moyen de la primitive GCC-Conference-Add (addition à une conférence)). Une conférence peut également être verrouillée à tout moment par l'utilisation de la primitive GCC-Conference-Lock (verrouillage de conférence). Il est possible d'utiliser la primitive GCC-Conference-Unlock (déverrouillage de conférence) pour déverrouiller une conférence.

*Conférence listée*: le positionnement de cet indicateur sur TRUE (vrai) signale que la conférence peut être listée en utilisant la facilité d'interrogation de conférence. Le positionnement de cet indicateur sur FALSE (faux) indique que la conférence ne sera pas listée.

*Conférence pouvant être présidée*: le positionnement de cet indicateur sur TRUE (vrai) signale que la conférence peut être placée en mode présidé au moyen de la primitive GCC-Conductor-Assign (assignation de président). Le positionnement de cet indicateur sur FALSE (faux) indique que la conférence se déroulera uniquement en mode non présidé et que les tentatives d'assignation de président seront rejetées.

*Méthode de fin*: paramètre indiquant la façon dont se terminera la conférence. Dans le cas de fin manuelle, la conférence se poursuivra jusqu'à ce qu'elle soit terminée explicitement par l'organisateur ou par un nœud désigné par l'organisateur au moyen de la primitive GCC-Conference-Terminate (fin de conférence). Dans le cas de fin automatique, la conférence se terminera lorsqu'il n'y a plus de nœuds qui en font partie.

*Liste de privilèges du président*: liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un éventuel président de conférence. Les indicateurs de cette liste correspondent aux opérations GCC-Conference-Terminate, GCC-Conference-Eject-User, GCC-Conference-Add, GCC-Conference-Lock, GCC-Conference-Unlock et GCC-Conference-Transfer (fin de conférence, éjection d'utilisateur, addition à une conférence, verrouillage, déverrouillage et transfert de conférence).

*Liste de privilèges de conférence en mode présidé*: liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un nœud quelconque pendant qu'une conférence est en mode présidé. Les indicateurs de cette liste correspondent aux mêmes opérations que pour le paramètre précédent.

*Liste de privilèges de conférence en mode non présidé*: liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un nœud quelconque pendant qu'une conférence est en mode non présidé. Les indicateurs de cette liste correspondent aux mêmes opérations que pour le paramètre précédent.

*Description de conférence*: chaîne de caractères unicode optionnelle d'une longueur maximale de 255 caractères pouvant être employée pour décrire la conférence. Cette chaîne est mémorisée par les fournisseurs GCC de la conférence afin d'être utilisée dans la réponse à des primitives de demande GCC-Conference-Query (interrogation de conférence).

*Identificateur de l'appelant*: chaîne de caractères unicode optionnelle (longueur maximale 255 caractères) pouvant être employée pour identifier le nœud qui appelle le nœud sur lequel la conférence doit être créée. L'utilisation de cette information ne fait pas partie du domaine d'application de la présente Recommandation. Elle peut, par exemple, être employée afin de permettre à un utilisateur de ce nœud de faire un choix parmi un ensemble limité de participants ayant l'autorisation de créer une conférence. Toutefois, comme cette chaîne peut recevoir une valeur quelconque, cette procédure de choix n'augmente pas nécessairement la sécurité d'une conférence.

*Adresse appelante*: adresse optionnelle pouvant être fournie dans la primitive MCS-Connect-Provider (connexion de fournisseur) lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Adresse appelée*: adresse optionnelle pouvant être fournie dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Paramètres de domaine*: paramètres de domaine à fournir dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Qualité de service*: paramètres de qualité de service à fournir dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Adresse réseau locale*: si cette information figure dans la demande ou la réponse, le fournisseur GCC local du nœud concerné l'utilisera comme paramètre "adresse réseau" dans la liste de descripteurs de conférence de la réponse à une primitive de demande GCC-Conference-Query (interrogation de conférence) issue d'un autre nœud. Dans le protocole de commande GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont la description et l'utilisation se trouvent dans l'Annexe B.

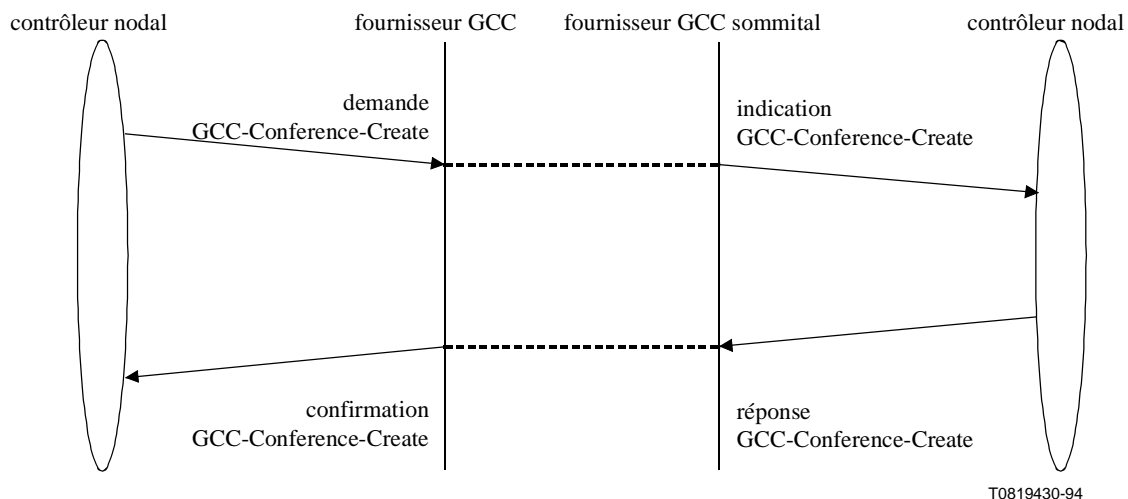
*Priorité d'une conférence*: paramètre facultatif qui spécifie la priorité d'une conférence. Ce paramètre peut être utilisé dans certaines situations, pour déterminer s'il convient d'accepter l'indication. Ce paramètre comporte deux sous-paramètres: priorité et procédure. La priorité est représentée par une valeur d'entier comprise entre zéro et 65535. La procédure indique les procédures à suivre pour interpréter la valeur de priorité. Actuellement, seules des procédures non normalisées sont prises en charge. Les procédures normalisées d'interprétation de ce paramètre feront l'objet d'un complément d'étude.

*Mode de conférence*: paramètre facultatif qui spécifie le mode d'une conférence. Ce paramètre est utilisé par la commande GCC et par le contrôleur nodal situé chez le fournisseur GCC sommital afin de déterminer les catégories nodales dans lesquelles un nœud entrant doit s'inscrire avant d'être autorisé à entrer dans la conférence. Les modes de conférence sont les suivants: *conventionnels-seulement*, qui n'autorise que des nœuds conventionnels à participer à la conférence; *comptés-contrôlés*, qui n'autorise que les nœuds comptés et contrôlés à participer à la conférence; *anonymes-contrôlés*, qui n'autorise que les nœuds comptés et conventionnels à participer à la conférence; et le *mode sans restriction*, qui autorise les trois catégories nodales à participer à la conférence. On notera que les nœuds conventionnels qui souhaitent entrer dans une conférence en mode comptés-contrôlés ou anonymes-contrôlés doivent toujours spécifier un mot de passe pour pouvoir participer.

*Données utilisateur*: données utilisateur optionnelles pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation, par exemple l'authentification, la taxation, etc.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "utilisateur rejeté", "ressources non

disponibles", "rejet pour violation de symétrie", "conférence verrouillée non prise en charge", "nom de conférence et modificateur de nom de conférence existent déjà", "paramètres de domaine non acceptés", "domaine non hiérarchique", "déconnexion par une couche inférieure" ou "erreur de connexion non spécifiée". Un résultat négatif de la primitive de confirmation GCC-Conference-Create ne signifie pas nécessairement que la connexion physique vers le nœud qui a fait l'objet d'une tentative de connexion a été interrompue.



T0819430-94

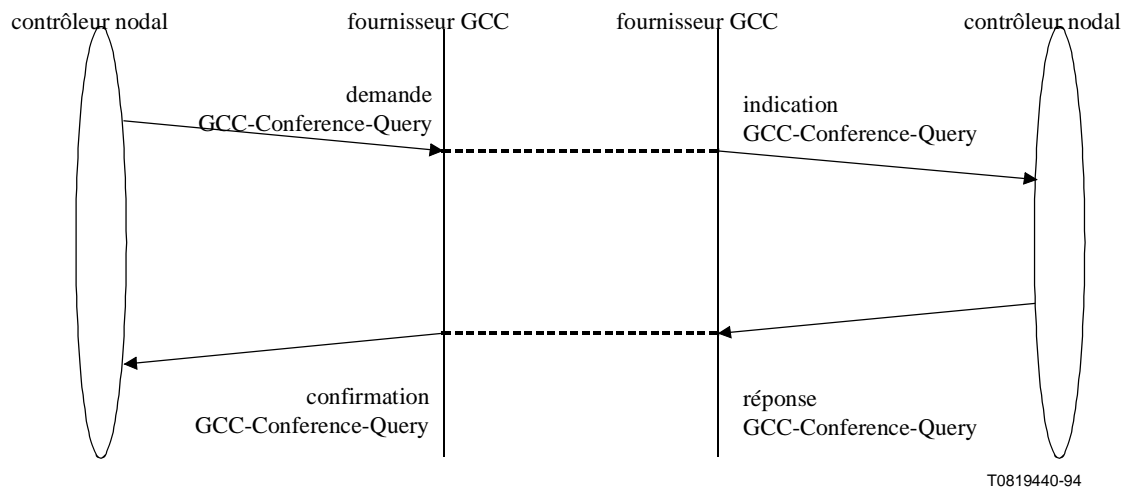
**Figure 7-1 – Séquence de primitives – GCC-Conference-Create**

### 7.1.2.2 GCC-Conference-Query (interrogation de conférence)

La primitive de demande GCC-Conference-Query peut être utilisée par un contrôleur nodal pour déterminer quelles sont les conférences qui existent actuellement sur une MCU donnée. Le Tableau 7-2 donne les paramètres et les types de primitives de ce service. La Figure 7-2 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-2 – GCC-Conference-Query – Types de primitives et paramètres**

Paramètre	Demande	Indication	Réponse	Confirmation
Type de nœud	M	M(=)	M	M(=)
Indicateur d'asymétrie	C	C(=)	C	C(=)
Liste de descripteurs de conférence				C
Indicateur d'attente d'invitation			O	O
Indicateur d'absence de conférence non listée			O	O
Adresse appelante	O	O(=)		
Adresse appelée	O	O		
Données utilisateur	O	O(=)	O	O(=)
Résultat			M	M(=)



**Figure 7-2 – Séquence de primitives – GCC-Conference-Query**

*Type de nœud:* le type de nœud est soit un terminal, une MCU ou un terminal multiaccès.

*Indicateur d'asymétrie:* paramètre nécessaire lors de certaines procédures d'établissement de conférence qui ont besoin de savoir qui est le nœud appelant (initiateur de la connexion physique) et qui est le nœud appelé dans une connexion point à point. Ce paramètre est exigé dans le cas d'une connexion physique entre deux nœuds (c'est-à-dire lorsque la connexion sous-jacente entre les deux nœuds utilise les cas RTPC, RNIS et RDCC de [UIT-T T.123]). Ce paramètre est optionnel dans le cas contraire. Un nœud qui connaît son statut d'appelant ou d'appelé positionnera cet indicateur sur la valeur appropriée. Dans certains cas, le nœud ne connaît pas avec certitude son statut d'appelant ou d'appelé. Dans ce cas, ce paramètre contiendra un nombre aléatoire de 32 bits. Si les deux nœuds indiquent qu'ils ne sont pas sûrs de leur état, les nombres aléatoires sont utilisés afin de déterminer lequel des nœuds doit être considéré comme appelant et lequel comme appelé par la procédure d'établissement de conférence (à noter que le résultat de cette procédure ne désignera pas nécessairement le véritable appelant et le véritable appelé). Le nœud qui a transmis le nombre aléatoire le plus grand sera considéré comme appelant par la procédure d'établissement de conférence. Si les deux nombres aléatoires sont égaux, le résultat sera considéré comme non satisfaisant et le demandeur émettra à nouveau la primitive de demande GCC-Conference-Query en utilisant un nombre aléatoire différent (la réponse utilisera également un nombre aléatoire). Si les deux nœuds répondent de manière non aléatoire, indiquant ainsi qu'ils connaissent leur statut d'appelant ou d'appelé, mais que les indications sont contradictoires (les nœuds signalant tous deux qu'ils sont appelants ou qu'ils sont appelés), le résultat sera aussi considéré comme non satisfaisant, et le demandeur émettra à nouveau la primitive de demande GCC-Conference-Query en utilisant un nombre aléatoire (la réponse utilisera également un nombre aléatoire). La valeur de ce paramètre dans toutes les primitives de demande et de réponse GCC-Conference-Query émises par ce nœud sur la même connexion physique restera inchangée tant qu'une primitive de demande reste sans confirmation par l'un des côtés d'une connexion donnée. Les nombres aléatoires doivent être distribués uniformément sur tout le domaine numérique.

*Liste de descripteurs de conférence:* liste de longueur variable de descripteurs de conférence. Chaque descripteur indique une conférence active et disponible pour une demande d'entrée. Cette liste ne contient pas les conférences qui ont été désignées comme non listées au moment de leur création. S'il n'existe pas de conférence disponible, la liste ne contient aucun élément. Tout descripteur de conférence contient les paramètres donnés par le Tableau 7-3.

*Adresse appelante*: adresse optionnelle pouvant être fournie dans la primitive MCS-Connect-Provider (connexion de fournisseur) lors de l'établissement d'une connexion au service MCS.

*Adresse appelée*: adresse optionnelle pouvant être fournie dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS.

*Données utilisateur*: données utilisateur optionnelles pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation, par exemple l'authentification, la taxation, etc.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "utilisateur rejeté", "paramètres de domaine non acceptés", "domaine non hiérarchique", "déconnexion par une couche inférieure" ou "erreur de connexion non spécifiée".

**Tableau 7-3 – Contenu d'un descripteur de conférence**

<b>Paramètre</b>	<b>Description</b>
Nom de conférence	Nom de conférence de la conférence. Si le nœud demandeur souhaite entrer dans cette conférence, ce paramètre contient la valeur qui sera utilisée comme paramètre "nom de conférence" dans la primitive de demande GCC-Conference-Join (entrée dans une conférence). Ce paramètre est constitué d'une chaîne numérique et d'une chaîne de caractères unicode rangée 00 facultative, chacune d'une longueur maximale de 255 caractères. Si les deux formes du nom sont fournies, l'une ou l'autre peut être utilisée dans la demande d'entrée dans la conférence.
Modificateur de nom de conférence (conditionnel)	Paramètre présent si le nœud renvoyant la réponse connaît la conférence sous un nom qui contient un modificateur de nom de conférence. Il sera utilisé comme paramètre "modificateur de nom de conférence du nœud appelé" dans la primitive de demande GCC-Conference-Join si le nœud demandeur souhaite entrer dans cette conférence. Ce paramètre est une chaîne numérique d'une longueur maximale de 255 chiffres.
Description de conférence (conditionnel)	Chaîne de caractères facultative unicode, d'une longueur maximale de 255 caractères, utilisée pour décrire la conférence. Ce paramètre peut être particulièrement utile pour faire une distinction entre des conférences présentes sous le même nom de conférence dans la liste de descripteurs de conférence.
Indicateur de verrouillage	Indicateur signalant si la conférence est actuellement verrouillée ou non.
Mot de passe en clair exigé	Indication que la conférence est protégée par un mot de passe, qui peut être utilisé sans chiffrement par une primitive de demande GCC-Conference-Join, sans demande préalable du mot de passe et sans épreuve.
Adresse réseau (conditionnel)	Information d'adressage fournie au nœud demandeur. Elle n'est fournie que si le paramètre facultatif d'adresse réseau locale a été utilisé pour ce nœud dans l'une des primitives d'établissement de connexion [GCC-Conference-Create, GCC-Conference-Join ou GCC-Conference-Invite (création, entrée ou invitation de/dans la conférence)]. Dans le protocole GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont on trouvera la description et l'utilisation en Annexe B.



**Tableau 7-3 – Contenu d'un descripteur de conférence**

Paramètre	Description
Indicateur de conférence par défaut	Indicateur facultatif signalant si une conférence particulière doit être considérée comme étant la conférence par défaut à laquelle participer. Dans une conférence rendez-vous où l'utilisateur est censé choisir manuellement, dans une liste, une conférence à laquelle participer, ce paramètre doit avoir la valeur FAUX pour toutes les conférences. Dans une conférence rendez-vous où une information <i>a priori</i> permet à une unité MCU (pont de conférence) de déterminer la conférence à laquelle il y a lieu de participer (et où, par exemple, des raisons de sécurité interdisent l'utilisation de la primitive GCC-Conference-Invite), cet indicateur peut être mis à la valeur VRAI pour une seule et unique conférence. Si cet indicateur a la valeur VRAI pour plus d'une seule conférence, ou pour une conférence verrouillée, le récepteur ne doit pas en tenir compte.
Mode de conférence	Paramètre facultatif servant à spécifier le mode d'une conférence. Il peut servir à déterminer la catégorie nodale appropriée qu'il convient de spécifier à l'entrée dans une conférence.

*Indicateur d'attente d'invitation:* indicateur facultatif qui peut être réglé par une unité MCU. S'il a la valeur VRAI, ce fanion indique que le nœud récepteur doit attendre de recevoir une invitation à une conférence et qu'il ne doit pas d'avoir accès ou de donner naissance à une conférence. L'absence de cet indicateur, ou la présence d'une valeur FAUX, n'implique pas qu'il faille tenter d'avoir accès ou de donner naissance à une conférence. Cet indicateur ne doit pas être mis à VRAI par un terminal à accès unique ou multiple. Aucune limite de temps n'est impliquée par une valeur VRAI pour cet indicateur. Un nœud souhaitant déterminer si l'unité MCU a toujours l'intention de le faire attendre l'invitation peut émettre une nouvelle demande GCC-Conference-Query afin de recevoir, en confirmation, le réglage actuel de cet indicateur.

*Indicateur d'absence de conférence non listée:* indicateur facultatif qui signale, lorsqu'il a la valeur VRAI, qu'il n'y a pas de conférences non listées ouvertes à participation. L'absence de cet indicateur, ou la présence d'une valeur FAUX, n'implique aucune information relative à la disponibilité de conférences listées.

### **7.1.2.3 GCC-Conference-Join (addition à une conférence)**

La primitive de demande GCC-Conference-Join peut être utilisée pour faire entrer le nœud local dans une conférence existante. Cette primitive peut être émise à tout instant. Un nœud peut appartenir simultanément à plusieurs conférences. Si la conférence est protégée par un mot de passe, le paramètre "mot de passe" doit contenir l'information correcte pour que l'entrée dans la conférence réussisse. Le Tableau 7-4 donne les paramètres et les types de primitives de ce service. La Figure 7-3 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

NOTE – La vérification du mot de passe est de la responsabilité du contrôleur nodal et non de celle du fournisseur GCC sommital. Il est possible que la définition du terme "correct" puisse être moins restrictive qu'une coïncidence stricte caractère par caractère. Par exemple, dans le cas d'un mot de passe sous forme de texte, le contrôleur nodal peut choisir de faire la comparaison en ignorant les différences entre majuscules et minuscules.

**Tableau 7-4 – GCC-Conference-Join – Types de primitives et paramètres**

Paramètre	Demande	Indication	Réponse	Confirmation
Nom de conférence	M			M
Modificateur de nom de conférence du nœud appelé	C			C(=RQ)
Modificateur de nom de conférence du nœud appelant	C			C(=RQ)
Identificateur de conférence		M	M(=)	M
Mot de passe d'organisateur	O	O(=)		
Mot de passe	C	C(=)	C	C(=)
Identificateur de l'appelant	O	O(=)		
Adresse appelante	O	C(=)		
Adresse appelée	O	C		
Paramètres de domaine	M			M
Qualité de service	M			M
Mot de passe en clair exigé				M
Conférence verrouillée				M
Conférence listée				M
Conférence pouvant être présidée				M
Méthode de fin				M
Liste de privilèges du président				C
Liste de privilèges de conférence en mode présidé				C
Liste de privilèges de conférence en mode non présidé				C
Description de conférence				C
Adresse réseau locale	O			
Catégorie nodale	O	O(=)	O	O(=)
Mode de conférence			O	O(=)
Données utilisateur	O	O(=)	O	O(=)
Résultat			M	M(=)

*Nom de conférence*: nom identifiant la conférence dans laquelle se fait l'entrée. Dans la primitive de demande, ce paramètre se compose soit d'une chaîne numérique soit d'une chaîne de texte unicode rangée 00, dans les deux cas d'une longueur maximale de 255 caractères. Si les parties numérique et alphabétique du nom de conférence ont été utilisées lors de la création de la conférence, le contrôleur nodal doit déterminer s'il convient d'envoyer le nom de conférence sous forme d'une chaîne alphabétique ou d'une chaîne numérique, selon sa valeur. Une valeur composée uniquement de caractères numériques doit être spécifiée comme étant une chaîne numérique, alors qu'une valeur comportant au moins un caractère non numérique doit être spécifiée sous forme de chaîne alphabétique. Dans la primitive de confirmation, ce paramètre contient le nom complet de la conférence comportant les formes numérique et alphabétique si celles-ci étaient présentes lors de la création.

*Modificateur de nom de conférence du nœud appelé:* ce paramètre figurera dans la primitive si le nœud connecté directement au nœud entrant (le nœud appelé vers lequel se fait une tentative de connexion) a fourni un modificateur de nom de conférence faisant partie du nom sous lequel il connaît la conférence. Il indique le modificateur de nom de conférence tel qu'il est connu par le nœud appelé. Ce paramètre est une chaîne numérique d'une longueur maximale de 255 chiffres.

*Modificateur de nom de conférence du nœud appelant:* ce paramètre figurera dans la primitive s'il existe déjà sur le nœud émettant la demande une conférence portant le même nom que la conférence dans laquelle doit se faire l'entrée. Dans ce cas, le paramètre indiquera le modificateur de nom de conférence sous lequel la conférence sera connue au sein du nœud local. Sa valeur éventuelle devra différer de tout modificateur de nom de conférence déjà utilisé par une conférence quelconque du nœud local portant le même nom de conférence. S'il est présent, ce paramètre devient le "modificateur de nom de conférence du nœud appelé" qui sera employé comme référence par un autre nœud lors d'une tentative d'entrer dans la conférence au moyen d'une connexion au nœud local. Ce modificateur est également fourni (dans le cas où la conférence est listée) comme paramètre "modificateur de nom de conférence" par toute primitive de réponse GCC-Conference-Query (interrogation de conférence) envoyée par ce nœud. Ce paramètre est une chaîne numérique d'une longueur maximale de 255 chiffres.

*Identificateur de conférence:* au niveau du fournisseur GCC sommital (dans les primitives d'indication/réponse), ce paramètre est l'identificateur de la conférence dans laquelle le nœud demandeur souhaite entrer. Ce paramètre est renvoyé dans la primitive de confirmation par le fournisseur GCC pour indiquer l'identificateur attribué localement et utilisé par la suite dans toutes les références à la conférence. L'identificateur de conférence sera identique au sélecteur de domaine MCS utilisé localement pour identifier le domaine MCS associé à la conférence.

*Mot de passe de l'organisateur:* paramètre numérique facultatif contenant une chaîne numérique ou une chaîne de texte unicode rangée 00, de longueur maximale 255 chiffres ou caractères, pouvant être utilisée par l'organisateur lorsqu'il entre de nouveau dans la conférence après s'être déconnecté. Si l'identificateur est identique à l'identificateur utilisé lors de la création de la conférence et si l'entrée se fait par une connexion directe au fournisseur GCC sommital (et non par une connexion à travers une unité MCU intermédiaire), le nœud entrant reçoit les privilèges d'organisateur. L'organisateur qui fournit un mot de passe correct a l'autorisation d'entrer même dans des conférences verrouillées. Si la conférence est protégée par un mot de passe, le mot de passe correct doit être fourni en même temps que le mot de passe d'organisateur pour que l'entrée réussisse. Si le mot de passe d'organisateur est présent mais n'est pas correct, la demande d'entrée sera rejetée. Le contrôleur nodal détermine le critère utilisé pour vérifier si le mot de passe d'organisateur est conforme à la valeur donnée lors de la création de la conférence. Le contrôleur nodal doit déterminer s'il convient d'envoyer le nom de conférence sous forme d'une chaîne alphabétique ou d'une chaîne numérique, selon sa valeur. Une valeur composée uniquement de caractères numériques doit être spécifiée comme étant une chaîne numérique, alors qu'une valeur comportant au moins un caractère non numérique doit être spécifiée sous forme de chaîne alphabétique.

*Mot de passe:* paramètre utilisé pour accéder à une conférence protégée par mot de passe. Dans la forme de demande de cette primitive, ce paramètre ne contiendra un mot de passe que si le résultat "Challenge Response Required" (réponse à l'épreuve requise) a été reçu pour cette conférence dans une primitive de confirmation GCC-Conference-Join ou si le paramètre "réponse en clair requise" est positionné pour cette conférence dans le descripteur de conférence d'une primitive de confirmation GCC-Conference-Query précédente. En cas de mot de passe en clair, ce paramètre est soit une chaîne numérique, soit une chaîne de caractères unicode rangée 00, d'une taille maximale de 255 chiffres ou caractères. Une chaîne de texte ne peut être utilisée que si un mot de passe alphabétique a été défini en plus du mot de passe numérique lors de la création de la conférence. Le contrôleur nodal doit déterminer s'il convient d'envoyer le nom de conférence sous forme d'une chaîne alphabétique ou d'une chaîne numérique, selon sa valeur. Une valeur composée uniquement de caractères numériques doit être spécifiée comme étant une chaîne numérique, alors qu'une valeur

comportant au moins un caractère non numérique doit être spécifiée sous forme de chaîne alphabétique. Dans le cas d'un mot de passe chiffré, ce paramètre contient le mot de passe chiffré en utilisant l'un des algorithmes spécifiés dans l'épreuve reçue précédemment. Dans le cas d'un mot de passe envoyé (en clair ou sous forme chiffrée) en réponse à une épreuve, ce paramètre contiendra également un indicateur qui doit être identique à l'indicateur reçu dans l'épreuve. Aucun indicateur n'est nécessaire dans le cas d'un mot de passe initialisé en réponse à l'indicateur "mot de passe en clair" de la primitive d'indication GCC-Conference-Query. Dans la forme demande de cette primitive, ce paramètre peut également contenir une épreuve pour le nœud récepteur. Il n'existe aucune limitation sur le moment auquel ce paramètre contient une épreuve.

Dans la forme réponse de cette primitive, le paramètre peut contenir une épreuve pour le demandeur indiquant qu'un mot de passe est nécessaire pour rejoindre cette conférence. Ce paramètre contient dans ce cas une information qui spécifie quelle sera la forme de mot de passe acceptée (en clair et/ou chiffré au moyen d'une liste d'algorithmes de chiffrement non normalisés), un nombre entier servant d'indicateur pour identifier l'épreuve ainsi que toute information supplémentaire nécessaire au chiffrement. Le paramètre de résultat de cette primitive sera positionné sur "Challenge Response Required" dans le cas où le paramètre "mot de passe" contient une épreuve. Aucune connexion n'est établie dans ce cas. Ce paramètre peut également contenir un mot de passe (en clair ou chiffré) dans la forme réponse de cette primitive envoyé en réponse à une épreuve du nœud demandeur.

Si ce paramètre figure dans une primitive d'indication avec un format qui ne satisfait pas le nœud récepteur, ce nœud émettra une réponse contenant le résultat "Invalid Challenge Response" (réponse à l'épreuve non valide). Si ce paramètre de la primitive d'indication a un format correct mais ne contient pas le mot de passe correct, la réponse contiendra le code résultat "mot de passe non valide".

*Identificateur de l'appelant*: chaîne de texte unicode d'une taille maximale de 255 caractères pouvant être utilisée pour fournir l'identité du nœud appelant au nœud sur lequel réside le fournisseur GCC sommital. L'utilisation de cette information ne fait pas partie du domaine d'application de la présente Recommandation. Elle peut, par exemple, être employée afin de permettre à un utilisateur de ce nœud de faire un choix parmi un ensemble limité de participants ayant le droit d'entrer dans la conférence. Toutefois, comme cette chaîne peut recevoir une valeur quelconque, cette procédure de choix n'augmente pas nécessairement la sécurité d'une conférence.

*Adresse appelante*: adresse facultative présente dans la primitive MCS-Connect-Provider (connexion de fournisseur) lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre. La présence de ce paramètre dans l'indication est conditionnelle: le paramètre n'est présent que lorsque le nœud qui émet la demande GCC-Conference-Join essaye d'établir une connexion directe avec le nœud contenant le fournisseur GCC sommital pour la conférence dont l'entrée est demandée.

*Adresse appelée*: adresse facultative présente dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre. La présence de ce paramètre dans l'indication est conditionnelle: le paramètre n'est présent que lorsque le nœud qui émet la demande GCC-Conference-Join essaye d'établir une connexion directe avec le nœud contenant le fournisseur GCC sommital pour la conférence dont l'entrée est demandée.

*Paramètres de domaine*: paramètres de domaine présents dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Qualité de service*: paramètres de qualité de service présents dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Mot de passe en clair exigé:* paramètre booléen indiquant qu'une conférence est protégée par un mot de passe pouvant être utilisé sans chiffrement dans une primitive de demande GCC-Conference-Join sans avoir fait au préalable l'objet d'une épreuve pour le mot de passe. Cette information sera utilisée par le fournisseur GCC pour générer la réponse à la primitive d'indication GCC-Conference-Query si le nœud entrant dans la conférence est une MCU. Cette information sera utilisée par le contrôleur nodal pour déterminer si le paramètre mot de passe peut être inclus dans la demande GCC-Conference-Transfer.

*Conférence verrouillée:* indicateur définissant si la conférence dans laquelle se fait l'entrée est verrouillée ou non. Si le nœud entrant est une MCU, cette information sera utilisée pour générer la réponse à une primitive d'indication GCC-Conference-Query (interrogation de conférence).

*Conférence listée:* indicateur définissant si la conférence dans laquelle se fait l'entrée est listée ou non. Si le nœud entrant est une MCU, cette information sera utilisée pour générer la réponse à une primitive d'indication GCC-Conference-Query. Les conférences indiquées comme non listées ne doivent pas figurer dans la liste de conférences fournie par la primitive de réponse GCC-Conference-Query.

*Conférence pouvant être présidée:* le positionnement de cet indicateur sur TRUE (vrai) signale que la conférence peut être placée en mode présidé au moyen de la primitive GCC-Conductor-Assign (assignation de président). Le positionnement de cet indicateur sur FALSE (faux) indique que la conférence se déroulera uniquement en mode non présidé et que les tentatives d'assignation de président seront rejetées.

*Méthode de fin:* indicateur définissant la règle de fin d'une conférence. La conférence peut se terminer de façon manuelle ou automatique.

*Liste de privilèges du président:* liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un président de conférence éventuel. Les indicateurs de cette liste correspondent aux opérations GCC-Conference-Terminate, GCC-Conference-Eject-User, GCC-Conference-Add, GCC-Conference-Lock, GCC-Conference-Unlock et GCC-Conference-Transfer (fin de conférence, éjection d'utilisateur, addition à une conférence, verrouillage, déverrouillage et transfert de conférence).

*Liste de privilèges de conférence en mode présidé:* liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un nœud quelconque pendant qu'une conférence est en mode présidé. Les indicateurs de cette liste correspondent aux mêmes opérations que pour le paramètre précédent.

*Liste de privilèges de conférence en mode non présidé:* liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un nœud quelconque pendant qu'une conférence est en mode non présidé. Les indicateurs de cette liste correspondent aux mêmes opérations que pour le paramètre précédent.

*Description de conférence:* chaîne de caractères unicode optionnelle d'une longueur maximale de 255 caractères utilisée pour décrire la conférence dans laquelle se fait l'entrée. Ce paramètre ne figure dans la primitive de confirmation que s'il a été fourni lors de la création de la conférence.

*Adresse réseau locale:* si ce paramètre figure dans la demande, le fournisseur GCC local du nœud correspondant l'utilisera comme paramètre "adresse réseau" dans la liste de descripteurs de conférence faisant partie de la réponse à une primitive de demande GCC-Conference-Query (interrogation de conférence) issue d'un autre nœud. Dans le protocole de commande GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont la description et l'utilisation se trouvent dans l'Annexe B.

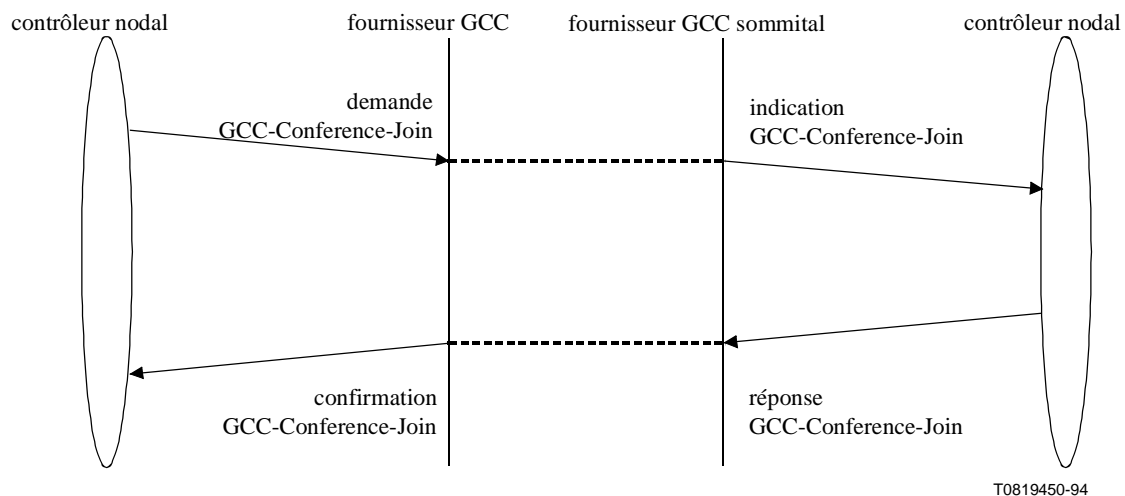
*Catégorie nodale:* si ce paramètre est inclus dans la demande, il peut être utilisé par le fournisseur GCC local se trouvant au nœud récepteur afin de déterminer la catégorie nodale préférée du nœud entrant. Cette option peut être neutralisée dans la réponse d'entrée si la catégorie suggérée n'est pas

acceptable par le nœud récepteur. S'il n'est pas spécifié, la catégorie requise par défaut est "conventionnel".

*Mode de conférence*: paramètre facultatif qui spécifie le mode d'une conférence. Ce paramètre est utilisé par la commande GCC et par le contrôleur nodal situé chez le fournisseur GCC sommital afin de déterminer les catégories nodales dans lesquelles un nœud entrant doit s'inscrire avant d'être autorisé à entrer dans la conférence. Les modes de conférence sont les suivants: *conventionnels-seulement*, qui n'autorise que des nœuds conventionnels à participer à la conférence; *comptés-contrôlés*, qui n'autorise que les nœuds comptés et contrôlés à participer à la conférence; *anonymes contrôlés*, qui n'autorise que les nœuds comptés et conventionnels à participer à la conférence; et le *mode sans restriction*, qui autorise les trois catégories nodales à participer à la conférence. On notera que les nœuds conventionnels qui souhaitent entrer dans une conférence en mode comptés-contrôlés ou anonymes-contrôlés doivent toujours spécifier un mot de passe pour pouvoir participer.

*Données utilisateur*: données utilisateur optionnelles pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation, par exemple l'authentification, la taxation, etc.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "utilisateur rejeté", "conférence non valide", "mot de passe non valide", "réponse à l'épreuve exigée", "réponse à l'épreuve non valide", "mot de passe d'organisateur non valide", "paramètres de domaine non acceptés", "domaine non hiérarchique", "déconnexion par une couche inférieure" ou "erreur de connexion non spécifiée". Un résultat négatif de la primitive de confirmation GCC-Conference-Join ne signifie pas nécessairement que la connexion physique vers le nœud qui a fait l'objet d'une tentative de connexion a été interrompue.



**Figure 7-3 – GCC-Conference-Join – Séquence de primitives**

#### 7.1.2.4 GCC-Conference-Invite (invitation à une conférence)

La primitive de demande GCC-Conference-Invite est utilisée par un contrôleur nodal pour inviter un nœud à entrer dans une conférence. Cette primitive peut être employée comme suite à une primitive d'indication GCC-Conference-Add (addition à une conférence) ou peut être envoyée directement par le nœud invitant. Si l'addition réussit, la MCU responsable de l'addition invite le nœud ajouté à entrer dans la conférence en envoyant la primitive de demande GCC-Conference-Invite. Il est à noter que le nœud invité n'a pas besoin de posséder le mot de passe pour accepter l'invitation même si la conférence est protégée par un mot de passe. Le Tableau 7-5 donne les

paramètres et les types de primitives de ce service. La Figure 7-4 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-5 – GCC-Conference-Invite – Types de primitives et paramètres**

Paramètre	Demande	Indication	Réponse	Confirmation
Identificateur de conférence	M	M	M(=IN)	M(=RQ)
Nom de conférence		M		
Modificateur de nom de conférence			C	
Identificateur de l'appelant	O	O(=)		
Adresse appelante	O	O(=)		
Adresse appelée	O	O		
Paramètres de domaine		M	M	
Qualité de service		M	M	
Mot de passe en clair exigé		M		
Conférence verrouillée		M		
Conférence listée		M		
Conférence pouvant être présidée		M		
Méthode de fin		M		
Liste de privilèges du président		C		
Liste de privilèges de conférence en mode présidé		C		
Liste de privilèges de conférence en mode non présidé		C		
Description de conférence		C		
Adresse réseau locale			O	
Priorité de conférence	O	O(=)		
Catégorie nodale	O	O(=)		
Mode de conférence	O	O(=)		
Données utilisateur	O	O(=)	O	O(=)
Résultat			M	M(=)

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Nom de conférence*: nom de la conférence tel qu'il est contenu dans le profil de conférence. Si le nom de conférence contient à la fois les parties numérique et texte, les deux parties doivent être présentes dans ce paramètre.

*Modificateur de nom de conférence*: si une conférence existe déjà avec un nom identique au nom de la conférence dans laquelle se fait l'entrée sur le nœud émettant la réponse à l'invitation, ce paramètre sera présent dans la primitive de réponse GCC-Conference-Invite et indiquera le modificateur de nom de conférence sous lequel la conférence sera connue du nœud local. Si ce modificateur est présent, il sera différent de tout modificateur de nom de conférence déjà utilisé pour toute autre conférence à laquelle appartient le nœud local et qui porte le même nom de conférence. Si ce paramètre est utilisé, il deviendra le "modificateur de nom de conférence du nœud appelant" qui servira de référence pour la conférence à un autre nœud qui tente d'entrer dans la conférence par connexion au nœud local. Ce nom figure également comme paramètre "modificateur de nom de conférence" dans toute réponse à une primitive GCC-Conference-Query (interrogation

de conférence) adressée à ce nœud, dans le cas où la conférence est listée. Ce paramètre est une chaîne numérique de 255 nombres au plus.

*Identificateur de l'appelant*: chaîne de texte unicode, de longueur maximale 255 caractères, pouvant être employée pour fournir l'identité du nœud sur lequel la conférence doit être créée. L'utilisation de cette information ne fait pas partie du domaine d'application de la présente Recommandation. Elle peut, par exemple, être employée afin de permettre à un utilisateur de ce nœud de faire un choix parmi un ensemble limité de participants ayant le droit d'inviter ce nœud dans une conférence. Toutefois, comme cette chaîne peut recevoir une valeur quelconque, cette procédure de choix n'augmente pas nécessairement la sécurité d'une conférence.

*Adresse appelante*: adresse optionnelle présente dans la primitive MCS-Connect-Provider (connexion de fournisseur) lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Adresse appelée*: adresse optionnelle présente dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Paramètres de domaine*: paramètres de domaine présents dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Qualité de service*: paramètres de qualité de service présents dans la primitive MCS-Connect-Provider lors de l'établissement d'une connexion au service MCS. Se référer à [UIT-T T.122] pour l'interprétation de ce paramètre.

*Mot de passe en clair exigé*: paramètre booléen indiquant qu'une conférence est protégée par un mot de passe pouvant être utilisé sans chiffrement dans une primitive de demande GCC-Conference-Join sans avoir fait au préalable l'objet d'une épreuve pour le mot de passe. Si le nœud entrant est une MCU, cette information sera utilisée lors de la génération de la réponse à une primitive d'indication GCC-Conference-Query (interrogation de conférence). Cette information sera utilisée par le contrôleur de nœud afin de déterminer s'il y a lieu ou non d'inclure le paramètre mot de passe dans une primitive de demande GCC-Conference-Transfer.

*Conférence verrouillée*: indicateur définissant si la conférence dans laquelle se fait l'entrée est verrouillée ou non. Si le nœud entrant est une MCU, cette information sera utilisée pour générer la réponse à une primitive d'indication GCC-Conference-Query.

*Conférence listée*: indicateur définissant si la conférence dans laquelle se fait l'entrée est listée ou non. Si le nœud entrant est une MCU, cette information sera utilisée pour générer la réponse à une primitive d'indication GCC-Conference-Query. En particulier, les conférences indiquées comme non listées ne doivent pas figurer sur la liste de conférences fournie par la primitive de réponse GCC-Conference-Query.

*Conférence pouvant être présidée*: le positionnement de cet indicateur sur TRUE (vrai) signale que la conférence peut être placée en mode présidé au moyen de la primitive GCC-Conductor-Assign (assignation de président). Le positionnement de cet indicateur sur FALSE (faux) indique que la conférence se déroulera uniquement en mode non présidé et que les tentatives d'assignation de président seront rejetées.

*Méthode de fin*: indicateur définissant la règle de fin d'une conférence. La conférence peut se terminer de façon manuelle ou automatique.

*Liste de privilèges du président*: liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un éventuel président de conférence. Les indicateurs de cette liste correspondent aux opérations GCC-Conference-Terminate, GCC-Conference-Eject-User, GCC-Conference-Add, GCC-Conference-Lock, GCC-Conference-Unlock et



GCC-Conference-Transfer (fin de conférence, éjection d'utilisateur, addition à une conférence, verrouillage, déverrouillage et transfert de conférence).

*Liste de privilèges de conférence en mode présidé*: liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un nœud quelconque pendant qu'une conférence est en mode présidé. Les indicateurs de cette liste correspondent aux mêmes opérations que pour le paramètre précédent.

*Liste de privilèges de conférence en mode non présidé*: liste d'indicateurs signalant quelles sont les fonctions dont l'organisateur autorise l'utilisation à un nœud quelconque pendant qu'une conférence est en mode non présidé. Les indicateurs de cette liste correspondent aux mêmes opérations que pour le paramètre précédent.

*Description de conférence*: chaîne de texte unicode d'une longueur maximale de 255 caractères décrivant la conférence. Elle n'est présente dans la primitive de confirmation que si elle a été fournie lors de la création de la conférence.

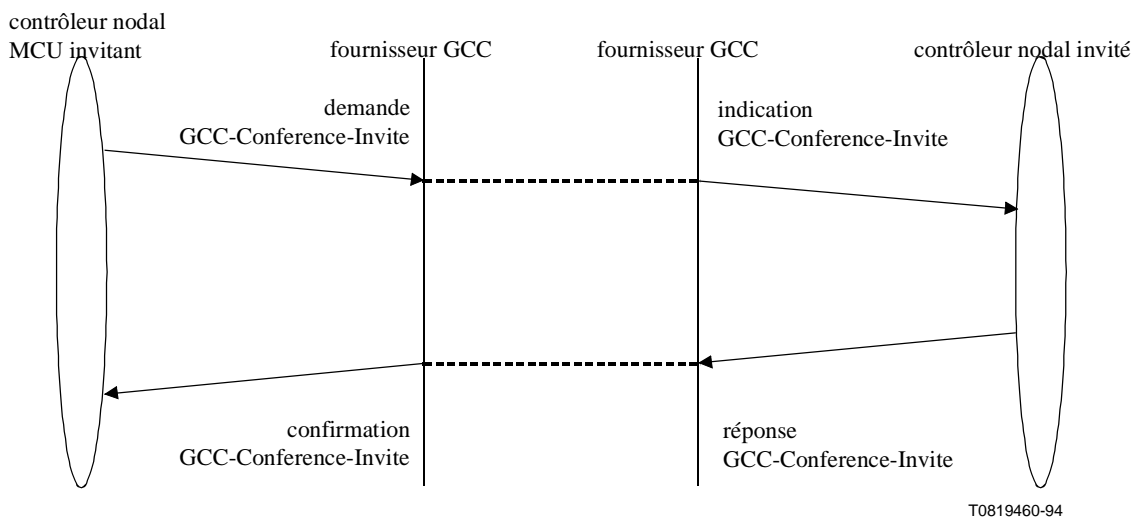
*Adresse réseau locale*: si cette information figure dans la réponse, le fournisseur GCC local l'utilisera comme paramètre "adresse réseau" dans la liste de descripteurs de conférence faisant partie de la réponse à une primitive de demande GCC-Conference-Query issue d'un autre nœud. Dans le protocole de commande GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont la description et l'utilisation se trouvent dans l'Annexe B.

*Priorité d'une conférence*: paramètre facultatif qui spécifie la priorité d'une conférence. Ce paramètre peut être utilisé dans certaines situations, pour déterminer s'il convient d'accepter l'indication. Ce paramètre comporte deux sous-paramètres: priorité et procédure. La priorité est représentée par une valeur d'entier comprise entre zéro et 65535. La procédure indique les procédures à suivre pour interpréter la valeur de priorité. Actuellement, seules des procédures non normalisées sont prises en charge. Les procédures normalisées d'interprétation de ce paramètre feront l'objet d'un complément d'étude.

*Mode de conférence*: paramètre facultatif qui spécifie le mode d'une conférence. Ce paramètre est utilisé par la commande GCC et par le contrôleur nodal situé chez le fournisseur GCC sommital afin de déterminer les catégories nodales dans lesquelles un nœud entrant doit s'inscrire avant d'être autorisé à entrer dans la conférence. Les modes de conférence sont les suivants: *conventionnels-seulement*, qui n'autorise que des nœuds conventionnels à participer à la conférence; *comptés-contrôlés*, qui n'autorise que les nœuds comptés et contrôlés à participer à la conférence; *anonymes-contrôlés*, qui n'autorise que les nœuds comptés et conventionnels à participer à la conférence; et le *mode sans restriction*, qui autorise les trois catégories nodales à participer à la conférence. On notera que les nœuds conventionnels qui souhaitent entrer dans une conférence en mode comptés-contrôlés ou anonymes-contrôlés doivent toujours spécifier un mot de passe pour pouvoir participer.

*Données utilisateur*: données utilisateur facultatives pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation, par exemple l'authentification, la taxation, etc.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "utilisateur rejeté", "conférence non valide", "paramètres de domaine non acceptés", "domaine non hiérarchique", "déconnexion par une couche inférieure" ou "erreur de connexion non spécifiée" ou "catégorie nodale inacceptable". Un résultat négatif de la primitive de confirmation GCC-Conference-Invite ne signifie pas nécessairement que la connexion physique vers le nœud qui a fait l'objet d'une tentative de connexion a été interrompue.



**Figure 7-4 – Séquence de primitives – GCC-Conference-Invite**

### 7.1.2.5 GCC-Conference-Add (addition à une conférence)

La primitive de demande GCC-Conference-Add peut être utilisée par un contrôleur nodal afin d'ajouter un nœud donné à une conférence existante en demandant à une MCU d'effectuer l'appel par numérotation du nœud indiqué. Cette primitive n'est valide que si elle est émise par l'organisateur ou un nœud désigné par l'organisateur. Si la MCU chargée de l'addition est indiquée, la séquence de primitives est échangée directement entre le demandeur et la MCU en charge. Si la MCU chargée de l'addition n'est pas indiquée, l'indication est fournie par le fournisseur GCC sommital. Dans ce cas, l'échange de primitives est également direct s'il existe sur ce nœud une jonction utilisable pour l'addition. Si le contrôleur nodal du fournisseur GCC sommital gère une base de données centrale sur les informations des jonctions de toutes les MCU de la conférence ou peut accéder à une telle base, il peut réaliser indirectement l'addition par l'émission d'une nouvelle primitive de demande GCC-Conference-Add qui spécifie une MCU chargée de l'addition, même si la MCU sommitale n'est pas un nœud ayant été désigné par l'organisateur comme ayant le privilège pour une telle opération.

Une fois que la mesure éventuelle d'établissement de la connexion physique vers le nœud ajouté a été prise, la MCU chargée de l'addition émettra la primitive de demande GCC-Conference-Invite (invitation à une conférence) afin d'inviter le nœud dans la conférence. Si la MCU chargée de l'addition dispose déjà d'une connexion physique vers le nœud ajouté, la MCU chargée de l'addition peut émettre la primitive de demande GCC-Conference-Invite directement vers le nœud ajouté sans établir de nouvelle connexion physique. Si la connexion a été établie sans connaître l'adresse réseau du nœud ajouté, par exemple si le nœud ajouté avait procédé à un appel par numérotation vers la MCU chargée de l'addition pour se connecter à une autre conférence, les paramètres d'adresse réseau du répertoire de conférence peuvent être utilisés afin de fournir l'information nécessaire pour mettre en correspondance l'adresse réseau d'un nœud déjà connecté avec celle d'un nœud à ajouter. Le Tableau 7-6 donne les paramètres et les types de primitives de ce service. Les Figures 7-5 et 7-6 indiquent la séquence d'événements correspondant à l'utilisation de cette primitive respectivement dans les cas direct et indirect.

**Tableau 7-6 – GCC-Conference-Add – Types de primitives et paramètres**

Paramètre	Demande	Indication	Réponse	Confirmation
Identificateur de conférence	M	M	M(=IN)	M(=RQ)
Adresse réseau	M	M(=)	M(=)	M(=)
Identificateur du nœud MCU chargé de l'addition	O			
Identificateur du nœud demandeur		M	M(=)	
Catégorie nodale	O	O(=)		
Données utilisateur	O	O(=)	O	O(=)
Résultat			M	M(=)

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Adresse réseau*: ce paramètre énumère une ou plusieurs descriptions de connexion, représentant chacune une portion de la connexion logique totale vers le nœud ajouté. Une description de connexion particulière donne des informations sur la connexion réseau à établir vers le nœud ajouté (type de réseau, type de circuit, adresse réseau) et, facultativement, sur les moyens permettant, le cas échéant, d'agréger des circuits numériques, sur les profils multimédias que l'unité MCU ajoutante peut exploiter sur la connexion, et sur les médias (par exemple audio, vidéo, données) intéressés par la connexion. Les paramètres composant chaque élément de l'adresse réseau sont présentés au 7.1.2.5.1. Dans le protocole de commande GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont la description et l'utilisation se trouvent dans l'Annexe B.

La partie adresse NSAP de l'adresse de transport peut encapsuler certaines formes suggérant par convention le choix d'une liaison commutée ou d'un protocole de réseau sans connexion. Dans ce cas, l'élément figurant dans la liste indique d'une manière autonome une possibilité indépendante de tout canal agrégé ou de toute adresse non normalisée qui peut également être présente. Une adresse de transport peut par ailleurs, et en remplacement, être incomplète et peut dépendre implicitement de l'utilisation partagée d'un canal de données ouvert au sein d'un multiplexage multimédia de canaux agrégés, tel que spécifié dans [UIT-T T.123] ou spécifié d'une manière non normalisée. Dans de tels cas, il peut être nécessaire de disposer d'une adresse NSAP et/ou d'un sélecteur de transport spécifié localement afin de piloter de nouvelles connexions au sein du canal de données vers le point de terminaison correct. Les adresses NSAP sont véhiculées, conformément à [UIT-T T.123], comme éléments de sous-adresses d'extrémité appelante ou appelée dans l'opération SETUP et les sélecteurs de transport sont véhiculés comme identificateurs de point d'accès TSAP dans une demande de connexion du service X.224.

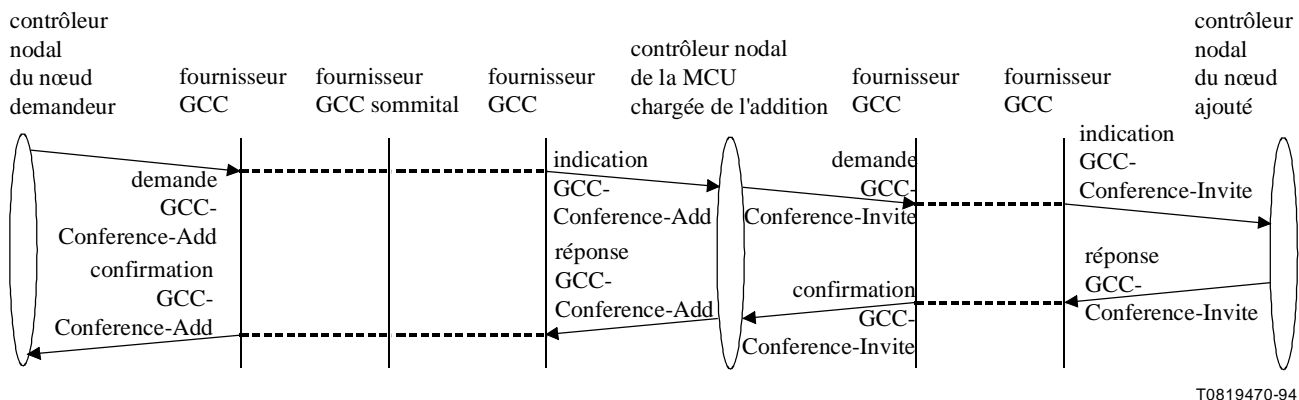
*Identificateur du nœud de la MCU chargée de l'addition*: paramètre optionnel pouvant être utilisé pour indiquer le nœud MCU particulier à partir duquel le nœud appelé sera ajouté. Si ce paramètre n'est pas fourni, le nœud appelé peut être ajouté à partir de toute MCU disponible dans la conférence. L'indication provient dans ce cas du fournisseur GCC sommital. Le contrôleur nodal de cette MCU peut soit fournir localement le service, soit émettre une autre demande vers une MCU spécifique.

*Identificateur du nœud demandeur*: identificateur du nœud demandeur.

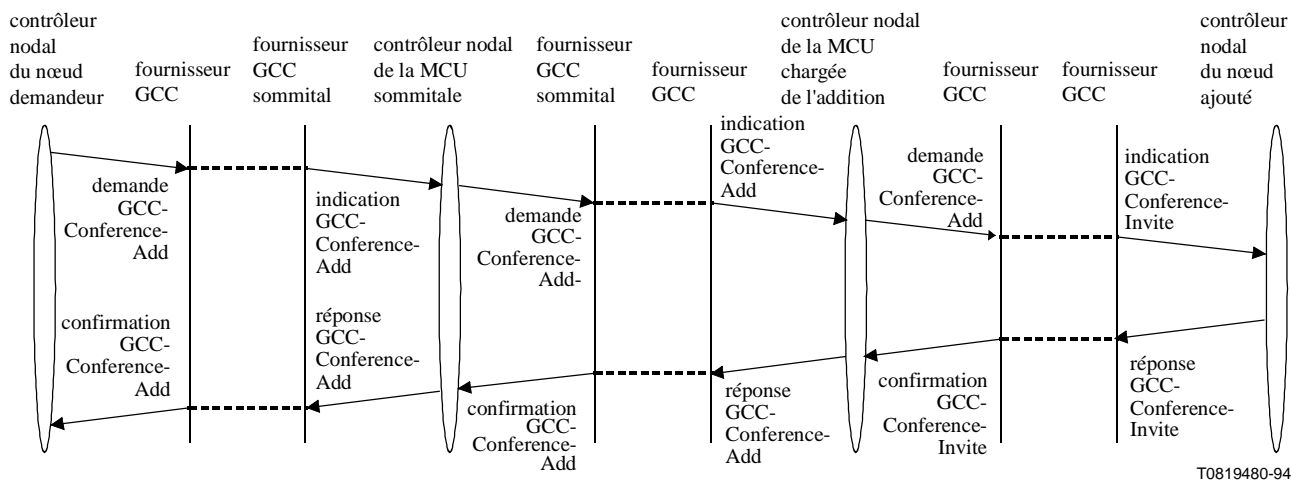
*Catégorie nodale*: ce paramètre indique au nœud invité la catégorie nodale dans laquelle il s'inscrit. Le nœud invité n'a pas d'influence sur cette sélection. L'invitation doit échouer si un nœud conforme à un protocole plus ancien, non sensible aux catégories nodales, est invité. Si cela se produit, le résultat doit indiquer que le paramètre *Catégorie nodale* était inacceptable.

*Données utilisateur:* données utilisateur optionnelles pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation, par exemple l'authentification, la taxation, etc.

*Résultat:* indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "MCU chargée de l'addition non valide", "nœud non organisateur ou non désigné par l'organisateur", "type de réseau non valide", "adresse réseau non valide", "nœud ajouté occupé", "réseau occupé", "échec de la connexion" ou "pas d'accès disponible".



**Figure 7-5 – Séquence de primitives – GCC-Conference-Add (cas direct)**



**Figure 7-6 – Séquence de primitives – GCC-Conference-Add (cas indirect)**

### 7.1.2.5.1 Élément d'adresse réseau

La structure de l'élément d'adresse réseau est présentée dans le Tableau 7-7.

**Tableau 7-7 – Structure d'un élément d'adresse réseau**

Paramètre	Description
Connexion réseau	Ce paramètre se compose soit d'une description de connexion unique soit d'une description de connexion formée par agrégation de circuits numériques RNIS ou RDCC. Ce paramètre est décrit en détail aux points 1) et 2) ci-dessous.

**Tableau 7-7 – Structure d'un élément d'adresse réseau**

Paramètre	Description
Profils	Ce paramètre est une liste d'un ou de plusieurs profils qui peuvent être exploités sur la connexion. Chaque profil peut décrire un mode de transfert de base comme la téléphonie simple ou des suites protocolaires plus complexes, comme H.320/T.120. Les valeurs possibles sont: "parole", "téléphonie à 3 kHz", "téléphonie à 7 kHz", "bande vocale", "relais de trames", "profil de base T.123 pour RDCP/RTGC", "profil de base T.123 pour RDCP", "profil de base T.123 pour RNIS-LB", "H.310", "H.320", "H.321", "H.322", "H.323", "H.324", "H.324m", "ASVD", "DSVD", "profil de téléchargement DSM-CC" et "non normalisé". Pour les profils multimédias (H.32x, ASVD et DSVD), un indicateur booléen est associé pour signaler si les suites protocolaires T.120 doivent être exploitées pour la portion données du multiplex. Les structures exactes des profils non normalisés sont définies par l'utilisateur.
Médias concernés	Liste de trois indicateurs booléens correspondant aux médias (c'est-à-dire audio, vidéo et données) qui sont concernés par la connexion.

1) *Connexions uniques*

Lorsque le paramètre *Connexion réseau* désigne une connexion unique, celle-ci peut prendre différentes formes selon qu'elle est ou non décrite avec précision et, si la description est précise, selon le type de réseau auquel le nœud ajouté est connecté. Les réseaux pris en compte dans la présente version de la Recommandation sont les suivants: RTGC, RNIS, RDCC, RDCP et ATM.

1a) Connexions RTGC:

Pour les réseaux de type RTGC, le paramètre *Connexion réseau* contient seulement une valeur *Adresse réseau de type E.164 étendu*, où le paramètre facultatif *Sous-adresse* n'est pas applicable. La structure d'une adresse réseau de type E.164 étendu est représentée dans le Tableau 7-8-1 ci-dessous.

**Tableau 7-8-1 (voir la Note) – Structure d'une adresse réseau de type E.164 étendu**

Paramètre	Description
Numéro international	Chaîne de caractères numériques, de longueur inférieure ou égale à 16 chiffres, qui représente le numéro international complet du nœud à ajouter.
Sous-adresse (optionnel)	Paramètre facultatif, valable dans le seul cas des modes de transfert RNIS, qui représente la sous-adresse RNIS du nœud à ajouter. Il s'agit d'une chaîne de caractères numériques, de longueur inférieure ou égale à 40 chiffres.
Chaîne de numérotation supplémentaire (optionnel)	Ce paramètre facultatif indique qu'une information additionnelle est requise pour atteindre le nœud à ajouter une fois la connexion physique établie. Dans le cas d'une connexion en capacité support "parole" ou "données en bande vocale" par exemple, cette information peut représenter les tonalités DTMF à transmettre en supraphonie une fois cette connexion établie. En variante, la numérotation supplémentaire peut correspondre à un numéro de réseau virtuel privé. Il s'agit d'une chaîne comportant jusqu'à 255 caractères qui sont soit des chiffres de 1 à 9, le caractère "#", le caractère "*", ou le caractère "," (virgule), celui-ci devant représenter un délai d'une seconde que l'unité MCU ajoutante doit insérer avant les caractères subséquents.

NOTE – Les Tableaux 7-7-2 à 7-7-6 de la T.124 (1998) ont été renumérotés Tableaux 7-8-1 à 7-8-5 dans la version 2007 de la présente Recommandation.

1b) Connexions RNIS:

Dans les réseaux de type RNIS, le paramètre *Connexion réseau* est structuré en deux éléments, dont le premier est une liste de types de circuit que le nœud ajouté et son réseau d'accès sont en mesure d'exploiter, le deuxième élément étant une adresse réseau de type E.164 étendu qui doit être composée pour atteindre le nœud ajouté. Le Tableau 7-8-2 montre la structure d'un descripteur de connexion RNIS unique.

**Tableau 7-8-2 (voir la Note) – Structure d'une connexion unique – Cas des réseaux de type RNIS**

Paramètre	Description
Types de circuit	Liste du ou des types de circuit pouvant être commutés pour atteindre le nœud ajouté. Les types possibles correspondent aux codes binaires de l'octet 4 de l'élément d'information <i>Capacité support</i> de la Rec. UIT-T Q.931, soit: " <i>canal numérique à 64 k</i> " (canal B), " <i>canal numérique à 2 x 64 k</i> ", " <i>canal numérique à 384 k</i> " (canal H0), " <i>canal numérique à 1536 k</i> " (canal H11), " <i>canal numérique à 1920 k</i> " (canal H12), et " <i>canaux à débit multiple de 64 k</i> ". Pour cette dernière option, une valeur d'entier est indiquée (le <i>multiplicateur</i> ), qui précise le débit effectif. Si plus d'un type de circuit est transmis, le choix est laissé à la discrétion de l'unité MCU ajoutante.
Adresse réseau de type E.164 étendu	Ce paramètre a le format décrit dans le Tableau 7-8-1.
Information de compatibilité de couche supérieure	Paramètre facultatif qui indique le mode de fonctionnement utilisé par cette partie de la connexion. Cette information est requise pour les connexions établies dans certains pays. Les modes de fonctionnement sont un ou plusieurs des suivants: "téléphonie à 3 kHz", "téléphonie à 7 kHz", "vidéophonie", "vidéoconférence", "audiographie", "audiovisuel" ou "multimédia". Si plusieurs de ces options sont choisies, cela signifie que l'utilisation d'un des modes indiqués est laissé à la discrétion de l'unité MCU ajoutante. Les définitions de ces choix peuvent dépendre du pays dans lequel est exploité l'unité MCU ajoutante. Les séquences binaires définies pour ce paramètre correspondent à celles qui ont été définies pour l'élément d'information HLC décrit dans le protocole de signalisation RNIS (Rec. UIT-T Q.931).

1c) Connexions RDCC:

Dans les réseaux de type RDCC, le paramètre *Connexion réseau* est structuré en deux éléments, dont le premier est une liste de types de circuit que le nœud ajouté et son réseau d'accès sont en mesure d'exploiter, le deuxième élément étant une adresse réseau de type E.164 étendu qui doit être composée pour atteindre le nœud ajouté. Le Tableau 7-8-3 montre la structure d'un descripteur de connexion RDCC unique.

**Tableau 7-8-3 (voir la Note)– Structure d'une connexion unique –  
Cas des réseaux de type RDCC**

Paramètre	Description
Types de circuit	Liste d'un ou deux types de circuit pouvant être commutés pour atteindre le nœud ajouté. Les types possibles sont " <i>canal numérique à 56 k</i> " et " <i>canal numérique à 64 k</i> ". Si les deux types sont transmis, le choix est laissé à la discrétion de l'unité MCU ajoutante.
Adresse réseau de type E.164 étendu	Ce paramètre a le format décrit dans le Tableau 7-8-1.

1d) Connexions ATM:

Dans les réseaux de type ATM, le paramètre *Connexion réseau* est structuré en deux éléments, dont le premier est soit une adresse réseau de type E.164 étendu, une adresse de point NSAP ou une adresse de format non normalisé qui est censée être interprétée par l'unité MCU ajoutante et dont le deuxième élément est une valeur d'entier facultative donnant le débit utile maximal possible vers le nœud ajouté, exprimé en cellules ATM par seconde.

NOTE – Certains réseaux ATM privés utilisent le format d'adresse E.164 normal, associé à un point NSAP pour atteindre les extrémités. Dans ce cas, le paramètre facultatif *Chaîne de numérotation supplémentaire* peut être utilisée pour acheminer le code du point NSAP.

1e) Connexions RDCP:

Dans les réseaux de type RDCP, le paramètre *Connexion réseau* se compose simplement soit de l'adresse réseau de type E.164 étendu qui est décrite au Tableau 7-8-1, soit d'un *Descripteur d'adresse de transport*, ou d'un format d'adresse non normalisé qui est censé être interprétable par l'unité MCU ajoutante. Le Tableau 7-8-4 montre la structure d'un descripteur d'adresse de transport.

**Tableau 7-8-4 (voir la Note) – Structure d'un descripteur d'adresse de transport**

Paramètre	Description
Adresse de point NSAP	Chaîne d'octets de longueur inférieure ou égale à 20 octets qui contient le codage binaire préféré (conformément au A.8.3.1/X.213) de l'adresse du point d'accès aux services de couche Réseau (NSAP) du nœud à ajouter.
Sélecteur de transport (optionnel)	Paramètre facultatif qui peut être utilisé pour sélectionner le point d'accès aux services de couche Réseau du nœud à ajouter.

1f) Connexions non décrites:

Dans ce cas, le paramètre *Connexion réseau* se compose soit de l'adresse réseau de type E.164 étendu qui est décrite dans le Tableau 7-8-1, soit d'un *Descripteur d'adresse de transport* ou d'un format d'adresse non normalisé qui est censé être interprétable par l'unité MCU ajoutante.

2) *Agrégation de circuits numériques RNIS ou RDCC*

Lorsque le paramètre *Connexion réseau* du Tableau 7-7 désigne des canaux RDCC ou RNIS agrégés, il est structuré en deux éléments, le premier étant une liste des circuits numériques qui doivent être agrégés ensemble, le second étant une liste facultative des algorithmes d'agrégation de canaux utilisables pour agréger les circuits. Le Tableau 7-8-5 précise cette structure.

**Tableau 7-8-5 (voir la Note) – Structure d'une connexion à agrégation de canaux**

Paramètre	Description
Connexions agrégées	Liste d'un ou de plusieurs éléments ayant chacun l'une des structures décrites dans les Tableaux 7-8-2 et 7-8-3. Lorsqu'un ou plusieurs des éléments de la liste indique plusieurs types de circuit, le choix de la combinaison finale est laissé à l'unité MCU ajoutante.
Méthode d'agrégation (optionnel)	Liste d'un ou de plusieurs algorithmes normalisés ou non normalisés d'agrégation de canaux qui peuvent être utilisés pour agréger ensemble des circuits numériques. Les algorithmes possibles sont les suivants: "H.221", "H.244" et "ISO/CEI 13871". Les algorithmes non normalisés sont censés être interprétables par l'unité MCU ajoutante. Ce paramètre est facultatif. S'il n'est pas fourni, le choix est laissé aux soins de l'unité MCU ajoutante.  NOTE – L'algorithme à retenir est implicite pour certains profils qui peuvent être énumérés par le paramètre <i>Profils</i> du Tableau 7-7 (par exemple Rec. UIT-T H.320).

#### 7.1.2.6 GCC-Conference-Lock (verrouillage de conférence)

La primitive de demande GCC-Conference-Lock peut être utilisée par un contrôleur nodal pour verrouiller une conférence, empêchant ainsi d'autres nœuds de se connecter par numérotation à la conférence. Cette primitive n'est valide que si elle est émise par l'organisateur ou par un nœud désigné par l'organisateur. Pendant qu'une conférence se trouve dans l'état verrouillé, des participants ne peuvent y être ajoutés qu'au moyen de la primitive GCC-Conference-Invite (invitation à une conférence) (ou, indirectement, au moyen d'une primitive GCC-Conference-Add). L'ordre chronologique dans lequel sont échangées des primitives GCC-Conference-Lock et GCC-Conference-Unlock (déverrouillage de conférence) entre un nœud et le fournisseur GCC sommital est préservé. Le Tableau 7-9 donne les paramètres et les types de primitives de ce service. La Figure 7-7 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-9 – Types de primitives et paramètres – GCC-Conference-Lock**

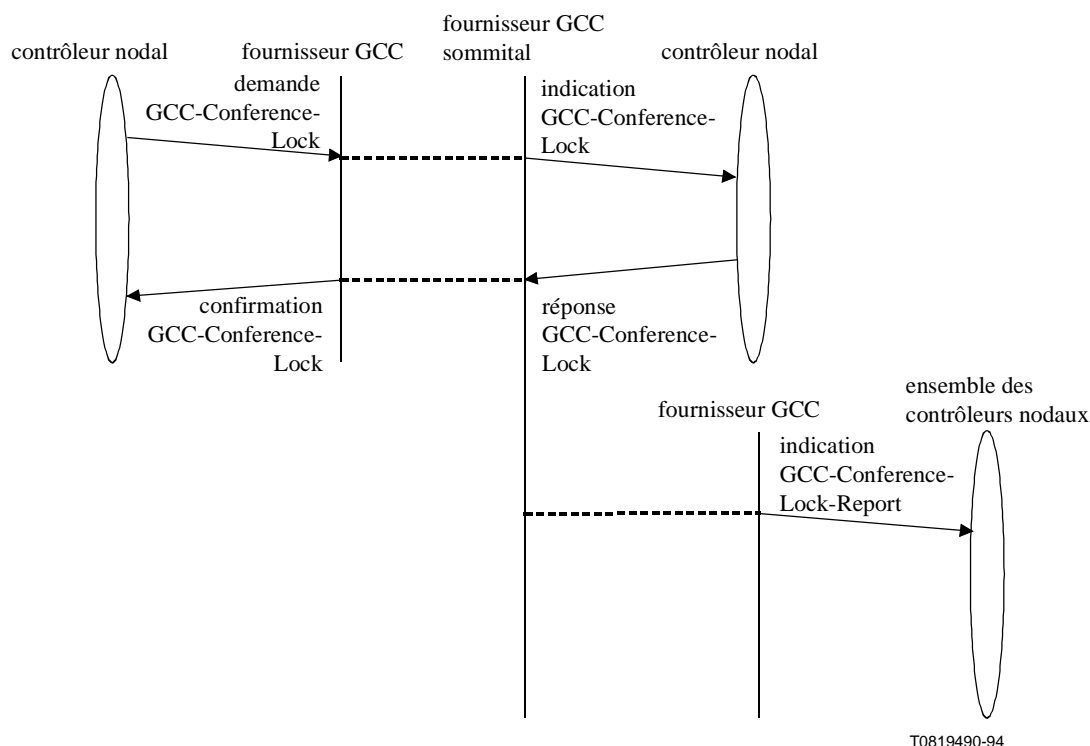
Paramètre	Demande	Indication	Réponse	Confirmation
Identificateur de conférence	M	M	M(=IN)	M(=RQ)
Identificateur du nœud origine		M	M(=)	
Résultat			M	M(=)

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Identificateur du nœud origine*: identificateur du nœud demandeur.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "fonction non disponible", "déjà verrouillée" ou "pas d'organisateur ou de nœud désigné par l'organisateur".





T0819490-94

**Figure 7-7 – Séquence de primitives – GCC-Conference-Lock**

### 7.1.2.7 GCC-Conference-Unlock (déverrouillage de conférence)

La primitive de demande GCC-Conference-Unlock peut être utilisée par un contrôleur nodal pour déverrouiller une conférence précédemment verrouillée. Le déverrouillage permet à d'autres nœuds d'entrer dans la conférence par numérotation. Si la conférence a été créée avec un mot de passe, le mot de passe reste exigé de tous les participants qui tentent d'entrer dans la conférence déverrouillée. Cette primitive n'est valide que si elle est émise par l'organisateur ou un nœud désigné par l'organisateur. L'ordre chronologique dans lequel sont échangées entre un nœud et le fournisseur GCC sommital des primitives GCC-Conference-Lock (verrouillage de conférence) et GCC-Conference-Unlock est préservé. Le Tableau 7-10 donne les paramètres et les types de primitives de ce service. La Figure 7-8 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

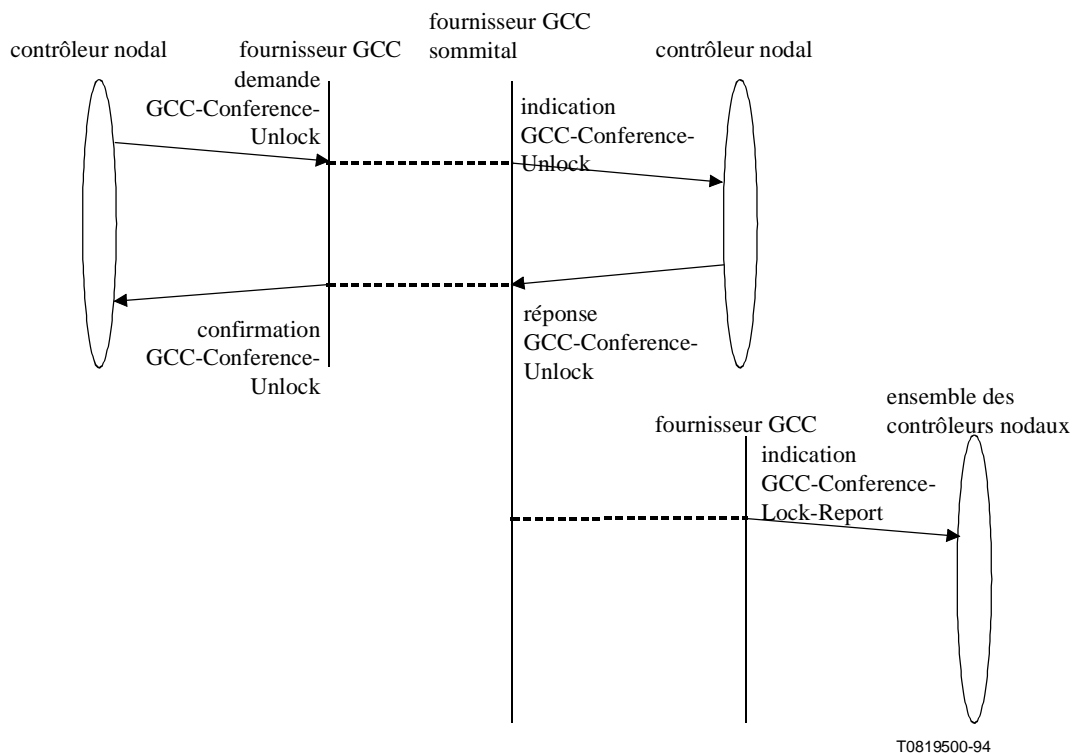
**Tableau 7-10 – GCC-Conference-Unlock – Types de primitives et paramètres**

Paramètre	Demande	Indication	Réponse	Confirmation
Identificateur de conférence	M	M	M(=IN)	M(=RQ)
Identificateur du nœud origine		M	M(=)	
Résultat			M	M(=)

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Identificateur du nœud origine*: identificateur du nœud demandeur.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "fonction non disponible", "déjà déverrouillée" ou "pas d'organisateur ou de nœud désigné par l'organisateur".



**Figure 7-8 – Séquence de primitives – GCC-Conference-Unlock**

#### 7.1.2.8 GCC-Conference-Lock-Report (compte rendu de verrouillage de conférence)

La primitive d'indication GCC-Conference-Lock-Report est envoyée par le contrôleur nodal à tous les nœuds d'une conférence à la suite de la réussite d'une primitive de demande GCC-Conference-Lock (verrouillage de conférence) ou d'une primitive de demande GCC-Conference-Unlock (déverrouillage de conférence). Les Figures 7-7 et 7-8 indiquent la séquence d'événements conduisant à l'utilisation de cette primitive. Voir également le Tableau 7-11.

**Tableau 7-11 – Types de primitives et paramètres – GCC-Conference-Lock-Report**

Paramètre	Indication
Identificateur de conférence	M
Indicateur de verrouillage	M

*Identificateur de conférence*: identificateur du domaine MCS correspondant à la conférence verrouillée.

*Indicateur de verrouillage*: indicateur signalant si la conférence est passée en mode verrouillé ou déverrouillé.

#### 7.1.2.9 GCC-Conference-Disconnect (déconnexion d'une conférence)

La primitive de demande GCC-Conference-Disconnect est utilisée par un contrôleur nodal pour se déconnecter d'une conférence. La déconnexion d'une conférence n'implique pas la déconnexion de la liaison physique. Un terminal peut entrer dans une autre conférence une fois qu'il s'est déconnecté d'une conférence. Si le service GCC détecte qu'un nœud a été déconnecté d'une conférence d'une manière anormale, par exemple si la connexion physique a été rompue, il enverra une primitive d'indication GCC-Conference-Disconnect à tous les nœuds restants de la conférence. Le Tableau 7-12 donne les paramètres et les types de primitives de ce service. Les Figures 7-9 et 7-13 indiquent la séquence d'événements correspondant à l'utilisation de cette primitive dans le cas de

lancement d'une déconnexion par un client. La Figure 7-11 montre le cas de lancement d'une déconnexion par le service GCC (correspondant à une déconnexion anormale).

**Tableau 7-12 – GCC-Conference-Disconnect – Types de primitives et paramètres**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Motif de la déconnexion		M	
Identificateur du nœud qui se déconnecte		M	
Résultat			M

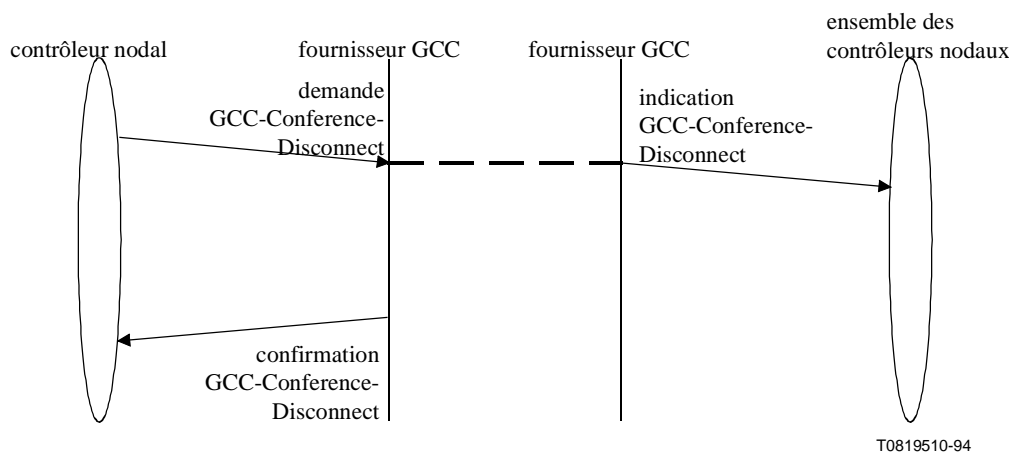
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Motif de la déconnexion*: indication du motif de la déconnexion de la conférence: soit lancée par l'utilisateur, nœud éjecté ou inconnu.

*Identificateur du nœud qui se déconnecte*: identificateur de nœud correspondant au nœud déconnecté.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".

NOTE – S'il est nécessaire de permettre à un contrôleur nodal de quitter une conférence avant la fin de l'établissement de la connexion, par exemple avant la réception d'une primitive de confirmation GCC-Conference-Join, il faudra fournir à cet effet un moyen local permettant d'établir la correspondance entre la conférence en cours de déconnexion et la conférence en cours d'établissement. Ceci est nécessaire parce que l'identificateur de la conférence n'est pas connu du contrôleur nodal avant que celui-ci ait créé la conférence ou y soit entré. Le procédé utilisé à cet effet est un sujet local qui ne fait pas partie du domaine d'application de la présente Recommandation.



**Figure 7-9 – Séquence de primitives – GCC-Conference-Disconnect (lancement par le client)**

#### 7.1.2.10 GCC-Conference-Terminate (fin de conférence)

La primitive de demande GCC-Conference-Terminate est utilisée par un contrôleur nodal pour mettre fin à une conférence dans sa totalité. La fin d'une conférence n'implique pas nécessairement la fin de la connexion physique. Cette primitive ne sera émise que par l'organisateur ou un nœud désigné par l'organisateur. Le Tableau 7-13 donne les paramètres et les types de primitives de ce service. Les Figures 7-10, 7-11 et 7-12 indiquent la séquence d'événements correspondant à l'utilisation de cette primitive dans le cas de fin lancée par un client et par le service GCC. Le cas de lancement par le service GCC peut résulter soit d'une fin anormale dans les couches inférieures de

protocole ou d'une action du fournisseur GCC sommital lorsque la conférence a été créée avec une fin automatique et que tous les autres nœuds sont sortis de la conférence.

NOTE – Dans la plupart des cas, la réception de la primitive d'indication GCC-Conference-Terminate implique que la totalité de la conférence s'est terminée, c'est-à-dire que tous les participants ont été déconnectés. Dans le cas où une fin anormale est donnée dans le code motif, cette primitive signifie que le nœud local a été retiré de la conférence d'une manière inattendue, mais n'implique pas nécessairement la fin de la totalité de la conférence.

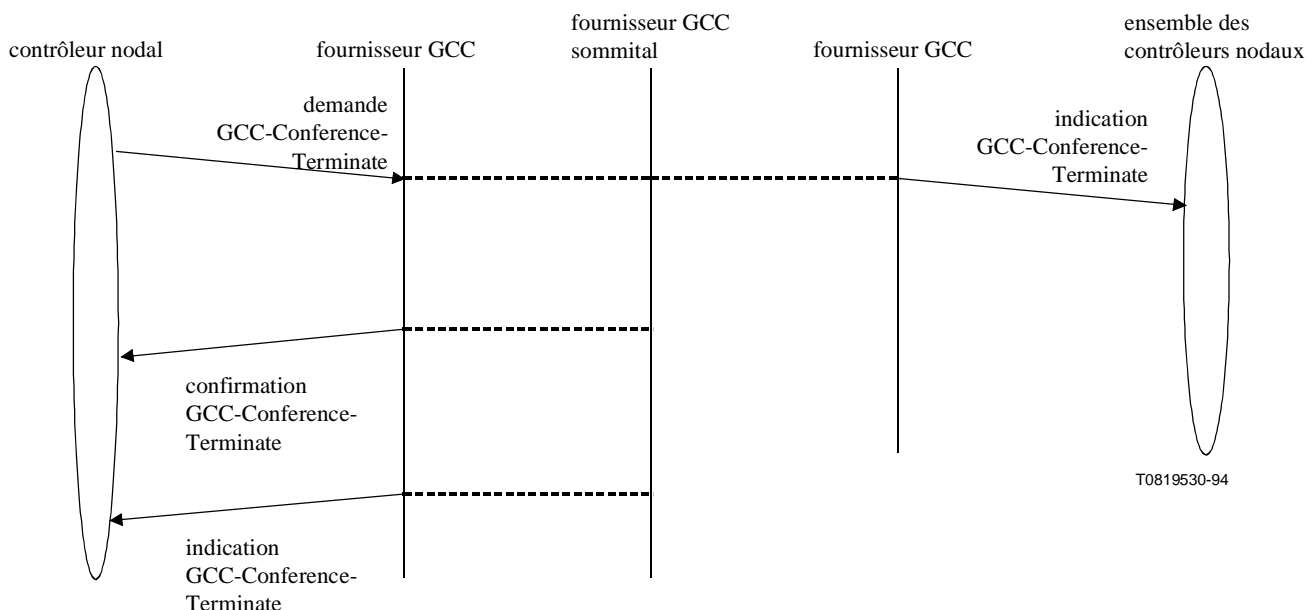
**Tableau 7-13 – Types de primitives et paramètres – GCC-Conference-Terminate**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Motif de la fin	O	O(=)	
Résultat			M

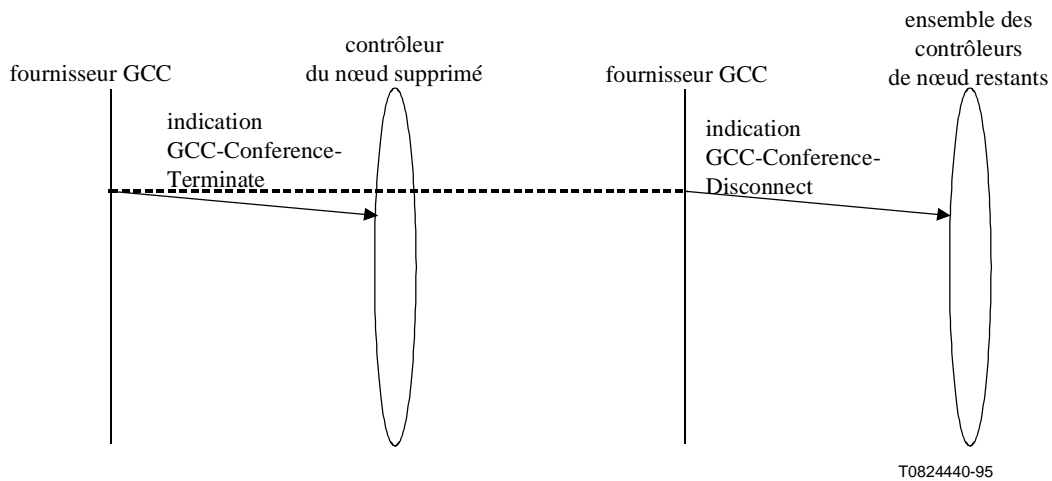
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Motif de la fin*: indication du motif de la fin de la conférence. Elle contient un des motifs possibles suivants: "demande de fin normale", "demande de fin de conférence limitée dans le temps", "plus de participants dans une conférence à fin automatique" ou "fin sur erreur".

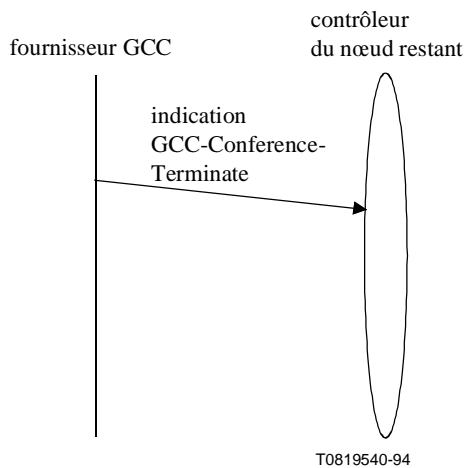
*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "pas d'organisateur ou de nœud désigné par l'organisateur".



**Figure 7-10 – Séquence de primitives – GCC-Conference-Terminate (lancée par un client)**



**Figure 7-11 – Séquence de primitives – GCC-Conference-Terminate (fin sur erreur)**



**Figure 7-12 – Séquence de primitives – GCC-Conference-Terminate (fin automatique)**

#### 7.1.2.11 GCC-Conference-Eject-User (éjection d'un utilisateur)

La primitive de demande GCC-Conference-Eject-User est utilisée par un contrôleur nodal pour forcer un nœud donné à se déconnecter de la conférence. Cette primitive sera utilisée uniquement par l'organisateur de la conférence, par un nœud désigné par l'organisateur ou par un nœud situé directement au-dessus du nœud à éjecter dans la hiérarchie de connexion. L'éjection d'une conférence n'implique pas la fin de la connexion physique correspondante. Lorsqu'un nœud est éjecté, une primitive d'indication GCC-Conference-Disconnect (déconnexion de conférence) est envoyée par le contrôleur nodal à tous les nœuds restants de la conférence pour indiquer que le nœud a été déconnecté à la suite d'une éjection. Le Tableau 7-14 donne les paramètres et les types de primitives de ce service. La Figure 7-13 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-14 – Types de primitives et paramètres – GCC-Conference-Eject-User**

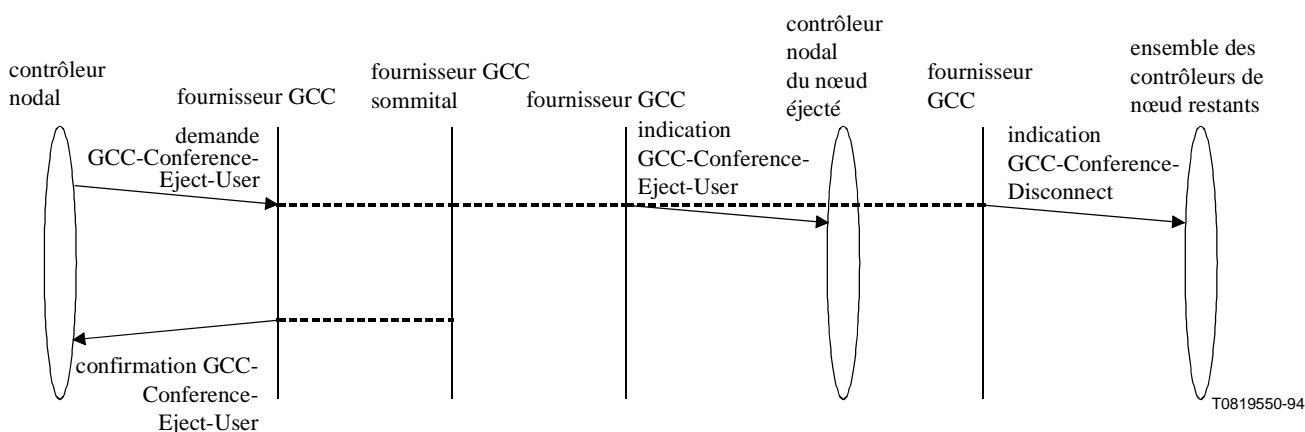
Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Identificateur de nœud éjecté	M	M(=)	M(=)
Motif de l'éjection	O	O(=)	
Résultat			M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Identificateur de nœud éjecté*: nœud à éjecter.

*Motif de l'éjection*: indication du motif de l'éjection: "initié par l'utilisateur", "nœud supérieur déconnecté" ou "nœud supérieur éjecté".

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "utilisateur à éjecter non valide" ou "pas d'organisateur ou de nœud désigné par l'organisateur".



**Figure 7-13 – Séquence de primitives – GCC-Conference-Eject-User (lancée par le client)**

### 7.1.2.12 GCC-Conference-Transfer (transfert de conférence)

La primitive de demande GCC-Conference-Transfer est utilisée par un contrôleur nodal pour déconnecter des nœuds sélectionnés au sein d'une conférence et les faire entrer dans une autre conférence. Cette primitive sera utilisée uniquement par l'organisateur de la conférence ou par un nœud désigné par l'organisateur. Certaines des unités MCU d'une conférence peuvent déjà, avant le transfert, participer à la fois à la conférence d'origine et à la conférence destinataire. Dans ce cas, ces MCU ne doivent pas figurer dans la liste des nœuds transférants de la demande. Si une unité MCU ne participe pas aux deux conférences mais se retrouvera connectée à des nœuds qui feront partie des deux conférences après le transfert (si tous les nœuds ne sont pas transférés), alors cette MCU devra faire son entrée dans la conférence destinataire avant l'émission de la primitive de demande GCC-Conference-Transfer. Toute MCU figurant dans la liste des nœuds transférants (ceux qui doivent effectuer un transfert) mènera à bien l'opération de transfert (déconnexion de la conférence d'origine et entrée dans la conférence destinataire) avant de traiter toute nouvelle primitive d'indication GCC-Conference-Join (entrée dans une conférence). Cela garantit le déroulement correct de la demande de transfert des nœuds subordonnés à l'unité MCU. S'ils ne l'ont pas déjà fait, les nœuds qui sont en cours de transfert et de réception d'une primitive d'indication GCC-Conference-Terminate (fin de conférence) de la conférence originale peuvent entrer

directement dans la conférence destinataire sans avoir à se déconnecter. Cette situation peut se présenter si l'unité MCU à laquelle le nœud est connecté est également chargée du transfert. Le Tableau 7-15 donne les paramètres et les types de primitives de ce service. La Figure 7-14 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-15 – Types de primitives et paramètres – GCC-Conference-Transfer**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Nom de conférence destinataire	M	M(=)	M(=)
Modificateur de nom de conférence destinataire	O	O(=)	O(=)
Adresse de réseau destinataire	O	O(=)	
Nœuds chargés du transfert (Liste d'identificateurs de nœud ou vide)	O		O(=)
Mot de passe	C	C(=)	
Résultat			M

*Identificateur de conférence*: identificateur du domaine MCS correspondant à la conférence dont font partie, avant l'opération de transfert, les nœuds désignés.

*Nom de conférence destinataire*: nom de la conférence dans laquelle doivent entrer les nœuds désignés. C'est le nom sous lequel la conférence est connue de l'unité MCU – ou des unités MCU – auxquelles les nœuds transférants sont connectés. Si la conférence a été créée au moyen des deux formes numériques et alphabétiques du nom de conférence, l'une ou l'autre forme peut être employée dans cette primitive.

*Modificateur de nom de conférence destinataire*: si une unité MCU de la conférence destinataire a choisi (à la suite d'un problème d'unicité des noms locaux) d'utiliser le modificateur de nom de conférence facultatif, les nœuds à transférer qui sont connectés à cette MCU doivent être transférés par une primitive GCC-Conference-Transfer distincte. Un transfert séparé est fait pour ceux des nœuds qui sont connectés à des unités MCU utilisant le nom de base de la conférence non modifié ou un nom modifié d'une façon différente. La manière dont le nœud demandeur prend connaissance des modificateurs de nom de conférence des diverses MCU est considérée comme ne faisant pas partie du domaine d'application de la présente Recommandation.

*Adresse de réseau destinataire*: ce paramètre facultatif décrit une connexion logique nominale que chaque nœud transféré doit utiliser pour entrer dans la conférence destinataire s'il n'est pas directement connecté à une unité MCU déjà entrée dans la conférence destinataire (ou hébergeant celle-ci). Chaque segment de la connexion logique est décrit en termes de type de réseau, d'adresse réseau, de débit utile le cas échéant et, facultativement, de profil multimédia devant être exploité, d'algorithme d'agrégation de circuits numériques si applicable, et de liste de médias (par exemple audio, vidéo, données) visés par ce segment. Les informations applicables (c'est-à-dire les éléments d'adresse réseau) doivent être transmises sous forme de paramètre *Adresse appelée* dans la primitive de demande GCC-Conference-Join qui doit être émise par un nœud transférant. Dans le protocole de commande GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont la description et l'utilisation se trouvent dans l'Annexe B.

NOTE – Ce modèle part du principe qu'une connexion logique nominale est disponible pour tous les nœuds transférants. Cette hypothèse n'est pas toujours valide. Par exemple, dans le cas d'une conférence à 2 canaux B du RNIS/type H.320, ce modèle implique que le même numéro E.164 (ou la même paire de numéros E.164) peut être utilisé par tous les nœuds effectuant un transfert vers la conférence destinataire. Si le RNIS ne fournit pas le paramètre *Numéro de l'appelant* à l'extrémité appelée, le pont MCU connecté n'a

aucun moyen pour appairer les canaux B individuels. L'indication nœud transférant par nœud transférant d'une adresse réseau destinataire fera l'objet d'un complément d'étude.

*Nœuds transférants (liste d'identificateurs de nœud ou néant)*: liste d'identificateurs de nœud identifiant des fournisseurs GCC destinataires de l'indication. L'omission de ce paramètre indique que tous les nœuds de la conférence désignée par l'identificateur de conférence sont destinataires.

*Mot de passe*: paramètre indiquant le mot de passe que les nœuds chargés du transfert utiliseront dans la primitive de demande GCC-Conference-Join pour entrer dans la conférence. Il s'agit d'une chaîne numérique ou d'une chaîne de caractères unicode rangée 00 facultative, d'une taille maximale de 255 chiffres ou caractères. Le contrôleur nodal doit déterminer s'il convient d'envoyer le nom de conférence sous forme d'une chaîne alphabétique ou d'une chaîne numérique, selon sa valeur. Une valeur composée uniquement de caractères numériques doit être spécifiée comme étant une chaîne numérique, alors qu'une valeur comportant au moins un caractère non numérique doit être spécifiée sous forme de chaîne alphabétique. Ce paramètre ne sera utilisé que si l'indicateur "mot de passe en clair" est positionné pour cette conférence.

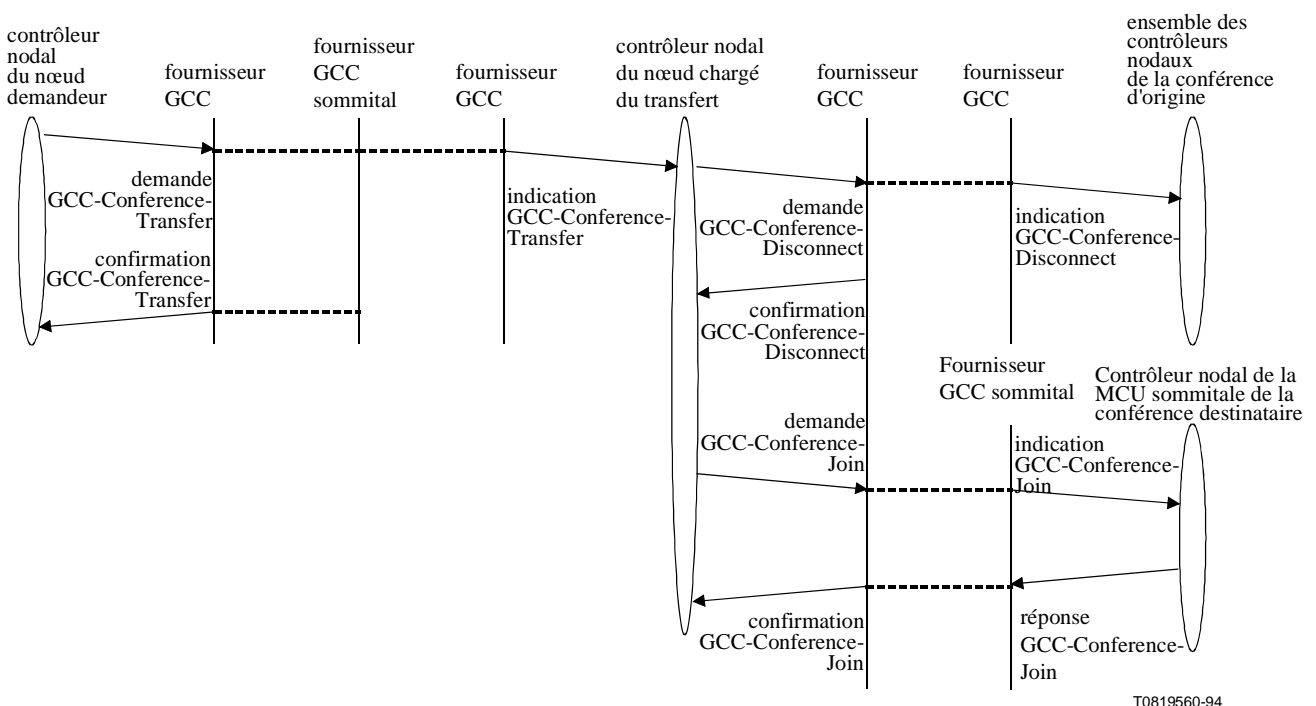


Figure 7-14 – Séquence de primitives – GCC-Conference-Transfer

### 7.1.3 Exigences pour l'établissement de conférence

Les exigences suivantes sont définies en ce qui concerne l'établissement physique d'une connexion, c'est-à-dire d'une connexion sous-jacente entre nœuds utilisant les cas de connexion RTPC, RNIS ou RDCC de la Recommandation T.123. Leur but est d'éviter la possibilité d'une situation de blocage mutuel dans laquelle les deux extrémités d'une connexion physique échouent dans l'établissement d'une connexion, chacune d'elles attendant que l'autre prenne l'initiative de la procédure d'établissement de la connexion. Ces exigences peuvent être remplacées par un accord bilatéral entre les nœuds concernés pouvant être défini *a priori* ou par un échange d'informations défini en dehors du domaine d'application de la présente Recommandation.

- Il doit d'abord être établi par les deux extrémités de la connexion physique quel est le nœud appelant et quel est le nœud appelé. Il doit également être établi quel est le type de chacun des nœuds (terminal, terminal multiaccès ou MCU). Cette information peut être obtenue au moyen de la primitive GCC-Conference-Query ou être connue *a priori*. La procédure de



rupture de la symétrie de la primitive GCC-Conference-Query est utilisée pour choisir d'une manière arbitraire un nœud appelant pour la procédure d'établissement de la connexion si aucun des deux nœuds ne connaît son statut d'appelant ou d'appelé.

- Le nœud appelant sera responsable du lancement de la procédure d'établissement de la conférence initiale pouvant être une demande de création d'une nouvelle conférence, l'entrée dans une conférence sur le nœud appelé ou l'invitation du nœud appelé dans une conférence. Il est à noter que cela n'empêche pas le nœud appelé de prendre des mesures pour établir également une conférence sur la même connexion physique, toutefois des précautions doivent être prises dans ce cas afin d'éviter d'interférer avec les actions du nœud appelant. Une unité MCU appelée peut activer l'indicateur d'invitation dans la réponse GCC-Conference-Query et ainsi retirer au nœud appelant la responsabilité du lancement d'une connexion.
- Des contraintes s'appliquent aux actions du nœud appelant lorsqu'il établit la connexion initiale entre différents types de nœuds (terminaux, terminaux multiaccès et MCU). Le Tableau 7-16 montre ces contraintes pour chaque permutation des types de nœud appelant et de nœud appelé. Les définitions des actions apparaissant dans le tableau sont les suivantes:
  - 1) le nœud appelant demande la création d'une nouvelle conférence sur le nœud appelé;
  - 2) le nœud appelant tente d'entrer dans la conférence existante sur le nœud appelé;
  - 3) le nœud appelant crée une conférence locale et invite le nœud appelé;
  - 4) le nœud appelant invite le nœud appelé dans une conférence existante.

**Tableau 7-16 – Actions du nœud appelant lors de l'établissement d'une conférence**

Nœud appelant	Nœud appelé		
	Terminal	Terminal multiaccès	MCU
Terminal	1 ou 3	1 ou 2	1 ou 2
Terminal multiaccès	3 ou 4	1, 2, 3 ou 4	1 ou 2
MCU	3 ou 4	3 ou 4	1, 2, 3 ou 4

Dans certains cas, le nœud appelant peut avoir besoin d'utiliser des informations issues du nœud appelé contenu dans la confirmation GCC-Conference-Query pour faciliter le choix entre les procédures autorisées qui sont définies dans le Tableau 7-16. Par exemple, lorsqu'un terminal à accès unique ou multiple appelle un terminal à accès multiple, il peut tirer parti de la présence ou de l'absence de conférences non verrouillées dans la liste des descripteurs de conférence afin de déterminer s'il est préférable de créer une conférence automatiquement (actions 1, 3 ou 4) ou de laisser l'utilisateur tenter d'entrer dans une conférence existante au nœud appelé (action 2). Lors de la création d'une conférence quelconque, le fait de la rendre listée et déverrouillée indique que cette conférence doit pouvoir être mise à la disposition d'un nœud appelant pour entrée. Si l'intention est de ne pas permettre qu'un appelant entre dans une conférence, celle-ci doit être créée avec les attributs "verrouillée" et/ou "non listée".

Le Tableau 7-17 définit l'ensemble des règles permettant de déterminer l'action par défaut du nœud appelant (ainsi que du nœud appelé) en fonction des types de nœud et des réglages paramétriques contenus dans la confirmation GCC-Conference-Query.

**Tableau 7-17 – Règles pour déterminer l'action par défaut aux nœuds appelés et appelants**

Type de nœud appelant	Type de nœud appelé	Conférences déverrouillées listées	Indicateur de conférence par défaut	Attente de l'indicateur d'invitation	Indicateur d'absence de conférence non listée	Action par défaut du nœud appelant	Action par défaut du nœud appelé
Terminal	Terminal	*	*	*	*	Invitation par appelant	Attente de l'appelant
Term. multiaccès	Terminal	*	*	*	*	Invitation par appelant	Attente de l'appelant
MCU	Terminal	*	*	*	*	Invitation par appelant	Attente de l'appelant
Terminal	Term. multiaccès	FAUX	*	*	*	Création distante par appelant	Attente de l'appelant
Terminal	Term. multiaccès	VRAI	ABSENT ou FAUX pour toutes conférences	*	ABSENT ou FAUX	L'appelant choisit la conférence à joindre	Attente de l'appelant
Terminal	Term. multiaccès	VRAI	ABSENT ou FAUX pour toutes conférences	*	VRAI	L'appelant choisit la conférence listée à joindre	Attente de l'appelant
Terminal	Term. multiaccès	VRAI	VRAI pour une seule conférence	*	*	L'appelant entre dans la conférence par défaut	Attente de l'appelant
Term. multiaccès	Term. multiaccès	FAUX	*	*	*	Invitation par appelant	Attente de l'appelant
Term. multiaccès	Term. multiaccès	VRAI	ABSENT ou FAUX pour toutes conférences	*	ABSENT ou FAUX	L'appelant choisit la conférence à joindre	Attente de l'appelant
Term. multiaccès	Term. multiaccès	VRAI	ABSENT ou FAUX pour toutes conférences	*	VRAI	L'appelant choisit la conférence listée à joindre	Attente de l'appelant
Term. multiaccès	Term. multiaccès	VRAI	VRAI pour une seule conférence	*	*	L'appelant entre dans la conférence par défaut	Attente de l'appelant
MCU	Term. multiaccès	*	*	*	*	Invitation par appelant	Attente de l'appelant
Terminal	MCU	FAUX	*	ABSENT ou FAUX	ABSENT ou FAUX	L'appelant choisit une conférence non listée à joindre OU l'appelant répète sa requête	Attente de l'appelant
Terminal	MCU	FAUX	*	ABSENT ou FAUX	VRAI	L'appelant répète sa requête	Attente de l'appelant

**Tableau 7-17 – Règles pour déterminer l'action par défaut aux nœuds appelés et appelants**

Type de nœud appelant	Type de nœud appelé	Conférences déverrouillées listées	Indicateur de conférence par défaut	Attente de l'indicateur d'invitation	Indicateur d'absence de conférence non listée	Action par défaut du nœud appelant	Action par défaut du nœud appelé
Terminal	MCU	VRAI	ABSENT ou FAUX pour toutes conférences	ABSENT ou FAUX	ABSENT ou FAUX	L'appelant choisit la conférence à joindre	Attente de l'appelant
Terminal	MCU	VRAI	ABSENT ou FAUX pour toutes conférences	ABSENT ou FAUX	VRAI	L'appelant choisit la conférence listée à joindre	Attente de l'appelant
Terminal	MCU	VRAI	VRAI pour une seule conférence	ABSENT ou FAUX	*	L'appelant entre dans la conférence par défaut	Attente de l'appelant
Terminal	MCU	*	*	VRAI	*	L'appelant attend l'invitation du nœud appelé	Called node invites
Term. multiaccès	MCU	FAUX	*	ABSENT ou FAUX	ABSENT ou FAUX	L'appelant choisit une conférence non listée à joindre OU l'appelant répète sa requête	Attente de l'appelant
Term. multiaccès	MCU	FAUX	*	ABSENT ou FAUX	VRAI	L'appelant répète sa requête	Attente de l'appelant
Term. multiaccès	MCU	VRAI	ABSENT ou FAUX pour toutes conférences	ABSENT ou FAUX	ABSENT ou FAUX	L'appelant choisit la conférence à joindre	Attente de l'appelant
Term. multiaccès	MCU	VRAI	ABSENT ou FAUX pour toutes conférences	ABSENT ou FAUX	VRAI	L'appelant choisit la conférence listée à joindre	Attente de l'appelant
Term. multiaccès	MCU	VRAI	VRAI pour une seule conférence	ABSENT ou FAUX	*	L'appelant entre dans la conférence par défaut	Attente de l'appelant
Term. multiaccès	MCU	*	*	VRAI	*	L'appelant attend l'invitation du nœud appelé	Called node invites
MCU	MCU	*	*	*	*	à étudier	à étudier

#### **7.1.4 Exemples de procédures d'établissement de conférence**

L'établissement de conférences peut se faire d'un certain nombre de façons et dans des conditions variées. La conférence la plus simple du type "point à point" n'implique pas de MCU dans l'appel. Dans les cas où la conférence est établie au moyen d'une ou de plusieurs MCU, un appel sera établi normalement dans le mode rendez-vous (tous les participants appellent une MCU), dans le mode conférence de départ (la MCU établit une communication de départ vers chacun des participants) ou dans le mode conférence en transit (un participant appelle une MCU, puis fait entrer d'autres participants qui sont appelés par la MCU).

##### **7.1.4.1 Etablissement d'une conférence rendez-vous**

Une conférence rendez-vous est créée au niveau d'une MCU. Les nœuds terminaux, ainsi qu'éventuellement d'autres MCU, appellent la MCU pour entrer dans la conférence. Si d'autres MCU sont entrées dans la conférence, les nœuds terminaux peuvent appeler l'une quelconque de ces MCU.

La création initiale d'une conférence rendez-vous peut être faite hors bande, initialisée par exemple par la MCU, ou bien la conférence peut être créée par le premier nœud qui appelle la MCU. Dans le premier cas, le service GCC n'est pas impliqué. Dans le second cas, la conférence est créée sur la MCU par l'émission d'une primitive de demande GCC-Conference-Create (création d'une conférence) sur la connexion du nœud organisateur vers la MCU. Dans les deux cas, le nœud sur lequel la conférence a été créée devient le fournisseur GCC sommital. Le nœud organisateur, la MCU elle-même dans le premier cas ou le nœud demandeur dans le second cas, reçoit le statut spécial d'organisateur de conférence.

Un nom de conférence est fourni quand une conférence est créée à distance en utilisant une primitive de demande GCC-Conference-Create. Si le nom est déjà utilisé dans la MCU, un modificateur de nom de conférence est attribué par le contrôleur nodal de la MCU de manière à définir un nom localement non ambigu pour la MCU. Ce modificateur doit être présent dans la demande d'entrée lorsque d'autres nœuds tentent d'entrer dans la conférence.

Quand une conférence est créée à distance, il est également possible de fournir un mot de passe d'organisateur optionnel. Cet identificateur n'est nécessaire que si l'organisateur a l'intention de se déconnecter de la conférence et de se connecter par la suite en retrouvant les privilèges d'organisateur.

Une conférence rendez-vous est généralement créée en mode non verrouillée de façon à permettre à d'autres nœuds d'y entrer. Une conférence rendez-vous peut être créée avec un mot de passe pour éviter l'entrée de nœuds indésirables.

Un nœud entrant dans une conférence rendez-vous émet une primitive de demande GCC-Conference-Join (entrée dans une conférence) sur la connexion du nœud vers l'une quelconque des MCU appartenant à la conférence. Un nœud ne connaît généralement pas le nom de la conférence dans laquelle se fait l'entrée. Le nœud peut dans ce cas, avant d'entrer dans la conférence, demander à la MCU une liste des conférences disponibles en émettant une primitive de demande GCC-Conference-Query (interrogation de conférence). La réponse est une primitive de confirmation GCC-Conference-Query indiquant le type de nœud auquel le terminal est connecté (une MCU dans ce cas) et une liste de toutes les conférences listées dont la MCU fait actuellement partie. Cette liste comprend le nom de conférence de chaque conférence, le modificateur de nom de conférence éventuel ainsi que d'autres caractéristiques de la conférence telles que la protection éventuelle par un mot de passe. Cette protection peut être employée afin d'exiger la fourniture d'un mot de passe de l'utilisateur avant qu'il ne tente d'entrer dans la conférence. Une fois que le nom de conférence est sélectionné, l'entrée dans la conférence peut se faire par l'émission d'une primitive de demande GCC-Conference-Join indiquant le nom de la conférence souhaitée ainsi que le

modificateur de nom de conférence éventuel. Si la conférence exige un mot de passe, celui-ci figure dans la demande d'entrée.

Il est à noter qu'il est normal que le cas d'une conférence rendez-vous sans connaissance préalable de la conférence dans laquelle se fait l'entrée est identique au cas d'une conférence point à point. Ceci signifie que la séquence d'événement de démarrage de la conférence ne peut être différente de ce qu'elle est dans une conférence point à point, étant donné que la connaissance préalable est la même dans les deux cas. La distinction ne peut être faite que par la présence dans la confirmation GCC-Conference-Query de l'indication que le nœud connecté directement est une MCU et qu'il existe des conférences en cours dans lesquelles il est possible d'entrer.

Dans le cas où le nom de conférence est diffusé hors bande aux nœuds entrants, un nœud peut entrer directement dans la conférence en indiquant le nom de conférence dans la primitive de demande GCC-Conference-Join sans demander au préalable quelles sont les conférences disponibles. S'il existe de multiples MCU indépendantes auxquelles les nœuds peuvent se connecter pour entrer dans une conférence, il est possible que des conflits de nom imposent l'utilisation de modificateurs de nom de conférence pour certaines MCU. Dans ce cas, un nœud entrant dans une conférence doit indiquer le modificateur de nom de conférence en usage dans la MCU à laquelle il est connecté. Comme il peut être difficile de déterminer ce modificateur sans l'utilisation de la primitive GCC-Conference-Query, il est recommandé d'utiliser cette primitive GCC-Conference-Query avant de tenter d'entrer dans une conférence rendez-vous, s'il existe une quelconque possibilité de conflit de nom. Il est également recommandé d'utiliser une description de conférence lors de la création d'une conférence rendez-vous. Dans ce cas, il est très probable que des conférences multiples ayant le même nom de conférence peuvent être différenciées par des descriptions de conférence différentes. Il est cependant en général préférable de choisir un nom de conférence non ambigu pour l'ensemble des MCU lors de la création de la conférence, ce qui évite l'utilisation d'un modificateur de nom de conférence.

#### **7.1.4.2 Etablissement d'une conférence de départ**

Lors de l'établissement d'une conférence de départ, la conférence est créée localement sur la MCU et les participants à la conférence sont appelés et invités dans la conférence par cette MCU. La conférence est généralement créée en indiquant une conférence verrouillée et peut également être non listée.

La MCU instigatrice établit ensuite les connexions physiques vers chacun des terminaux devant participer à la conférence puis invite tour à tour chaque nœud dans la conférence. Ceci se fait par l'envoi d'une primitive de demande GCC-Conference-Invite (invitation à une conférence) au nœud invité. Comme la conférence a été créée par la MCU, cette MCU est le fournisseur GCC sommital de la conférence de départ.

#### **7.1.4.3 Etablissement d'une conférence en transit**

Le cas d'une conférence en transit est très voisin de celui d'une conférence de départ, mis à part le fait que la conférence est créée par un terminal. Le terminal se connecte à une MCU et crée la conférence en utilisant une primitive de demande GCC-Conference-Create (création d'une conférence). Comme dans le cas de la conférence de départ, la conférence est généralement verrouillée et non listée. Une conférence de départ est en général créée comme devant se terminer automatiquement lorsque tous les nœuds se sont déconnectés. Un nom de conférence NULL (néant) est normalement donné pour une conférence de départ car il n'est pas nécessaire que des nœuds entrent dans la conférence sans y être invités. S'il existe déjà sur le nœud organisateur une conférence avec le nom de conférence NULL (néant), le contrôleur nodal choisit arbitrairement un nom de conférence non ambigu. Le nom n'a pas besoin d'être lisible pour un opérateur humain puisqu'il n'est pas utilisé pour entrer dans une conférence.

#### 7.1.4.4 Etablissement d'une conférence point à point

Une conférence point à point diffère des types précédents dans la mesure où elle n'implique que deux nœuds terminaux sans nécessiter la présence d'une unité MCU. Si le terminal appelant qui est à l'origine de la connexion physique et le terminal appelé sont tous deux connus, une conférence point à point peut être établie depuis le terminal appelant par l'envoi d'une demande GCC-Conference-Query (interrogation de conférence) indiquant le terminal appelé. Cela permet au nœud demandeur de déterminer si l'autre nœud est un terminal, un pont de conférence ou un terminal multiaccès sans connaissance *a priori* à ce sujet. La confirmation GCC-Conference-Query produite comme réponse à la demande indique (dans le cas d'un appel de point à point) si le nœud directement connecté est un terminal utilisateur. Une fois connu le fait que le nœud connecté directement est un terminal, la conférence peut être établie par le terminal appelant soit par l'envoi d'une primitive de demande GCC-Conference-Create (création de conférence) qui crée une nouvelle conférence, soit par la création d'une conférence locale suivie de l'invitation de l'autre terminal au moyen de l'émission d'une primitive de demande GCC-Conference-Invite. Généralement la conférence sera spécifiée avec un nom de conférence arbitraire, par exemple "0", elle sera verrouillée, non listée et se termine automatiquement.

Dans le cas où un nœud ne sait pas quel terminal joue le rôle d'appelant ou d'appelé, ce terminal doit émettre une primitive de demande GCC-Conference-Query afin de déterminer le rôle de l'autre nœud, à moins qu'il n'ait déjà reçu une primitive d'indication GCC-Conference-Query qui fournit cette information. Le paramètre "indicateur d'asymétrie" de la demande signale que le rôle du nœud local n'est pas connu. Si l'autre nœud sait dans quelle situation il se trouve, il fournit cette indication dans la confirmation GCC-Conference-Query résultante. Si aucun des nœuds ne connaît son statut d'appelant ou d'appelé, la confirmation ou le contenu d'une primitive d'indication GCC-Conference-Query reçue indique un positionnement inconnu qui est un nombre aléatoire de 32 bits utilisé pour rompre la symétrie. Le nœud qui a généré le plus petit des deux nombres aléatoires sera considéré comme le nœud appelé et ne doit pas essayer d'établir la conférence. Le nœud qui a généré le plus grand nombre sera considéré comme le nœud appelant et tentera d'établir la conférence.

Si un terminal sait *a priori* que l'appel est du type point à point entre deux terminaux, et s'il sait si il est l'appelant ou l'appelé, il n'a pas besoin d'émettre une primitive de demande GCC-Conference-Query. Au lieu de cela, s'il sait qu'il est le terminal appelant, il peut émettre immédiatement la primitive de demande GCC-Conference-Create ou GCC-Conference-Invite. S'il sait qu'il est le terminal appelé, il peut également omettre la primitive de demande GCC-Conference-Query et attendre simplement de recevoir une primitive d'indication GCC-Conference-Create ou GCC-Conference-Invite. Si le terminal distant n'a pas de connaissance *a priori* du type de la connexion, il est possible que le terminal local reçoive une primitive d'indication GCC-Conference-Query du terminal distant, à laquelle il a l'obligation de répondre.

#### 7.1.4.5 Etablissement de conférence entre terminaux multiaccès

Un terminal multiaccès est un équipement qui doit en général être traité comme un terminal, mais qui a la capacité d'établir simultanément, comme une MCU, des connexions vers de nœuds multiples. Lorsqu'un terminal ou un terminal multiaccès appelle un nœud et établit qu'il s'agit d'un terminal multiaccès (soit par une connaissance *a priori* ou par un échange de primitives GCC-Conference-Query), l'action effectuée dépend de la disponibilité éventuelle de conférences sur le terminal multiaccès et des caractéristiques de ces conférences.

Il est en général souhaitable qu'une connexion de ce type soit établie automatiquement, de la même façon qu'une conférence point à point plutôt que comme une conférence rendez-vous. Plus précisément, si à cet instant aucun des nœuds appelant et appelé ne fait partie de conférences les connectant à d'autres nœuds, l'appel devrait être traité exactement comme un appel point à point. Si le nœud appelant a déjà une conférence en cours, il est courant que ce nœud invite simplement le nouveau nœud dans la conférence existante.

Si c'est le nœud appelé qui a déjà une conférence en cours, l'action dépend de l'état verrouillé ou non, listé ou non de cette conférence. Si la conférence n'est pas listée et en absence d'information *a priori*, le nœud appelant ne sera pas prévenu de l'existence de la conférence et traitera l'appel comme s'il n'existait pas de conférence (s'il avait eu la connaissance *a priori* de l'existence de la conférence non listée, il aurait pu y entrer). Si la conférence est verrouillée, le nœud appelant n'a aucun moyen d'y entrer et traitera également l'appel comme s'il n'existait pas de conférence. Dans chacun de ces cas, une fois qu'une nouvelle conférence est établie, le nœud appelé peut choisir de transférer vers la nouvelle conférence les nœuds connectés à l'ancienne conférence. S'il existe une ou plusieurs conférences listées et non verrouillées au nœud appelé, le nœud appelant doit considérer que le nœud appelé héberge une conférence rendez-vous et qu'il faudrait normalement qu'un utilisateur choisisse la conférence dans laquelle il souhaite entrer – dans ce cas la procédure ne serait pas entièrement automatique.

S'il existe des conférences sur les deux nœuds, le nœud appelant invitera généralement le nœud appelé à entrer dans sa conférence, ou bien essaiera d'entrer dans la conférence du nœud appelé. Dans les deux cas, le terminal multiaccès pour lequel la conférence n'a pas été élargie par l'inclusion de l'autre terminal multiaccès devrait transférer tous les nœuds subordonnés vers la conférence dans laquelle il vient d'entrer (ou à laquelle il vient d'être invité).

Le transfert de nœuds vers la nouvelle conférence peut dans ce cas se faire de deux façons. La primitive GCC-Conference-Transfer peut être utilisée pour donner l'instruction à chacun des nœuds de se déconnecter de la conférence en cours et d'entrer dans la nouvelle conférence. Ceci n'est évidemment possible que si la nouvelle conférence n'est pas verrouillée et si le nœud demandeur possède le privilège de transfert pour la conférence précédente (ou en était l'organisateur). En variante, la primitive GCC-Conference-Invite pourrait être utilisée pour inviter dans la nouvelle conférence les nœuds connectés directement. Ceci sera fait en général après avoir déconnecté ces nœuds de la conférence précédente. Si des nœuds autres que ceux connectés directement étaient présents, les nœuds connectés directement ont la responsabilité d'inviter d'une manière récursive dans la nouvelle conférence les nœuds qui leur sont connectés directement. Ceci signifie que les invitations se propageront à travers la hiérarchie physique de connexion des nœuds. Quoique les nœuds devraient normalement procéder ainsi, il n'est pas garanti que cette action sera effectivement prise. Pour cette raison, l'utilisation de la primitive GCC-Conference-Transfer est un procédé plus sûr dans les cas où cela est possible.

## **7.2 Le répertoire de conférence**

Le répertoire de conférence permet à un nœud d'une conférence de savoir quels sont les autres nœuds de la conférence et fournit des informations sur chacun de ces nœuds. Immédiatement après qu'un nœud a fait son entrée dans la conférence, soit en la créant, en entrant dans une conférence existante ou en y étant invité, son contrôleur nodal annoncera sa présence par l'émission d'une primitive de demande GCC-Conference-Announce-Presence (annonce de présence dans la conférence). Les résultats de cette demande dépendent de la catégorie nodale du nœud entrant.

Pour les nœuds conventionnels (lorsque tous les nœuds qui existaient avant l'introduction des catégories nodales sont considérés comme étant conventionnels), les fournisseurs GCC des nœuds de la conférence échangent les informations de mise à jour nécessaires pour inclure le nœud entrant dans le répertoire de conférence. Le répertoire de conférence à jour, ou une mise à jour delta du répertoire de conférence, est distribué à tous les nœuds de la conférence, au moyen d'une primitive d'indication GCC-Conference-Roster-Report (compte rendu de répertoire de conférence). Un nœud conventionnel qui entre dans une conférence n'est pas considéré comme un membre actif de la conférence tant qu'il n'a pas reçu un répertoire de conférence dans lequel il figure.

Pour les nœuds comptés, les fournisseurs GCC des nœuds affectés par la conférence échangent les informations nécessaires pour mettre à jour le répertoire de conférence afin d'y inclure le nœud nouvellement entré. Le répertoire de conférence mis à jour, ou une mise à jour delta du répertoire de

conférence, est distribué à tous les nœuds conventionnels de la conférence ainsi qu'au nœud compté qui entre dans la conférence, ce qui produit une indication GCC-Conference-Roster-Report. Un nœud compté qui entre dans une conférence n'est pas considéré comme étant un membre actif de celle-ci tant qu'il n'a pas reçu un répertoire de conférence dans lequel il est inclus.

Les nœuds anonymes n'ont pas d'incidence sur le répertoire de conférence. Néanmoins, leurs contrôleurs nodaux locaux restent appelés à annoncer leur présence au fournisseur GCC local.

L'information contenue dans le répertoire de conférence peut être modifiée à tout instant par le contrôleur nodal à tout nœud conventionnel ou compté. Cela se fait par l'émission d'une nouvelle primitive de demande GCC-Conference-Announce-Presence. Il en résulte l'envoi d'une autre primitive d'indication GCC-Conference-Roster-Report distribuant à l'ensemble des nœuds un répertoire de conférence à jour.

Quand des nœuds conventionnels ou comptés quittent une conférence pour une raison quelconque, un nouveau répertoire de conférence est distribué par le service GCC qui produit une nouvelle primitive d'indication GCC-Conference-Roster-Report.

### 7.2.1 Description des services abstraits

La liste qui suit récapitule les primitives définies dans le présent sous-paragraphe et en résume les fonctions:

- GCC-Conference-Announce-Presence (*annonce de présence dans la conférence*) – Utilisée par le contrôleur nodal pour annoncer la présence d'un nœud dans une conférence. Cette primitive doit être émise immédiatement après que le nœud est entré ou a été ajouté à une conférence.
- GCC-Conference-Roster-Report (*compte rendu de répertoire de conférence*) – Générée par le service GCC en réponse à une modification du répertoire de conférence à la suite de l'entrée ou la sortie de nœuds de la conférence.
- GCC-Conference-Roster-Inquire (*interrogation de répertoire de conférence*) – Utilisée par le contrôleur nodal ou par des entités de protocole d'application, pour interroger le répertoire d'une conférence.

#### 7.2.1.1 GCC-Conference-Announce-Presence (annonce de présence dans la conférence)

Dès qu'un nœud est entré dans une conférence, il annoncera sa présence à la conférence au moyen de la primitive de demande GCC-Conference-Announce-Presence. Le contrôleur nodal est responsable de l'émission de cette demande. La primitive peut être émise de nouveau à tout instant si l'information qu'elle contient a changé. Le Tableau 7-18 donne les paramètres et les types de primitives de ce service. La Figure 7-15 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

NOTE – Le répertoire de conférence contient la liste de tous les terminaux et de toutes les MCU faisant partie de la conférence à un moment donné. Il est probable qu'une vue de ce répertoire par un utilisateur, ainsi que les indications fournies par l'utilisateur au sujet des terminaux entrant ou sortant de la conférence ne prennent en compte que des terminaux et des terminaux multiaccès. Le paramètre "type de nœud" de cette primitive peut être employé pour distinguer ces types de nœuds et exclure des vues utilisateur les nœuds sans intérêt. En outre, le paramètre "propriété de nœud" fournit également une information supplémentaire concernant l'utilisation de l'équipement en indiquant si un nœud est un équipement de gestion et si un nœud est un périphérique subordonné à un autre nœud. Ces deux caractéristiques peuvent également être utilisées par un système afin de déterminer s'il y a lieu ou non d'inclure un nœud particulier dans sa vue des participants de la conférence. Les équipements de gestion et les périphériques ne font pas généralement partie d'une telle liste.



**Tableau 7-18 – Types de primitives et paramètres –  
GCC-Conference-Announce-Presence**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=RQ)
Type de nœud	M	
Propriétés du nœud	M	
Nom de nœud	O	
Nom(s) de participant	O	
Information de site	O	
Adresse réseau	O	
Autre nom de nœud	O	
Données utilisateur	O	
Résultat		M

*Identificateur de conférence:* identificateur de la conférence à laquelle se réfère la primitive.

*Type de nœud:* le type de nœud est soit un terminal, une MCU ou un terminal multiaccès.

*Propriétés du nœud:* constitué de deux indicateurs indépendants. L'un d'eux indique si un nœud est ou non un équipement de gestion – par exemple un système de réservation, l'autre indique si le nœud est ou non un périphérique subordonné à un autre nœud.

*Nom de nœud:* chaîne de texte unicode contenant le nom du nœud, par exemple "London", de longueur maximale 255 caractères.

*Nom(s) de participant:* liste de chaînes de texte unicode contenant chacune le nom d'un participant à une réunion. La longueur maximale de chaque chaîne est de 255 caractères.

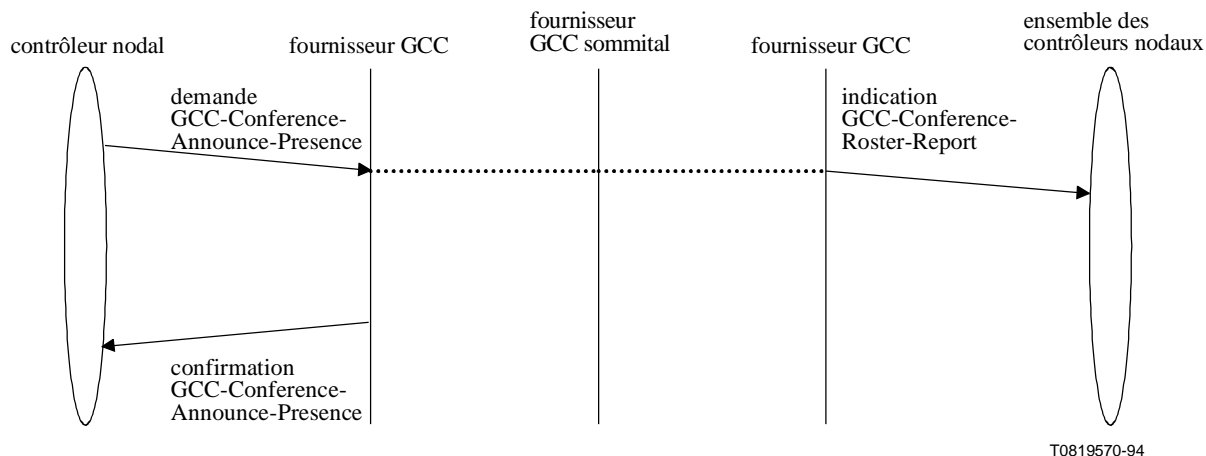
*Information de site:* chaîne de texte unicode contenant d'autres informations sur le nœud. Elle peut, par exemple, être employée pour indiquer des numéros de téléphone vocal ou de télécopie sur le site.

*Adresse réseau:* paramètre optionnel indiquant l'adresse réseau du nœud. Ce paramètre contient des champs indiquant une information de type de réseau suivie d'une ou de plusieurs adresses réseau effectives. Il sera utilisé s'il est possible que le nœud qui annonce sa présence peut faire ultérieurement l'objet d'une tentative d'addition dans une autre conférence au moyen de la primitive GCC-Conference-Add (addition à une conférence). Ceci permet à une MCU de connaître l'adresse réseau et de déterminer, lors de l'addition d'un nœud, s'il existe déjà une connexion physique pour ce nœud. L'omission de ce paramètre peut faire obstacle à la réutilisation ultérieure d'une connexion physique existante pour ajouter le nœud à une autre conférence. Ce paramètre devrait également être utilisé s'il est possible qu'une conférence établie initialement en mode point à point peut être routée de nouveau d'une manière automatique sur une MCU afin de permettre l'addition de nouveaux nœuds. Ce paramètre permet dans ce cas à l'un des nœuds initiaux de se connecter de nouveau à l'autre nœud en passant par la MCU au moyen d'une primitive GCC-Conference-Add.

*Autre identificateur de nœud:* champ pouvant être utilisé pour associer au nœud annonceur (et à son identificateur de nœud) un autre identificateur de nœud ayant été défini dans un but différent. Cet autre identificateur n'est pas prévu pour représenter des identités dans le même espace de dénombrement que les identificateurs de nœud, mais plutôt selon un plan de dénombrement non spécifié par la présente Recommandation. Par exemple, dans le cas de nœuds RNIS qui prennent en charge la Rec. UIT-T H.243, l'autre identificateur de nœud peut être l'identificateur de site H.243 d'une longueur de deux octets.

*Données utilisateur:* données utilisateur optionnelles pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation. Il est à noter que ces données sont stockées dans le fournisseur GCC sommital comme partie du répertoire de conférence. Il s'ensuit que ce paramètre n'est pas prévu pour stocker des volumes importants de données. Un utilisateur qui ne se conforme pas à cette indication pourrait risquer d'être éjecté involontairement.

*Résultat:* indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



**Figure 7-15 – Séquence de primitives – GCC-Conference-Announce-Presence**

### 7.2.1.2 GCC-Conference-Roster-Report (compte rendu de répertoire de conférence)

Le répertoire de conférence est distribué par le service GCC à tous les nœuds de la conférence par l'envoi d'une primitive GCC-Conference-Roster-Report à l'ensemble des contrôleurs nodaux chaque fois que le répertoire de conférence est modifié pour une cause quelconque: nouveau nœud entrant dans la conférence, nœud quittant la conférence ou mise à jour d'une entrée de la liste. Le Tableau 7-19 donne les paramètres et les types de primitives de ce service. La Figure 7-16 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-19 – Types de primitives et paramètres – GCC-Conference-Roster-Report**

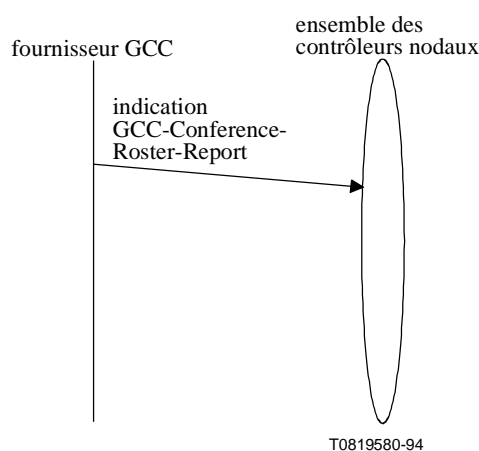
Paramètre	Indication
Identificateur de conférence	M
Répertoire de conférence	M

*Identificateur de conférence:* l'identificateur de la conférence à laquelle se réfère la primitive.

*Répertoire de conférence:* une liste de tous les nœuds faisant partie de la conférence en question. Le contenu du répertoire de conférence est donné par le Tableau 7-20.

**Tableau 7-20 – Contenu du paramètre répertoire de conférence**

Paramètre	Description
Liste de nœuds de conférence	Liste des nœuds faisant partie de la conférence ainsi qu'une information concernant chaque nœud. Le contenu d'une entrée de la liste est donné par le Tableau 7-22.
Numéro d'instance	Numéro d'instance du répertoire de conférence. Ce nombre de 16 bits est incrémenté modulo $2^{16}$ chaque fois que le contenu du répertoire de conférence est modifié.
Indicateur d'ajout de nœud	Indicateur signalant qu'un ou plusieurs nœuds ont été ajoutés à la conférence depuis la dernière instance. Cet indicateur n'est pas mutuellement exclusif avec l'indicateur de suppression de nœud.
Indicateur de suppression de nœud	Indicateur signalant qu'un ou plusieurs nœuds ont été supprimés de la conférence depuis la dernière instance. Cet indicateur n'est pas mutuellement exclusif avec l'indicateur d'ajout de nœud.



**Figure 7-16 – Séquence de primitives – GCC-Conference-Roster-Report**

### 7.2.1.3 GCC-Conference-Roster-Inquire (interrogation de répertoire de conférence)

La primitive de demande GCC-Conference-Roster-Inquire renvoie le répertoire de conférence complet de la conférence indiquée. La primitive est utilisable par des entités de protocole d'application et par le contrôleur nodal. Elle leur permet d'obtenir d'une manière indépendante un répertoire de conférence de la part de leur fournisseur GCC local. Le Tableau 7-21 donne les paramètres et les types de primitives de ce service. La Figure 7-17 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-21 – Types de primitives et paramètres – GCC-Conference-Roster-Inquire**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Nom de conférence		M
Description de conférence		C
Répertoire de conférence		M
Résultat		M

*Nom de conférence*: nom de conférence tel qu'il est contenu dans le profil de conférence.

*Description de conférence*: description de conférence telle qu'elle est contenue dans le profil de conférence. Ce paramètre n'est présent que si une description de conférence a été définie lors de la création de la conférence.

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Répertoire de conférence*: liste des nœuds faisant partie de la conférence en question. Le format du répertoire de conférence est donné par le Tableau 7-23.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".

**Tableau 7-22 – Contenu d'une entrée de la liste de nœuds de conférence**

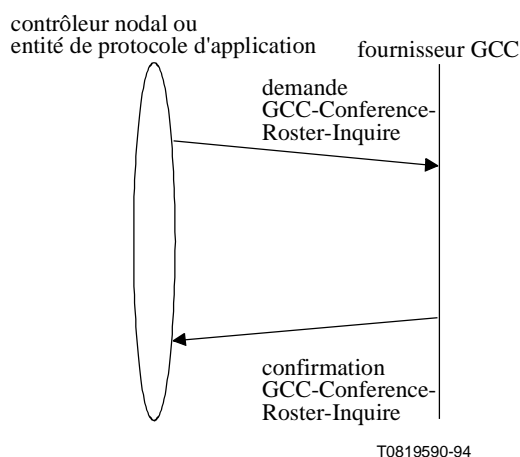
Paramètre	Description
Identificateur de nœud	Identificateur d'utilisateur du service MCS du fournisseur GCC de ce nœud.
Identificateur du nœud supérieur (conditionnel)	Identificateur d'utilisateur du service MCS du fournisseur GCC du nœud, s'il existe, qui est le supérieur immédiat de ce nœud dans la hiérarchie de connexion. Ce paramètre n'existe pas pour le fournisseur GCC sommital.
Type de nœud	Le type de nœud est soit un terminal, une unité MCU ou un terminal multiaccès.
Propriétés du nœud	Constitué de deux indicateurs indépendants. Le premier indique si le nœud est ou non un équipement de gestion, par exemple un système de réservation. L'autre indique si le nœud est ou non un périphérique subordonné à un autre nœud.
Nom de nœud (conditionnel)	Chaîne de texte unicode de longueur maximale 255 caractères contenant le nom du nœud, par exemple "London".
Noms des participants (conditionnel)	Liste de chaînes de texte unicode contenant chacune le nom d'un participant à une réunion. La longueur maximale de chaque chaîne est de 255 caractères.
Information de site (conditionnel)	Chaîne de texte unicode contenant d'autres informations sur le nœud. Elle peut être employée pour indiquer par exemple des numéros de téléphone vocal ou de télécopie sur le site.
Adresse réseau (conditionnel)	Ce paramètre décrit la connexion logique utilisée par le nœud pour entrer dans la conférence. Chaque segment de la connexion logique est décrit en termes de type de réseau, d'adresse réseau, de débit utile le cas échéant et, facultativement, de profil multimédia devant être exploité, d'algorithme d'agrégation de circuits numériques si applicable, et de liste de médias (par exemple audio, vidéo, données) visés par ce segment. Dans le protocole de commande GCC, ce paramètre est représenté par les structures ASN.1 "NetworkAddress" et "NetworkAddressV2", dont la description et l'utilisation se trouvent dans l'Annexe B.
Autre nom de nœud (conditionnel)	Champ utilisé pour associer au nœud annonceur, et à son identificateur de nœud, un autre identificateur de nœud ayant été défini dans un but différent. Cet autre identificateur n'est pas prévu pour représenter des identités dans le même espace de dénombrement que les identificateurs de nœud, mais plutôt selon un plan de dénombrement non spécifié par la présente Recommandation. Par exemple, en cas de nœuds RNIS qui prennent en charge la Rec. UIT-T H.243, l'autre identificateur de nœud peut être l'identificateur de site H.243 d'une longueur de deux octets.

**Tableau 7-22 – Contenu d'une entrée de la liste de nœuds de conférence**

Paramètre	Description
Catégorie nodale (conditionnel)	Ce champ indique la catégorie nodale du nœud. Les nœuds conventionnels et comptés sont les seules catégories valides pour ce champ car les nœuds anonymes n'apparaissent pas dans un répertoire de conférence.
Données utilisateur (conditionnel)	Données utilisateur facultatives pouvant être employées pour réaliser des fonctions qui ne font pas partie du domaine d'application de la présente Recommandation.

**Tableau 7-23 – Contenu du paramètre répertoire de conférence**

Paramètre	Description
Liste de nœuds de conférence	Liste des nœuds faisant partie de la conférence ainsi qu'une information sur chaque nœud. Le contenu d'une entrée de la liste est donné par le Tableau 7-22.
Numéro d'instance	Numéro d'instance du répertoire de conférence. Ce nombre de 16 bits est incrémenté modulo $2^{16}$ chaque fois que le contenu du répertoire de conférence est modifié.



**Figure 7-17 – Séquence de primitives – GCC-Conference-Roster-Inquire**

### 7.3 Le répertoire d'application

Le répertoire d'application permet à un nœud faisant partie d'une conférence de savoir quelles sont les entités de protocole d'application disponibles sur les nœuds de la conférence et fournit sur ces entités une information qui suffit pour initialiser une communication directe entre entités de protocole d'application homologues. Le répertoire peut contenir des entités de protocole d'application basées sur des protocoles d'application normalisés et des entités basées sur des protocoles d'application non normalisés.

#### 7.3.1 Contenu du répertoire d'application

Le fournisseur GCC établit pour chaque conférence une liste d'informations associée à chaque entité de protocole d'application. Ces listes constituent le répertoire d'application local qui pourra être échangé avec les autres nœuds dans le cadre de la procédure d'échange de répertoire d'application. Le résultat de cet échange sera une liste d'informations associée à toutes les entités de protocole

d'application de la totalité de la conférence, définie d'une manière collective comme répertoire d'application de conférence.

Le répertoire d'application local est tenu par chaque fournisseur GCC et mémorisé dans sa base de données locale. A un moment pertinent de l'établissement de la conférence, un nœud peut échanger son répertoire d'application local avec les autres nœuds et le fournisseur GCC recevra le répertoire d'application de conférence, dont il fournira le compte rendu à toutes les entités de protocole d'application et au contrôleur nodal. Cet échange se passe au sein du service GCC et n'est visible pour les entités de protocole d'application que par les primitives qui peuvent être échangées. La question de savoir si un fournisseur GCC échange ou non son répertoire d'application local dépend de la catégorie nodale dans laquelle le nœud s'inscrit. Pour les nœuds conventionnels et comptés, le répertoire d'application local est échangé, alors qu'il ne l'est pas pour les nœuds anonymes.

Les contenus des répertoires d'application locaux et du répertoire d'application de conférence sont les suivants:

- *répertoire d'application local* – Le répertoire d'application local contient une clé de session, un enregistrement d'application et une liste de capacités d'application pour toute entité de protocole d'application qui s'est inscrite auprès du fournisseur GCC local;
- *répertoire d'application de conférence* – Le répertoire d'application de conférence contient, pour toute session de protocole d'application, la clé de session de cette session, une liste de capacités d'application comportant l'ensemble des capacités négociées pour la session et une liste des entités de protocole d'application qui se sont inscrites dans la session. La liste des entités de protocole d'application inscrites contient l'identificateur du nœud sur lequel les entités de protocole d'application se sont inscrites, un identificateur de l'entité de protocole d'application au sein de son nœud individuel et un enregistrement d'application.

Chaque ensemble d'entités de protocole d'application homologues appartenant à une session de protocole d'application donnée est caractérisé par le fait que les entités se sont inscrites en utilisant la même valeur de la clé de session. Une clé de session est définie comme suit:

- *clé de session* – Identificateur utilisé pour identifier sans ambiguïté une session de protocole d'application. Une clé de session contient deux composants. Le premier identifie le protocole d'application. Le second composant optionnel identifie une session particulière de ce protocole d'application.

Le premier composant de la clé de session, appelé plus simplement la clé de protocole d'application, identifie une spécification de protocole d'application normalisée ou non. Les clés sont structurées de telle manière qu'un protocole d'application – normalisé ou non – peut être défini comme possédant une clé non ambiguë. Une clé peut être spécifiée comme un identificateur d'objet de type ASN.1. Cette forme de clé peut être utilisée pour spécifier des protocoles d'application normalisés. L'attribution de composants d'identificateur d'objet pour les Normes et Recommandations est décrite dans l'Annexe B de [UIT-T X.680] et l'Annexe C de [UIT-T X.680]. Des identificateurs d'objet peuvent également être utilisés pour spécifier des protocoles d'application non normalisés dans le cas où des autorités administratives nationales ou privées ont reçu une autorisation directe ou indirecte de l'ISO ou de l'UIT.

Une clé peut également être spécifiée comme une chaîne d'octets OCTET STRING type ASN.1 interprétée d'une manière comparable à la structure adoptée dans [UIT-T H.221] pour désigner des commandes et des capacités non normalisées. Dans ce cas, les deux premiers octets du type chaîne d'octets définissent un code de pays et les deux octets suivants définissent un code de constructeur. Le premier octet du code de pays est attribué conformément à [UIT-T T.35], le deuxième octet et le code constructeur sont attribués sur une base nationale. Les octets supplémentaires seront choisis librement par le constructeur ou l'organe national responsable.

Le deuxième composant d'une clé de session est un identificateur de session optionnel. Il s'agit d'un identificateur de canal MCS utilisé comme identificateur non ambigu d'une session de protocole d'application et pouvant être employé comme identificateur de canal de communication par les entités de protocole d'application prenant part à cette session. Cet identificateur de canal restera attribué pendant toute la durée de la session pour en garantir l'unicité. L'absence d'identificateur de session dans une clé de session d'une entité de protocole d'application active identifie une session particulière qui est la session par défaut.

L'identificateur de session n'est pas exigé pour une session de protocole d'application inactive. Son absence peut être interprétée dans ce cas comme l'indication de la prise en charge du protocole d'application mentionné et comme la capacité d'invoquer une ou plusieurs entités de protocole d'application faisant partie de toute session identifiée ou par défaut.

Un enregistrement d'application contient les paramètres suivants, dont certains sont optionnels:

- *indicateur d'activité/inactivité* – Une entité de protocole d'application peut s'inscrire sans être immédiatement prête à recevoir des données, par exemple si elle n'a pas encore effectué les branchements sur les canaux appropriés. Une entité de protocole d'application peut s'inscrire avec l'indicateur positionné sur inactif pour signaler ce fait aux autres nœuds. Ceci peut être fait pour permettre à l'entité de protocole d'application d'utiliser les services GCC (tels que l'enregistrement) en vue de préparer son activité ou pour permettre à d'autres nœuds de prendre connaissance de la présence de cette entité de protocole d'application qui restera inactive tant qu'il n'est pas établi qu'il existe des entités similaires sur d'autres nœuds. Lorsqu'une entité de protocole d'application inactive devient active, elle peut s'inscrire de nouveau en positionnant l'indicateur sur actif. Cet indicateur peut être modifié à tout instant par l'entité de protocole d'application en réitérant l'inscription;
- *identificateur d'utilisateur d'application* – Identificateur d'utilisateur du service MCS associé à la clé de session. L'identificateur d'utilisateur d'application est pour un nœud le seul moyen de désigner un autre nœud comme participant lorsqu'il établit la convention d'utiliser un canal privé pour cette session de protocole d'application. Ce paramètre est optionnel pour des entités de protocole d'application inscrites comme inactives mais obligatoire pour des entités de protocole d'application actives. S'il n'existe pas parce que l'entité de protocole d'application ne s'est pas encore attachée au domaine MCS, il peut être fourni par la suite au moyen d'une nouvelle inscription lorsque l'entité de protocole d'application se sera attachée;
- *capacité de présidence d'opération* – Indicateur signalant si l'entité de protocole d'application possède la capacité d'agir comme entité de protocole d'application présidente dans le cas où la spécification du protocole d'application associée définit des procédures exigeant une telle entité. Le fournisseur GCC de chaque nœud choisit, pour chaque session de protocole d'application, au plus une entité de protocole d'application homologue possédant cet indicateur actif lorsqu'il génère le répertoire d'application échangé. Si un nœud devient le président de conférence, l'entité de protocole d'application désignée de ce nœud devient, si elle existe, l'entité de protocole d'application désignée pour la présidence. Cet indicateur ne s'applique qu'aux entités de protocole d'application actives. Si l'indicateur d'activité est positionné sur inactif lors de l'inscription, l'indicateur de capacité de présidence est ignoré et le fournisseur GCC fait l'hypothèse que sa valeur est FALSE (faux);
- *liste de capacités non condensée* – Paramètre optionnel permettant à une entité de protocole d'application de donner la liste de capacités, normalisées ou non, devant être gérées dans le répertoire d'application comme faisant partie de l'enregistrement d'application de chaque entité de protocole d'application plutôt que d'être condensée par l'application d'un ensemble

de règles comme c'est le cas pour les applications listées dans la liste de capacités d'application.

La liste de capacités d'application est définie comme suit:

- *liste de capacités d'application*: paramètre facultatif pouvant être utilisé pour fournir une liste spécifique des capacités de l'entité de protocole d'application. Alors que les capacités en elles-mêmes sont spécifiques du protocole d'application, elles figurent dans la liste avec une information complémentaire qui permet au service GCC de déterminer l'ensemble commun de capacités pour la session de protocole d'application. Cette information est fournie aux entités de protocole d'application homologues et évite l'échange complet de la liste de toutes les capacités entre tous les nœuds. Chaque capacité de la liste des capacités est marquée au moyen d'un indicateur de classe. Cet indicateur de classe indique la règle à appliquer pour déterminer l'ensemble commun de capacités. Le Tableau 7-24 donne la liste des classes de capacité. On notera que ni les nœuds comptés ni les nœuds anonymes ne peuvent avoir d'incidence sur les capacités de répertoire d'application de conférence.

**Tableau 7-24 – Règles de combinaison des capacités**

Classe	Description
Logique	Si l'une quelconque des entités de protocole d'application homologues d'une conférence indique qu'elle utilise cette capacité, la liste finale des capacités indique le nombre d'entités de protocole d'application homologues qui ont annoncé cette capacité.
Minimum sans signe	Les octets du paramètre sont traités en bloc comme un entier sans signe. La liste finale des capacités indique la valeur minimale de la capacité pour l'ensemble des entités de protocole d'application homologues qui l'ont annoncée ainsi que le nombre d'entités de protocole d'application homologues qui ont annoncé cette capacité.
Maximum sans signe	Les octets du paramètre sont traités en bloc comme un entier sans signe. La liste finale des capacités indique la valeur maximale de la capacité pour toutes les entités de protocole d'application homologues qui l'ont annoncée ainsi que le nombre d'entités de protocole d'application homologues qui ont annoncé cette capacité.

En plus des classifications de base, il est également possible d'imbriquer des capacités. L'imbrication n'est pas faite directement mais plutôt par une interprétation pertinente des trois classes de capacité. Par exemple un protocole d'application peut définir une capacité Y donnée comme étant conditionnée par une capacité logique X. Si la règle est appliquée et que le protocole d'application stipule que la capacité Y est conditionnée par la capacité X, alors la valeur du paramètre Y figurant dans l'ensemble de capacités final est obtenue de la manière suivante. Si Y est une capacité de la classe logique, le paramètre Y contient le nombre d'entités de protocole d'application homologues qui, disposant de la capacité X, disposent également de la capacité Y. Si Y est une capacité de la classe numérique, le paramètre Y contient la valeur, minimale ou maximale selon le cas, des paramètres Y, l'évaluation étant effectuée sur l'ensemble des entités de protocole d'application homologues disposant de la capacité X. A noter dans ce cas que si le nombre d'entités homologues disposant de la capacité Y est inférieur à celui des entités disposant de la capacité X – ce qui signifie qu'une des entités ne prend pas en charge la capacité Y au-delà de la valeur par défaut du paramètre correspondant – alors Y ne sera pas négociée. La procédure de négociation d'un paramètre Y numérique telle qu'elle est donnée ci-dessus ne conduit à un résultat valide que si le nombre d'entités homologues disposant de la capacité X est égal à celui des entités disposant de la capacité Y.



### 7.3.2 Description du processus d'échange de répertoire d'application

Toute entité de protocole d'application signale sa présence à son fournisseur GCC local à travers un point d'accès GCCSAP qui lui permet de communiquer avec ce fournisseur. La création et la gestion du point d'accès GCCSAP constituent un problème local non traité par la présente Recommandation.

Les entités de protocole d'application sont informées de l'existence d'une conférence à laquelle appartient le nœud par la primitive d'indication GCC-Application-Permission-To-Enroll (autorisation d'inscription d'application). Une fois qu'une entité de protocole d'application a été informée de l'existence d'une conférence, elle enverra une primitive de demande GCC-Application-Enroll (inscription d'application) au fournisseur GCC. Cette demande d'inscription peut signaler que l'entité de protocole d'application désire s'inscrire ou non dans la conférence selon que l'indicateur d'inscription est positionné sur "inscription" ou "résiliation". Dans le dernier cas, les paramètres de l'enregistrement d'application n'ont pas besoin de figurer dans la primitive. Une entité de protocole d'application qui n'a pas choisi de s'inscrire initialement dans la conférence peut s'inscrire à tout moment par l'émission d'une primitive de demande GCC-Application-Enroll tant que l'autorisation d'inscription n'aura pas été révoquée par la réception d'une primitive GCC-Application-Permission-To-Enroll ayant l'indicateur d'autorisation positionné sur "résiliation".

Il est nécessaire de prendre un certain nombre d'actions préliminaires concernant des paramètres utilisés dans une primitive de demande GCC-Application-Enroll. Ceci concerne en premier lieu l'identificateur d'utilisateur d'application dont la présence est obligatoire dans la primitive de demande si l'indicateur d'activité/inactivité est positionné. L'entité de protocole d'application doit tout d'abord s'attacher au domaine MCS donné dans la primitive d'indication GCC-Application-Permission-To-Enroll afin d'obtenir un identificateur d'utilisateur d'application. Ceci se fait par l'émission de la primitive de demande MCS-Attach-User (rattachement d'utilisateur). La réception de la confirmation MCS-Attach-User signale qu'un identificateur d'utilisateur d'application a été attribué à l'entité de protocole d'application. Cet identificateur doit être fourni dans la primitive de demande GCC-Application-Enroll. Si la tentative de s'attacher au domaine échoue à cause d'une perte de connexion survenue entre-temps, l'entité de protocole d'application supposera que le nœud ne fait plus partie de la conférence et attendra la réception d'une nouvelle primitive d'indication GCC-Application-Permission-To-Enroll avant de continuer. Quand le nœud se déconnecte de la conférence, l'entité de protocole d'application reçoit directement une notification du service MCS indiquant que l'attachement n'est plus valide. L'entité de protocole d'application supposera que le nœud ne fait plus partie de la conférence et attendra la réception d'une nouvelle primitive d'indication GCC-Application-Permission-To-Enroll avant de continuer. Si l'autorisation d'inscription est révoquée par une primitive GCC-Application-Permission-To-Enroll, l'entité de protocole d'application ne tentera de s'attacher au domaine correspondant.

Le second paramètre de ce type est l'identificateur de session optionnel faisant partie de la clé de session. L'identificateur de session sera déterminé à partir d'un identificateur de canal MCS (et non à partir d'un identificateur d'utilisateur). L'identificateur de session est fourni dans la primitive d'indication GCC-Application-Invoke (invocation d'application) lors de l'inscription dans une session. Si l'entrée se fait dans une session en cours, l'identificateur de session peut être obtenu du dernier répertoire d'application reçu. Si une nouvelle session est créée, un nouvel identificateur de canal peut être attribué au moyen d'une primitive MCS-Channel-Join (branchement sur canal) ou d'une primitive MCS-Channel-Convenc (instigation de canal).

Le fournisseur GCC crée une nouvelle entrée dans son répertoire d'application local comme conséquence de la primitive de demande GCC-Application-Enroll. Si une entrée existe déjà pour une entité de protocole d'application, le contenu du répertoire d'application local est modifié pour tenir compte de la nouvelle information. Si l'indicateur d'inscription est positionné sur "résiliation", l'entrée est retirée du répertoire.

Lors du démarrage d'une conférence, le fournisseur GCC attend, après avoir envoyé la primitive GCC-Application-Permission-To-Enroll, que toutes les entités de protocole d'application connectées au fournisseur GCC à travers un point d'accès GCCSAP envoient une primitive GCC-Application-Enroll afin de s'inscrire ou d'indiquer explicitement leur intention de ne pas s'inscrire. Pour les nœuds conventionnels et comptés, une fois que toutes les entités de protocole d'application ont répondu, le fournisseur GCC échange le répertoire d'application local avec tous les autres nœuds de la conférence. Toutes les modifications du répertoire d'application local sont immédiatement échangées avec les autres nœuds pendant toute la durée d'une conférence. On notera que seuls les nœuds conventionnels peuvent créer de nouvelles sessions de protocole d'application (car ce sont les premiers nœuds à échanger un répertoire d'application local associé à une session inexistante). Les fournisseurs GCC des nœuds comptés doivent donc attendre l'ouverture d'une session associée à une entité du protocole d'application (APE, *application protocol entity*) particulière avant de pouvoir échanger ses informations associées de répertoire d'application local.

Pour les nœuds anonymes, le fournisseur GCC local n'échangera jamais son répertoire d'application local. Par contre, il utilisera les informations d'inscription pour déterminer à quelles sessions de protocole d'application les entités locales du protocole d'application s'intéressent. Lorsqu'un répertoire d'application de conférence associé à l'une de ces entités APE est reçu par le fournisseur GCC local, ce répertoire est immédiatement acheminé à l'entité APE intéressée.

Dans tous les cas ci-dessus, lorsqu'un nœud quelconque a procédé à un nouvel échange, un nouveau répertoire d'application de conférence est produit et distribué à tous les nœuds intéressés. Le fournisseur GCC envoie la totalité du répertoire au contrôleur nodal et envoie des parties de ce répertoire aux entités de protocole d'application inscrites au moyen de la primitive d'indication GCC-Application-Roster-Report (compte rendu de répertoire d'application). La diffusion du répertoire à une entité de protocole d'application concerne la partie du protocole d'application de conférence qui s'applique d'une manière spécifique à cette session de protocole d'application, mais peut également comprendre d'autres parties. Cela se fait sous la forme d'une liste pour chaque session. Chaque entrée de la liste contient l'identificateur de nœud et l'identificateur de protocole d'application qui, associés, permettent d'identifier l'entité de protocole d'application. Elle peut également contenir la liste des capacités d'application pour la session de protocole d'application. Il s'agit d'une liste des capacités communes pour cette session de protocole d'application, fondée sur l'application des règles de classe de capacité aux capacités déclarées par toutes les entités de protocole d'application de nœud conventionnel homologues. Dans le cas d'un contrôleur nodal, la primitive d'indication GCC-Application-Roster-Report contient la totalité du répertoire d'application de conférence, c'est-à-dire l'identificateur de nœud et les enregistrements d'application de tous les protocoles d'application qui se sont inscrits sur ce nœud. Elle peut également contenir une liste de capacités d'application pour chaque protocole d'application.

Une entité de protocole d'application n'est pas considérée comme faisant partie d'une conférence tant qu'elle n'a pas reçu une primitive d'indication GCC-Application-Roster-Report contenant un répertoire d'application dans lequel elle figure. Si une entité de protocole d'application précédemment inscrite reçoit à un instant quelconque une primitive d'indication GCC-Application-Roster-Report dans laquelle elle ne figure plus, cette entité de protocole d'application ne sera plus considérée comme inscrite. L'entité peut tenter de s'inscrire de nouveau en envoyant une primitive de demande GCC-Application-Enroll. Dans les nœuds comptés et anonymes, il appartient au fournisseur GCC local d'insérer, dans le répertoire, un enregistrement (associé à une entité APE locale qui essayait de s'inscrire à une session) avant d'acheminer l'indication associée GCC-Application-Roster-Report.

Une entité de protocole d'application peut se retirer à tout moment du répertoire d'application par l'émission d'une primitive de demande GCC-Application-Enroll avec un indicateur d'inscription positionné sur "résiliation". Ceci supprime l'enregistrement d'application de cette entité de protocole d'application dans le répertoire d'application local ainsi que dans le répertoire d'application de conférence de tous les nœuds de la conférence.

Une entité de protocole d'application ou le contrôleur nodal peut interroger à tout moment une partie du répertoire d'application de conférence. Tous les enregistrements d'application qui correspondent à la clé de session de la primitive de demande GCC-Application-Roster-Inquire sont renvoyés pour tous les nœuds de la conférence. Une clé nulle fournit comme résultat la totalité du répertoire d'application de la conférence.

Une entité de protocole d'application ou un contrôleur nodal peut tenter d'invoquer à distance une entité de protocole d'application d'un autre nœud. Ceci se fait par l'émission d'une primitive de demande GCC-Application-Invoke. La demande peut, d'une manière optionnelle, contenir la spécification d'une liste de nœuds destinataires. La primitive d'indication GCC-Application-Invoke correspondante est reçue par le contrôleur nodal de la destination indiquée ou par l'ensemble des nœuds si aucune destination n'est indiquée. Le contrôleur nodal peut, d'une manière optionnelle, répondre à cette indication en invoquant une entité de protocole d'application homologue qui peut à son tour s'inscrire dans la conférence.

### **7.3.3 Description des services abstraits**

La liste qui suit récapitule les primitives définies dans le présent sous-paragraphe et en résume les fonctions:

- GCC-Application-Permission-To-Enroll (*autorisation d'inscription d'application*) – Générée par le service GCC et envoyée à toutes les entités de protocole d'application qui ont signalé leur présence au fournisseur GCC chaque fois que le nœud local est entré dans une conférence. Cette primitive signale que l'entité de protocole d'application peut s'inscrire. Lorsque l'indicateur d'inscription est positionné sur "résiliation", elle révoque la possibilité d'inscription une fois que le nœud ne fait plus partie de la conférence.
- GCC-Application-Enroll (*inscription d'application*) – Utilisée par des entités de protocole d'application pour établir ou terminer les communications avec d'autres entités de protocole d'application dans une conférence. L'utilisation de cette primitive génère, modifie, ou supprime une entrée du répertoire d'application échangé avec les autres nœuds.
- GCC-Application-Roster-Report (*compte rendu de répertoire d'application*) – Générée par le service GCC en réponse à toute modification du répertoire d'application induite par l'inscription ou le retrait d'entités de protocole d'application ou encore par l'entrée ou la sortie des nœuds de la conférence.
- GCC-Application-Roster-Inquire (*demande de répertoire d'application*) – Utilisée par le contrôleur nodal ou des entités de protocole d'application pour demander une partie ou la totalité du répertoire d'application en cours à tout instant de la conférence.
- GCC-Application-Invoke (*invocation d'application*) – Utilisée pour signaler à un ensemble d'autres nœuds d'une conférence l'ordre d'invoquer une entité de protocole d'application pour une session particulière d'un protocole d'application.

#### **7.3.3.1 GCC-Application-Permission-To-Enroll (autorisation d'inscription d'application)**

La primitive d'indication GCC-Application-Permission-To-Enroll indique à une entité de protocole d'application locale que le nœud local est entré dans la conférence mentionnée et que l'entité de protocole d'application peut s'inscrire dans cette conférence. Cette primitive est également utilisée pour résilier une telle autorisation si le nœud local ne fait plus partie de la conférence. Alors qu'une entité de protocole d'application attachée au service MCS reçoit de ce service une notification lorsque la conférence se termine pour ce nœud, une entité de protocole d'application qui a choisi de ne pas se connecter ou de ne pas s'inscrire ne reçoit pas nécessairement une telle notification. Cette primitive indique à une telle entité qu'elle ne peut plus s'inscrire dans la conférence ou s'attacher au domaine MCS correspondant. La primitive d'indication GCC-Application-Permission-To-Enroll est envoyée par un fournisseur GCC à toutes les entités de protocole d'application qui ont informé le fournisseur GCC de leur présence après que le nœud est entré dans la conférence. Le Tableau 7-25

donne les paramètres et les types de primitives de ce service. La Figure 7-18 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

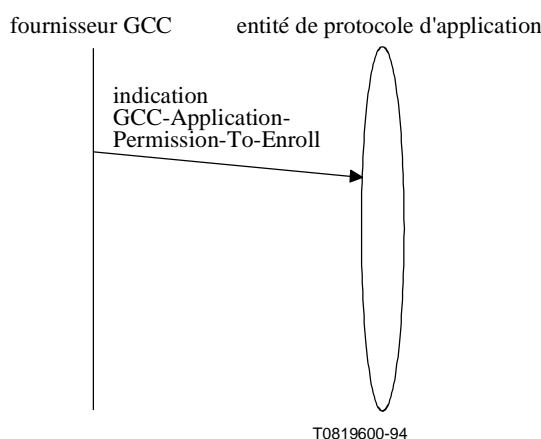
**Tableau 7-25 – Types de primitives et paramètres – GCC-Application-Permission-To-Enroll**

Paramètre	Indication
Identificateur de conférence	M
Indicateur d'autorisation	M
Catégorie nodale	C

*Identificateur de conférence*: identificateur de la conférence dans laquelle l'entité de protocole d'application peut s'inscrire. Ce paramètre est égal à l'identificateur du domaine MCS auquel l'entité de protocole d'application peut s'attacher.

*Indicateur d'autorisation*: indicateur signalant si l'entité de protocole d'application reçoit l'autorisation de s'inscrire ou si elle reçoit une résiliation.

*Catégorie nodale*: champ qui informe l'entité APE de la catégorie établie pour un nœud dans cette conférence, lors de l'entrée de ce nœud.



**Figure 7-18 – Séquence de primitives – GCC-Application-Permission-To-Enroll**

### 7.3.3.2 GCC-Application-Enroll (inscription d'application)

La primitive de demande GCC-Application-Enroll est émise par une entité de protocole d'application dans le but de prendre part à la conférence indiquée. Cette primitive sera émise en réponse à une primitive d'indication GCC-Permission-To-Enroll. L'entité de protocole d'application peut toutefois positionner l'indicateur d'inscription sur "résiliation" pour signaler qu'elle ne souhaite pas s'inscrire pour le moment. Une entité de protocole d'application peut à tout moment s'inscrire, s'inscrire de nouveau (pour modifier son entrée dans le répertoire d'application) ou annuler l'inscription. Si une entité de protocole d'application qui a déjà retiré son inscription se retire de nouveau, la demande est acceptée avec un résultat positif, mais le statut de l'entité de protocole d'application n'est pas modifié. Lorsqu'une entité de protocole d'application est inscrite, les paramètres correspondants constituent un enregistrement d'application qui est diffusé à la conférence au cours de l'échange de répertoire d'application. Des entités de protocole d'application non inscrites ne sont disponibles pour aucun participant local ou distant et ne reçoivent pas d'indication GCC-Application-Roster-Report. Le Tableau 7-26 donne les paramètres et les types de primitives de ce service. La Figure 7-19 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-26 – Types de primitives et paramètres – GCC-Application-Enroll**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de session	M	M(=)
Indicateur d'activité ou d'inactivité	M	
Identificateur d'utilisateur d'application	C	
Capacité de présidence d'opération	O	
Canal de démarrage	O	
Liste de capacités d'application non condensée	O	
Liste de capacités d'application	O	
Indicateur d'inscription ou de résiliation	M	
Identificateur d'entité de protocole d'application		C
Identificateur de nœud		C
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Clé de session*: identificateur non ambigu d'une session de protocole d'application particulière. Cette information correspond à l'identificateur du protocole d'application concaténé avec un identificateur de session optionnel.

*Indicateur d'activité ou d'inactivité*: indicateur utilisé pour signaler si l'entité de protocole d'application qui s'inscrit est pleinement active, indiquant qu'elle peut exécuter les opérations requises par la spécification du protocole d'application correspondant, ou si elle est inactive. Une entité de protocole d'application peut indiquer qu'elle est inactive si elle ne s'est pas encore branchée à des canaux dynamiques mais doit d'abord consulter le répertoire pour déterminer les canaux à utiliser. De même, elle peut s'inscrire comme inactive pour indiquer la prise en charge du protocole d'application correspondant sans devenir active avant de savoir qu'il existe des entités de protocole d'application similaires sur d'autres nœuds. La signification spécifique de cet indicateur sera définie par la spécification de chaque protocole d'application.

*Identificateur d'utilisateur d'application*: identificateur d'utilisateur MCS attribué à cette entité de protocole d'application. Ce paramètre est optionnel dans le cas d'entités de protocole d'application inactives mais obligatoire pour celles qui sont actives. Une entité de protocole d'application inscrite avec un identificateur d'utilisateur MCS fait une erreur si elle s'inscrit à nouveau avec un identificateur d'utilisateur MCS différent. Ce paramètre n'est pas nécessaire lors d'une résiliation.

*Capacité de présidence d'opération*: indicateur signalant si l'entité de protocole d'application possède la capacité d'agir comme entité de protocole d'application présidente dans le cas où la spécification du protocole d'application associée définit des procédures exigeant une entité de protocole d'application présidente. Le fournisseur GCC de chaque nœud choisit, pour chaque session de protocole d'application, au plus une entité de protocole d'application homologue possédant cet indicateur actif lorsqu'il génère le répertoire d'application échangé. Si un nœud devient le président de conférence, l'entité de protocole d'application désignée de ce nœud, si elle existe, devient l'entité de protocole d'application désignée pour la présidence. L'application qui s'inscrit ne doit pas supposer qu'elle est entité de protocole d'application présidente à moins qu'elle ne constate qu'elle figure avec cet indicateur positionné dans le répertoire d'application. Cet indicateur ne s'applique qu'aux entités de protocole d'application actives. Si l'indicateur d'activité/inactivité est positionné sur inactif lors de l'inscription, l'indicateur de capacité de présidence est ignoré et le fournisseur GCC suppose que sa valeur est FALSE (faux).

*Canal de démarrage*: indicateur optionnel pouvant prendre les valeurs statique, dynamique, multidestinataire, dynamique privé ou identificateur dynamique d'utilisateur. Ce paramètre spécifie le type de canal MCS utilisé par l'entité de protocole d'application pour la séquence de démarrage. L'interprétation exacte de ce paramètre et toute exigence concernant son utilisation sont spécifiques du protocole d'application.

*Liste de capacités d'application non condensée*: paramètre optionnel permettant à des entités de protocole d'application de donner la liste de capacités, normalisées ou non, qui doivent être gérées dans le répertoire d'application comme faisant partie de l'enregistrement d'application de chaque entité de protocole d'application plutôt que de figurer sous une forme condensée dans la liste de capacités d'application. Chaque entrée de cette liste comprend un identificateur de capacité normalisée ou non ainsi qu'un champ destiné aux données spécifiques du protocole d'application.

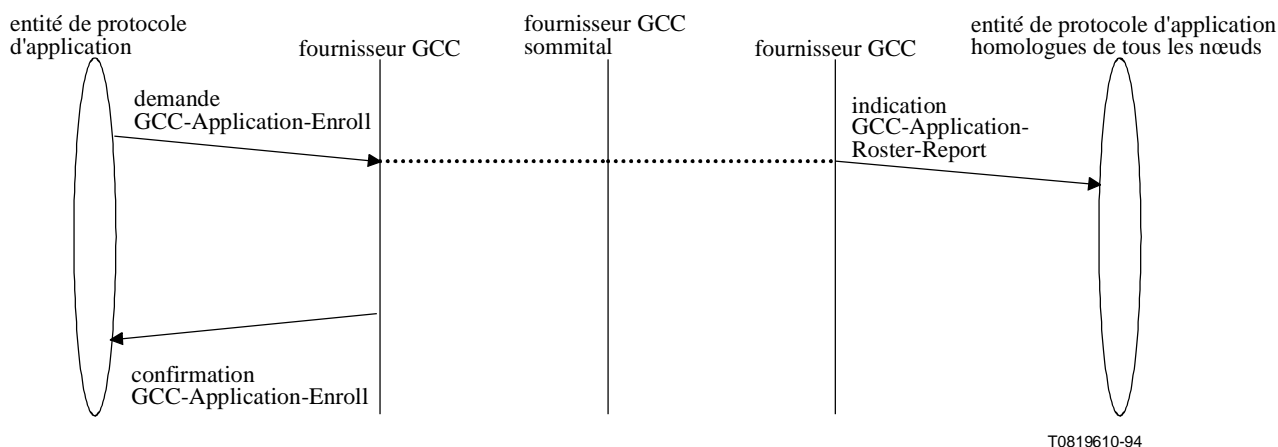
*Liste de capacités d'application*: liste optionnelle de capacités spécifiques du protocole d'application. Chaque capacité de la liste des capacités est marquée au moyen d'un indicateur de classe. Cet indicateur de classe indique la règle à appliquer pour déterminer l'ensemble commun de capacités. Le Tableau 7-24 donne la liste des classes de capacité.

*Indicateur d'inscription ou de résiliation*: indicateur utilisé pour signaler si l'entité de protocole d'application souhaite faire une addition ou une modification de son enregistrement d'application ou supprimer son enregistrement d'application. Lorsqu'une entité de protocole d'application reçoit une primitive d'indication GCC-Permission-To-Enroll (autorisation d'inscription), elle répondra par une primitive de demande GCC-Enroll-Request (demande d'inscription). Cet indicateur sera positionné sur "résiliation" si l'entité de protocole d'application ne souhaite pas s'inscrire.

*Identificateur d'entité de protocole d'application*: présent seulement en cas de succès d'une nouvelle inscription mais non dans les cas "déjà inscrit", "inscription infructueuse" et "résiliation". Identifie sans ambiguïté l'entité de protocole d'application au sein du nœud local. Un paramètre correspondant se trouve dans chaque entrée du répertoire d'application. Ceci permet à une entité de protocole d'application de déterminer si elle figure dans les mises à jour ultérieures du répertoire d'application.

*Identificateur de nœud*: présent uniquement dans le cas de réussite d'une inscription mais pas en cas d'échec de l'inscription ou de résiliation. Ce paramètre fournit à l'entité de protocole d'application l'identificateur de son nœud local. Ceci aide l'entité de protocole d'application à retrouver son entrée dans des répertoires d'application reçus lorsqu'elle cherche à déterminer si elle est effectivement inscrite dans la conférence. La combinaison de l'identificateur de nœud et de l'identificateur de l'entité de protocole d'application détermine sans ambiguïté l'entité de protocole d'application.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "n'a pas la permission de modifier l'identificateur d'utilisateur MCS".



**Figure 7-19 – Séquence de primitives – GCC-Application-Enroll**

### 7.3.3.3 GCC-Application-Roster-Report (compte rendu de répertoire d'application)

Cette primitive est utilisée par un fournisseur GCC pour envoyer une partie ou la totalité du répertoire d'application de conférence à chaque entité de protocole d'application inscrite ainsi qu'au contrôleur nodal. Une primitive d'indication GCC-Application-Roster-Report peut être produite automatiquement par un fournisseur GCC chaque fois que celui-ci découvre une modification dans une partie quelconque du répertoire. Cela se produit, par exemple, comme résultat d'une primitive de demande GCC-Application-Enroll (inscription d'application), lors de la détection d'une entité de protocole d'application qui se détache de la conférence ou lors de la détection d'un nœud quittant la conférence. Une primitive d'indication GCC-Application-Roster-Report sera émise par le contrôleur nodal à destination d'une entité de protocole d'application active lorsqu'une partie quelconque du répertoire concernant une entité homologue quelconque a été modifiée. Il est nécessaire, dans ce cas, de n'inclure que la partie du répertoire concernant la session de protocole d'application impliquée. Le fournisseur GCC peut également émettre à d'autres moments une primitive d'indication GCC-Application-Roster-Report destinée à une entité de protocole d'application, par exemple, lorsque des entrées autres que celles des entités de protocole d'application homologues ont été modifiées. Il peut également transmettre dans la primitive d'indication des parties du répertoire autres que celles correspondant aux entités de protocole d'application homologues. Les règles concernant une entité de protocole d'application active s'appliquent également à des entités de protocole d'application inactives lorsque ces entités se sont inscrites avec un identificateur de session spécifique. Le fournisseur GCC enverra à une entité de protocole d'application inactive, inscrite sans identificateur de session, une primitive d'indication GCC-Application-Roster-Report lorsque les contenus du répertoire ont été modifiés pour l'une quelconque des sessions de protocole d'application qui sont basées sur le même protocole d'application que celui de l'entité en question. Le Tableau 7-27 donne les paramètres et les types de primitives de ce service. Chaque paramètre spécifique du protocole d'application, à l'exception de l'identificateur de conférence, est répété séparément pour chaque protocole d'application de session lorsqu'il est envoyé au contrôleur nodal. La Figure 7-20 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-27 – Types de primitives et paramètres – GCC-Application-Roster-Report**

Paramètre	Indication
Identificateur de conférence	M
Répertoire d'application à jour	M

*Identificateur de conférence*: identificateur du domaine MCS correspondant à la conférence indiquée.

*Répertoire d'application à jour*: le répertoire d'application à jour contient l'information donnée par le Tableau 7-28 pour chaque session de protocole d'application mentionnée par cette indication.

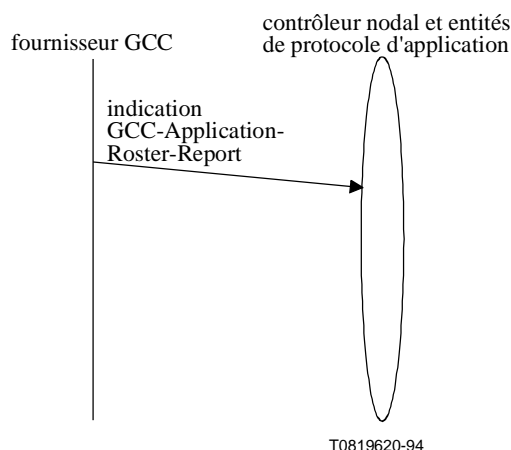
**Tableau 7-28 – Contenu du répertoire d'application à jour pour chaque session de protocole d'application**

<b>Paramètre</b>	<b>Description</b>
Clé de session	Clé de session, y compris l'identificateur de session éventuel, désignant la session de protocole d'application.
Indicateur de liste d'entités de protocole d'application à jour	Indicateur signalant si la liste d'entités de protocole d'application pour cette clé de session a été mise à jour dans ce compte rendu. Dans ce cas, le paramètre suivant contient la liste d'entités de protocole d'application.
Liste d'entités de protocole d'application (conditionnel)	Liste des entités de protocole d'application inscrites dans la conférence comme faisant partie de la session de protocole d'application ainsi qu'un enregistrement d'application pour chacune d'elles. Le contenu d'une entrée de la liste est donné par le Tableau 7-29.
Numéro d'instance	Numéro d'instance du répertoire d'application pour cette clé de session. Ce nombre de 16 bits est incrémenté modulo $2^{16}$ chaque fois que le contenu du répertoire d'application est modifié pour cette clé de session. Cela permet aux entités de protocole d'application d'exécuter des opérations en tenant compte d'un changement éventuel en cours dans un ensemble de capacités donné et d'éviter ainsi toute situation de compétition.
Indicateur d'ajout d'entités homologues	Indicateur signalant si une ou plusieurs entités de protocole d'application homologues ont été ajoutées au répertoire d'application pour un nœud quelconque depuis la dernière instance. Cet indicateur n'est pas mutuellement exclusif avec l'indicateur de suppression d'entités de protocole d'application homologues.
Indicateur de suppression d'entités homologues	Indicateur signalant si une ou plusieurs entités de protocole d'application homologues ont été supprimées du répertoire d'application pour un nœud quelconque depuis la dernière instance. Cet indicateur n'est pas mutuellement exclusif avec l'indicateur d'ajout d'entités de protocole d'application homologues.
Indicateur de mise à jour de liste de capacités d'application	Indicateur signalant si la liste de capacités d'application pour cette clé de session a été mise à jour dans ce compte rendu. Dans ce cas, le paramètre suivant indique la liste de capacités d'application.
Liste de capacités d'application (conditionnel)	Liste condensée de capacités d'application pour cette clé de session.



**Tableau 7-29 – Contenu d'une entrée de la liste d'entités de protocole d'application**

Paramètre	Description
Identificateur de nœud	Identificateur de nœud identifiant le nœud sur lequel s'est inscrite l'entité de protocole d'application.
Identificateur d'entité de protocole d'application	Identificateur désignant sans ambiguïté l'entité de protocole d'application au sein du nœud indiqué par l'identificateur de nœud.
Indicateur d'activité/inactivité	Indicateur signalant si l'entité de protocole d'application inscrite est active ou inactive.
Identificateur d'utilisateur d'application (conditionnel)	Identificateur d'utilisateur MCS associé avec l'entité de protocole d'application inscrite.
Indicateur de capacité de présidence d'opération	Indicateur signalant si l'entité de protocole d'application possède la capacité d'agir comme entité de protocole d'application présidente dans le cas où la spécification du protocole d'application associée définit des procédures exigeant une telle entité. Il existe au plus une entité de protocole d'application homologue par nœud possédant cet indicateur actif. Si un nœud devient le président de conférence, l'entité de protocole d'application désignée de ce nœud devient, si elle existe, l'entité de protocole d'application désignée pour la présidence pour cette session de protocole d'application.
Canal de démarrage (conditionnel)	Paramètre pouvant prendre, s'il est présent, les valeurs statique, dynamique, multidestinataire, dynamique privé ou identificateur d'utilisateur dynamique. Ce paramètre indique le type de canal MCS utilisé par l'entité de protocole d'application pour la séquence de démarrage. L'interprétation exacte de ce paramètre est spécifique du protocole d'application. Certains de ces types de canal peuvent ne pas être valides pour des protocoles d'application particuliers.
Liste de capacités d'application non condensée (conditionnel)	Liste de capacités, normalisées ou non, spécifiques d'un protocole d'application qui sont gérées dans le répertoire comme partie de l'enregistrement d'application pour chaque entité de protocole d'application.



**Figure 7-20 – Séquence de primitives – GCC-Application-Roster-Report**

### 7.3.3.4 GCC-Application-Roster-Inquire (interrogation de répertoire d'application)

Une entité de protocole d'application peut interroger au moyen de la primitive de demande GCC-Application-Roster-Inquire une partie du répertoire d'application de conférence correspondant à une seule session de protocole d'application, à un groupe de sessions ou à toutes les entités de

protocole d'application. Cette information est renvoyée par le fournisseur GCC dans la primitive de confirmation GCC-Application-Roster-Inquire. Le Tableau 7-30 donne les paramètres et les types de primitives de ce service. La Figure 7-21 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-30 – Types de primitives et paramètres – GCC-Application-Roster-Inquire**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de session	M	M(=)
Répertoire d'application		M
Résultat		M

*Identificateur de conférence*: l'identificateur de la conférence à laquelle se réfère la primitive.

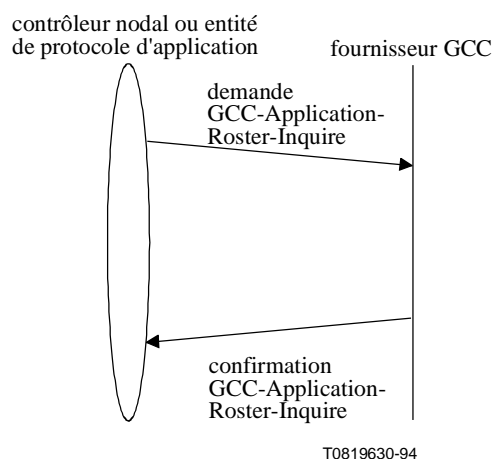
*Clé de session*: identificateur non ambigu d'une session de protocole d'application. Il peut s'agir d'une clé de session partielle ou d'une clé nulle indiquant l'ensemble des entités de protocole d'application.

*Répertoire d'application*: le répertoire d'application contient une liste d'entrées de répertoire pour chaque session de protocole d'application qui correspond à la demande selon le critère suivant. La clé de session de la session doit être identique à la clé de session de la demande pour un nombre d'octets déterminé par la longueur de la clé de la demande. Si des identificateurs de session font partie de la clé de session, ces derniers doivent également coïncider. Le Tableau 7-31 donne l'information renvoyée pour chaque entrée en correspondance.

**Tableau 7-31 – Contenu du répertoire d'application pour une session de protocole d'application**

Paramètre	Description
Clé de session	Clé de session, y compris l'identificateur de session éventuel, désignant la session de protocole d'application particulière.
Liste d'entités de protocole d'application	Liste des entités de protocole d'application inscrites dans la conférence comme faisant partie de cette session de protocole d'application ainsi qu'un enregistrement d'application pour chacune d'elles. Le contenu d'une entrée de la liste est donné par le Tableau 7-28.
Numéro d'instance	Numéro d'instance du répertoire d'application pour cette clé de session. Ce nombre de 16 bits est incrémenté modulo $2^{16}$ chaque fois que le contenu du répertoire d'application est modifié. Ceci permet aux entités de protocole d'application d'exécuter des opérations en tenant compte d'un changement éventuellement en cours dans un ensemble de capacités donné et d'éviter ainsi toute situation de compétition.
Liste de capacités d'application	La liste condensée de capacités d'application pour cette clé de session.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



**Figure 7-21 – Séquence de primitives – GCC-Application-Roster-Inquire**

### 7.3.3.5 GCC-Application-Invoke (invocation d'application)

Cette primitive peut être émise par une entité de protocole d'application ou le contrôleur nodal pour demander l'invocation d'une entité de protocole d'application ou d'une liste d'entités de protocole d'application sur un ensemble de nœuds donné. Au niveau du nœud destinataire, la forme indication de cette primitive est envoyée au contrôleur nodal qui peut invoquer les entités de protocole d'application mentionnées. Cette indication signale, pour chaque entité correspondante, l'intention de créer une nouvelle entité de protocole d'application avec la clé de session fournie par la demande. L'entité de protocole d'application créée s'inscrira dans la conférence dans l'état actif et deviendra partie prenante dans la session de protocole d'application correspondante. En variante, si une entité de protocole d'application inactive existe déjà avec une clé de session identique sur le nœud destinataire, la réception de cette indication signale le souhait de faire s'inscrire de nouveau l'entité dans l'état actif. Si une entité de protocole d'application déjà active existe avec une clé de session identique sur le nœud destinataire, cette indication sera ignorée. Le Tableau 7-32 donne les paramètres et les types de primitives de ce service. La Figure 7-22 indique la séquence d'événements correspondant à l'utilisation de cette primitive. On notera que les entités APE ne peuvent être invoquées qu'aux nœuds conventionnels ou comptés. De même, une demande GCC-Application-Invoke ne peut être lancée que d'un nœud conventionnel. Un nœud conventionnel doit avoir déjà créé la session indiquée dans la demande d'invocation, avant d'invoquer un nœud compté.

**Tableau 7-32 – Types de primitives et paramètres – GCC-Application-Invoke**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=)
Liste d'invocation d'entités de protocole d'application	M	M(=)	M(=)
Nœuds destinataires [liste d'identificateurs de nœud ou NULL (néant)]	O		
Identificateur du nœud origine de l'invocation		M	
Résultat			M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Liste d'invocation d'entités de protocole d'application*: liste d'une ou de plusieurs entités de protocole d'application à invoquer. Le contenu d'une entrée de la liste est donné par le Tableau 7-33.

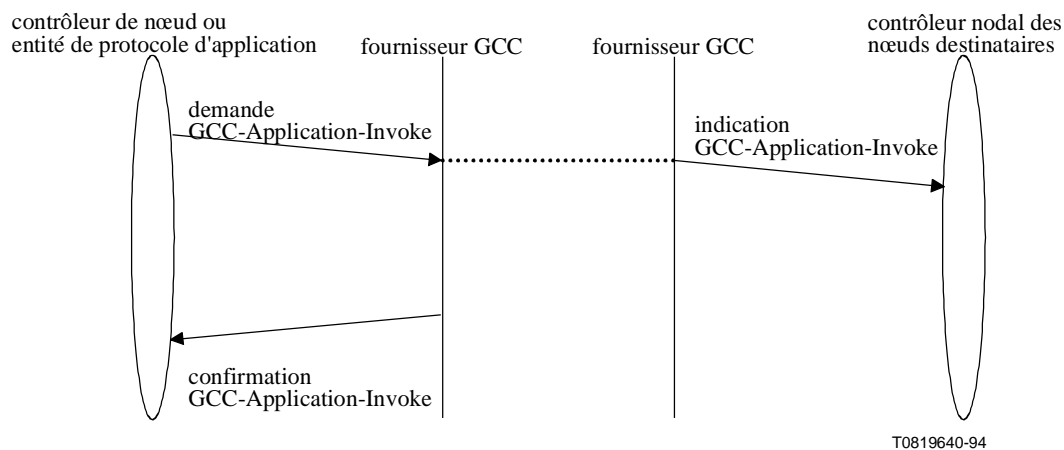
**Tableau 7-33 – Contenu des entrées dans la liste d'entités de protocole d'application**

Paramètre	Description
Clé de session	Clé de session, y compris l'identificateur de session éventuel, désignant la session de protocole d'application à invoquer.
Ensemble de capacités attendues	Liste optionnelle de capacités d'application, indiquant selon le format de la liste de capacités d'application, l'ensemble de capacités que les entités de protocole d'application à invoquer sont supposées posséder. Si une entité de protocole d'application d'un nœud particulier ne peut fournir cette capacité, elle ne sera pas invoquée sur ce nœud. Si ce paramètre n'est pas présent, il n'est imposé aucune contrainte à l'entité de protocole d'application invoquée. Il convient de noter que les définitions de classes de capacités sont utilisées pour interpréter l'ensemble des capacités attendues, mais que l'interprétation faite par fonction GCC diffère de celle de la liste de capacités d'application. Pour une capacité de la classe maximum sans signe, l'entité de protocole d'application invoquée doit avoir une capacité inférieure ou égale à la capacité spécifiée alors que pour une capacité de la classe minimum sans signe, l'entité de protocole d'application invoquée doit avoir une capacité supérieure ou égale à la capacité indiquée. Pour une capacité de la classe logique, l'entité de protocole d'application invoquée doit posséder la capacité indiquée.
Canal de démarrage (conditionnel)	Paramètre pouvant prendre les valeurs statique, dynamique, multidestinataire, dynamique privé ou identificateur d'utilisateur dynamique. Ce paramètre spécifie le type de canal MCS que l'entité de protocole d'application est censée utiliser lors de l'invocation. Certains de ces canaux peuvent ne pas être valides pour des entités de protocole d'application particulières. Si une valeur non valide est utilisée, l'entité de protocole d'application ne sera pas invoquée.
Indicateur d'invocation obligatoire/optionnel	Indicateur signalant si l'invocation de cette entité de protocole d'application conditionne l'invocation des autres entités de protocole d'application de cette liste. Cet indicateur est utilisé par le nœud qui est à l'origine de l'invocation afin d'indiquer que les nœuds destinataires ne doivent invoquer les entités de protocole d'application de la liste que si toutes les entités de protocole d'application marquées comme obligatoires peuvent être invoquées avec succès.

*Nœuds destinataires (liste d'identificateurs de nœud ou NULL):* liste d'identificateurs de nœud identifiant des fournisseurs GCC destinataires de la demande ou néant indiquant l'ensemble des nœuds de la conférence.

*Identificateur du nœud origine de l'invocation:* identificateur du nœud qui a lancé la primitive de demande GCC-Application-Invoke.

*Résultat:* indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



**Figure 7-22 – Séquence de primitives – GCC-Application-Invoke**

## 7.4 Le référentiel d'application

Le référentiel d'application est un composant fonctionnel du service GCC. Le référentiel offre aux entités de protocole d'application un ensemble de fonctions mettant en œuvre une base de données centrale localisée sur le fournisseur GCC sommital. Le référentiel a un contenu unique pour une conférence donnée. La signification de l'information stockée dans la base de données du référentiel est définie par les protocoles d'application. Le référentiel central peut fournir une assistance en établissant une communication entre des entités de protocole d'application. Il peut aider des entités de protocole d'application à trouver un identificateur de canal commun pour leur communication, un identificateur de jeton commun pour gérer un accès exclusif ou un positionnement commun de paramètre (un paramètre est une entrée de référentiel qui a une utilisation spécifique dans un protocole d'application). Le référentiel propose une solution autre que l'attribution statique des identificateurs de canal et de jeton et des autres valeurs de paramètres. Comme il prend en charge une recherche dynamique, le référentiel facilite l'introduction de caractéristiques nouvelles, normalisées ou non, qui permettent d'améliorer les conférences audiographiques et audiovisuelles. Le référentiel fournit également un service de caractère général pour l'attribution de "poignées" numériques non ambiguës pour l'ensemble d'une conférence.

### 7.4.1 Clés de référentiel

Les index utilisés pour stocker et retrouver les entrées dans la base de données du référentiel sont des clés fournies par les protocoles d'application. Des clés normalisées et non normalisées sont autorisées. Les clés normalisées sont attribuées par des Recommandations telles que la présente. Des clés non normalisées peuvent être propriétaires. Dans les deux cas, les clés sont structurées de manière à éviter les conflits dans le choix de leurs valeurs.

Les clés de référentiel sont des clés de session définies au § 7.3.1 associées à un identificateur de ressource. La clé de session utilisée pour toute opération de référentiel nécessitant une telle clé doit être identique à la clé de session d'une session de protocole d'application figurant dans le référentiel d'application avec des membres pouvant être actifs ou inactifs. L'identificateur de ressource est une chaîne d'octets de type ASN.1. L'identificateur de ressource permet à une entité de protocole d'application donnée d'utiliser des clés de référentiel multiples. Les valeurs des identificateurs de ressource sont définies par les spécifications des protocoles d'application qui les utilisent.

### 7.4.2 Propriété et persistance

Le référentiel attribue la propriété d'une clé de référentiel à la première entité de protocole d'application qui l'utilise pour mémoriser un item dans l'entrée correspondante au moyen d'une des primitives de demande GCC-Registry-Register-Channel, GCC-Registry-Assign-Token ou GCC-Registry-Set-Parameter (enregistrement de canal, attribution de jeton ou positionnement de

paramètre par le référentiel). Un nœud conventionnel est la seule catégorie nodale autorisée pour avoir la propriété d'une entrée de référentiel.

Le propriétaire d'un élément de référentiel du type "paramètre" peut spécifier, lors de sa création, la classe des entités de protocole d'application ayant l'autorisation de modifier cette entrée. Le propriétaire peut spécifier des autorisations de modification "propriétaire", "session" ou "public" indiquant respectivement que le paramètre peut être modifié uniquement par le propriétaire (comme c'est le cas pour les canaux et les jetons) par l'ensemble des entités de protocole d'application homologues appartenant à la même session de protocole d'application ou par toute entité de protocole d'application appartenant à la conférence. L'identité du propriétaire d'un paramètre n'est pas modifiée lors d'une modification effectuée par une entité de protocole d'application non propriétaire. Le propriétaire d'un paramètre peut redéfinir à tout moment la portée des autorisations de modification. Si un autre nœud tente de changer les autorisations de modification par l'émission d'une primitive de demande GCC-Registry-Set-Parameter, les autorisations ne seront pas modifiées, mais le paramètre est positionné sur la valeur demandée si le demandeur possède le droit de modification.

La propriété de l'entrée est exigée pour supprimer un élément de type quelconque, "canal", "jeton" ou "paramètre". Une entrée supprimée n'a pas de propriétaire et peut être reprise par une autre entité de protocole d'application.

Les entrées de référentiel ne sont pas supprimées automatiquement lorsque leur propriétaire se désinscrit de la conférence. Leur contenu reste inchangé et persiste indéfiniment. La propriété d'une entrée est toutefois supprimée lorsque le propriétaire se désinscrit. Cela permet à une autre entité de protocole d'application de modifier (uniquement dans le cas d'une entrée paramètre) ou de supprimer une entrée sans propriétaire lorsque son utilisation est devenue caduque. La première entité de protocole d'application qui demande à stocker un élément dans une entrée sans propriétaire en devient le nouveau propriétaire.

L'entrée de référentiel est supprimée automatiquement lorsque toutes les entités de protocole d'application correspondant à la clé de session utilisée pour déterminer la clé de référentiel pour cette entrée ont été supprimées.

### **7.4.3 Attribution dynamique**

Le service MCS fait une distinction entre identificateurs de canal statiques et dynamiques. Les identificateurs statiques comprennent les identificateurs d'utilisateur, les identificateurs de canal privé et les identificateurs de canal affecté. Les identificateurs dynamiques de canal sont créés et supprimés directement par des entités de protocole d'application, indépendamment de l'usage que celles-ci font des services de référentiel. Le référentiel d'application stocke les identificateurs dynamiques de canal dans un référentiel central afin de permettre leur accès par d'autres entités de protocole d'application. Il n'effectue pas de test ou d'opération sur ces identificateurs de canal à l'aide de primitives MCS.

Le service MCS ne fait pas de distinction entre les identificateurs de jeton statiques ou dynamiques. Une distinction artificielle est créée par fonction GCC dans le but de conserver une sémantique unique. Les identificateurs de jeton de 1 à 16 383 sont définis comme étant statiques et sont réservés pour une affectation par d'autres spécifications. Les identificateurs de jeton de 16 384 à 65 535 sont définis comme étant dynamiques et sont attribués à la demande par le fournisseur GCC sommital à la suite de la création d'une entrée dans la base de données du référentiel. Lorsque l'entrée correspondante est supprimée, l'identificateur de jeton qu'elle contient redevient disponible pour une nouvelle attribution. Le référentiel n'invoque pas de primitives MCS de prise ou de libération pour son assignation d'identificateurs de jeton. Son rôle se limite au choix et à la diffusion des valeurs spécifiques d'identificateur. Les entités de protocole d'application peuvent utiliser librement les identificateurs de jeton ainsi attribués.

#### 7.4.4 Description de services abstraits

La liste qui suit récapitule les primitives définies dans le présent sous-paragraphe et en résume les fonctions:

- GCC-Registry-Register-Channel (*enregistrement de canal par le référentiel*) – Utilisée par une entité de protocole d'application pour enregistrer l'identificateur d'un canal MCS dynamique. Les entités de protocole d'application peuvent examiner l'entrée au moyen de la primitive GCC-Registry-Retrieve-Entry pour déterminer si un nœud a déjà enregistré l'identificateur de canal et le cas échéant en retrouver la valeur. Seuls les nœuds conventionnels sont autorisés à enregistrer des canaux.
- GCC-Registry-Assign-Token (*attribution de jeton par le référentiel*) – Utilisée par une entité de protocole d'application pour allouer un jeton dynamique et enregistrer l'identificateur de jeton attribué. Les entités de protocole d'application peuvent examiner l'entrée au moyen de la primitive GCC-Registry-Retrieve-Entry pour déterminer si un nœud a déjà enregistré l'identificateur de canal et le cas échéant en retrouver la valeur. Seuls les nœuds conventionnels sont autorisés à attribuer un jeton.
- GCC-Registry-Set-Parameter (*positionnement de paramètre par le référentiel*) – Utilisée par une entité de protocole d'application pour positionner dans la base de données une valeur qui peut être examinée ou modifiée par un nœud quelconque d'une conférence. Seuls les nœuds conventionnels sont autorisés à positionner un paramètre du référentiel.
- GCC-Registry-Retrieve-Entry (*recherche d'entrée par le référentiel*) – Utilisée par une entité de protocole d'application pour extraire le contenu d'une entrée de la base de données du référentiel.
- GCC-Registry-Delete-Entry (*suppression d'entrée par le référentiel*) – Utilisée par une entité de protocole d'application pour supprimer une entrée du référentiel. Seuls les nœuds conventionnels sont autorisés à supprimer une entrée du référentiel.
- GCC-Registry-Monitor (*supervision du référentiel*) – Utilisée par une entité de protocole d'application pour mettre en service ou hors service la supervision d'une entrée du référentiel. Une fois la supervision mise en service, la primitive d'indication correspondante notifie à l'entité de protocole d'application qui a fait la demande toutes les modifications effectuées sur cette entrée, y compris sa suppression. Toute catégorie nodale peut superviser le référentiel.
- GCC-Registry-Allocate-Handle (*attribution de poignée par le référentiel*) – Utilisée pour produire un nombre 32 bits appelé "poignée" (ou "pointeur") contenant une valeur non ambiguë dans le domaine d'une conférence donnée. Toute catégorie nodale peut attribuer une poignée.

Pour un demandeur donné de service de référentiel, l'ordre de succession de la primitive de demande, de l'exécution de l'action résultante par le fournisseur GCC sommital (en cas de succès), et de la primitive de réponse associée est préservé.

##### 7.4.4.1 GCC-Registry-Register-Channel (enregistrement de clé de référentiel)

La primitive de demande GCC-Registry-Register-Channel peut être émise par une entité de protocole d'application située dans un nœud conventionnel afin d'informer des entités de protocole d'application situées sur d'autres nœuds qu'un canal MCS donné a été désigné pour être utilisé par l'entité de protocole d'application de la manière indiquée par la clé de référentiel. Une fois qu'une entité de protocole d'application quelconque a effectué l'enregistrement en utilisant une clé donnée, des entités de protocole d'application situées sur d'autres nœuds peuvent déterminer si un canal a été enregistré pour cette clé et, le cas échéant, obtenir la valeur de cet identificateur de canal, par l'émission d'une primitive de demande GCC-Registry-Retrieve-Entry (recherche de clé de référentiel) indiquant la clé de référentiel en question. Une fois que le jeton a été attribué, une

tentative d'une entité de protocole d'application (y compris le propriétaire) pour faire attribuer un jeton en utilisant la même clé de référentiel, sera rejetée avec l'indication que la clé de référentiel est déjà utilisée. Le Tableau 7-34 donne les paramètres et les types de primitives de ce service. La Figure 7-23 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-34 – Types de primitives et paramètres –  
GCC-Registry-Register-Channel**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de référentiel	M	M(=)
Identificateur de canal	M	
Élément de référentiel		C
Propriétaire		C
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Clé de référentiel*: index de stockage de l'élément dans la base de données.

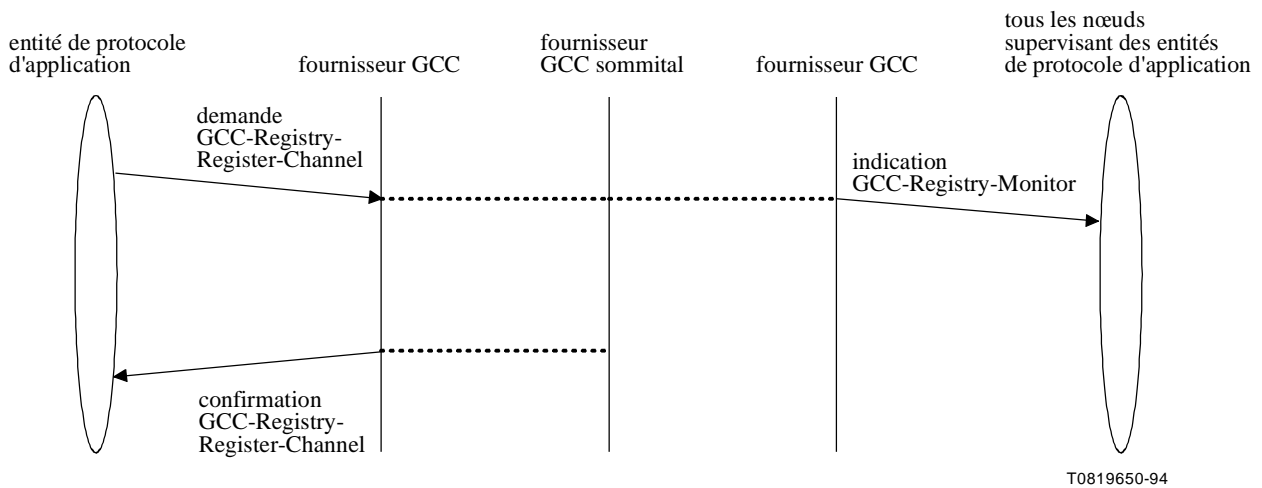
*Identificateur de canal*: identificateur dynamique de canal (identificateur d'utilisateur, privé ou attribué) fourni par l'entité de protocole d'application.

*Élément de référentiel*: valeur de l'entrée après l'exécution de la demande. Si la demande aboutit, il s'agit de la valeur du paramètre. En cas d'échec causé par un type d'index inconsistant ou parce que l'entrée a déjà un propriétaire, ce paramètre contient la valeur avant la demande. La valeur est soit un identificateur de canal, un identificateur de jeton, une valeur de paramètre ou néant si l'entrée est vacante. Ce paramètre indique à la fois le type et, le cas échéant, la valeur. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Propriétaire*: paramètre indiquant le propriétaire actuel de cette entrée de référentiel. Si le propriétaire existe, ce paramètre indique l'identificateur du nœud sur lequel réside le propriétaire et l'identificateur de l'entité de protocole d'application propriétaire. Ce paramètre indique également si l'entrée n'a pas de propriétaire. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Résultat*: indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "référentiel plein", "index existant déjà", "type inconsistant" ou "demandeur non valide".





**Figure 7-23 – Séquence de primitives – GCC-Registry-Register-Channel**

#### 7.4.4.2 GCC-Registry-Assign-Token (attribution de jeton de référentiel)

La primitive de demande GCC-Registry-Assign-Token peut être émise par une entité de protocole d'application située dans un nœud conventionnel pour attribuer un jeton associé à une clé de référentiel donnée. En cas de succès, l'identificateur de jeton est renvoyé comme paramètre dans la primitive de confirmation. Une fois qu'une entité de protocole d'application quelconque a fait attribuer un jeton en utilisant une clé de référentiel donnée, des entités de protocole d'application d'autres nœuds peuvent déterminer si un jeton a été attribué à cette clé et, le cas échéant, la valeur de l'identificateur de jeton en émettant la primitive de demande GCC-Registry-Retrieve-Entry (recherche d'entrée par le référentiel) indiquant la clé de référentiel concernée. Une fois que le jeton a été attribué, une tentative d'une entité de protocole d'application quelconque (y compris le propriétaire) pour faire attribuer un jeton en utilisant la même clé de référentiel, sera rejetée avec l'indication que la clé de référentiel est déjà utilisée. Le Tableau 7-35 donne les paramètres et les types de primitives de ce service. La Figure 7-24 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-35 – Types de primitives et paramètres – GCC-Registry-Assign-Token**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de référentiel	M	M(=)
Identificateur de jeton		C
Élément de référentiel		C
Propriétaire		C
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Clé de référentiel*: index de stockage de l'élément dans la base de données.

*Identificateur de jeton*: identificateur dynamique de jeton, supérieur ou égal à 16 384, attribué par le fournisseur GCC sommital. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Élément de référentiel*: valeur de l'entrée après l'exécution de la demande. Si la demande aboutit, il s'agit de la valeur du paramètre. En cas d'échec causé par un type d'index incorrect ou parce que

l'entrée a déjà un propriétaire, ce paramètre contient la valeur avant la demande. La valeur est soit un identificateur de canal, un identificateur de jeton, une valeur de paramètre ou néant si l'entrée est vacante. Ce paramètre indique à la fois le type et, le cas échéant, la valeur. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Propriétaire:* paramètre indiquant le propriétaire actuel de cette entrée de référentiel. Si le propriétaire existe, ce paramètre indique l'identificateur du nœud sur lequel réside le propriétaire et l'identificateur de l'entité de protocole propriétaire. Ce paramètre indique également si l'entrée n'a pas de propriétaire. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Résultat:* indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "référentiel plein", "index existant déjà", "type inconsistant" ou "demandeur non valide".

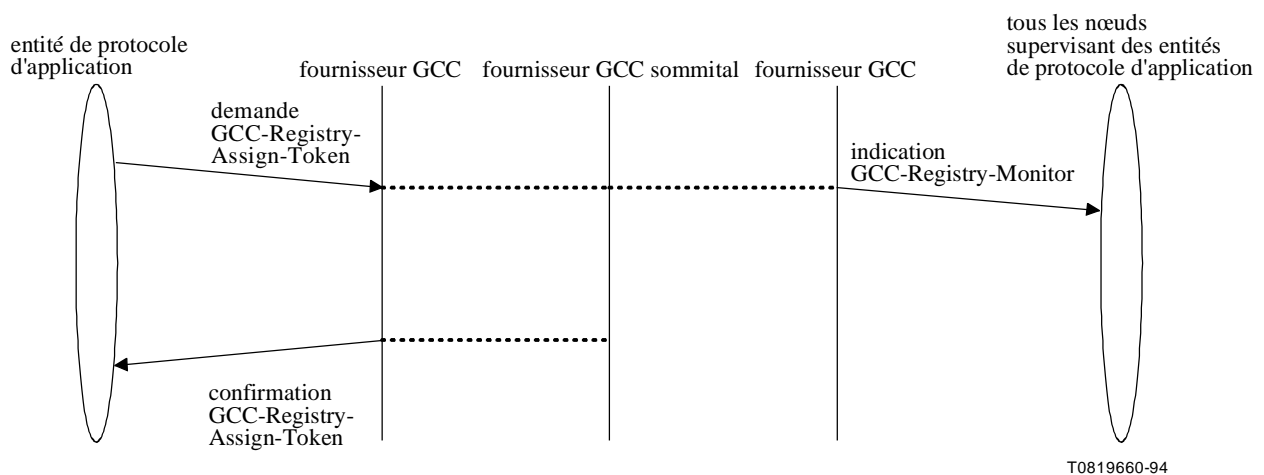


Figure 7-24 – Séquence de primitives – GCC-Registry-Assign-Token

#### 7.4.4.3 GCC-Registry-Set-Parameter (positionnement de paramètre par le référentiel)

La primitive de demande GCC-Registry-Set-Parameter peut être émise par une entité de protocole d'application située dans un nœud conventionnel afin de positionner ou de modifier la valeur d'un paramètre de référentiel. Si l'entrée de référentiel a été désignée au moyen de la primitive de demande GCC-Registry-Monitor (supervision du référentiel) comme devant être supervisée, toute primitive de demande GCC-Registry-Set-Parameter qui réussit induit une primitive d'indication GCC-Monitor-Indication (indication de supervision). Cette indication est envoyée aux entités de protocole d'application de tous les nœuds ayant activé la supervision de l'entrée de référentiel. Si une entrée de référentiel existe pour une clé donnée, une demande de positionnement de paramètre ne sera acceptée que si l'entrée est du type paramètre. Le Tableau 7-36 donne les paramètres et les types de primitives de ce service. La Figure 7-25 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-36 – Types de primitives et paramètres –  
GCC-Registry-Set-Parameter**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de référentiel	M	M(=)
Valeur de paramètre	M	
Élément de référentiel		C
Propriétaire		C
Droits de modification	O	C
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Clé de référentiel*: index de stockage de l'élément dans la base de données.

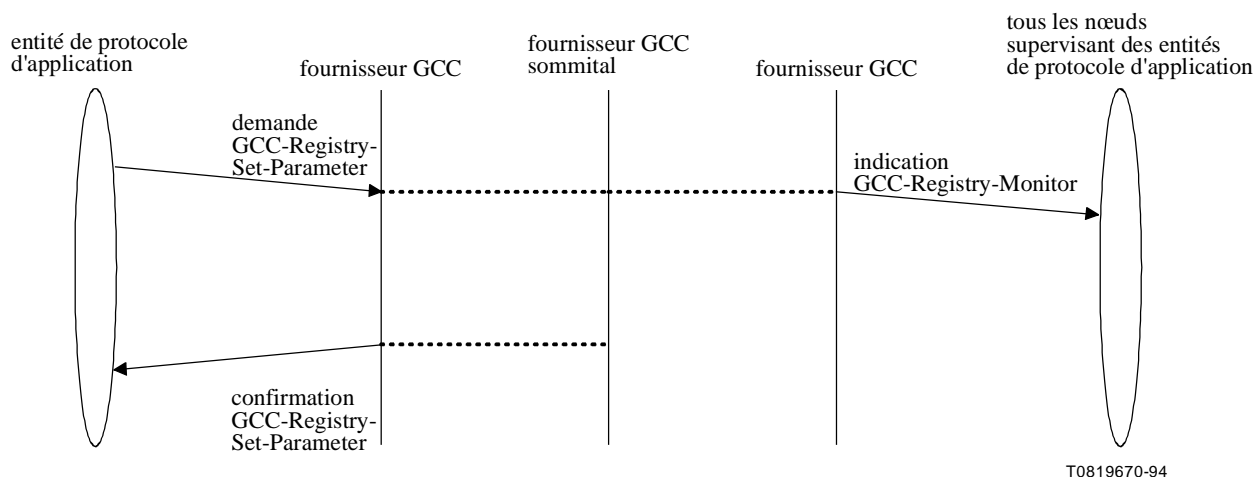
*Valeur de paramètre*: chaîne d'octets indiquée par l'entité de protocole d'application.

*Élément de référentiel*: valeur de l'entrée après l'exécution de la demande. Si la demande aboutit, il s'agit de la valeur du paramètre. En cas d'échec causé par un type d'index inconsistant ou parce que l'entrée a déjà un propriétaire, ce paramètre contient la valeur avant la demande. La valeur est soit un identificateur de canal, un identificateur de jeton, une valeur de paramètre ou néant si l'entrée est vacante. Ce paramètre indique à la fois le type et, le cas échéant, la valeur. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Propriétaire*: paramètre indiquant le propriétaire actuel de cette entrée de référentiel. Si le propriétaire existe, ce paramètre indique l'identificateur du nœud sur lequel réside le propriétaire et l'identificateur de l'entité de protocole d'application propriétaire. Ce paramètre indique également si l'entrée n'a pas de propriétaire. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Droits de modification*: paramètre optionnel indiquant la classe des entités de protocole d'application ayant l'autorisation de modifier cette entrée. Les valeurs possibles sont: "propriétaire", "session" ou "public". Le positionnement sur "propriétaire" spécifie que seul le propriétaire éventuel peut modifier cette entrée. Le positionnement sur "session" spécifie que toute entité de protocole d'application faisant partie de la même session de protocole d'application que le propriétaire peut modifier cette entrée. Le positionnement sur "public" spécifie que toute entité de protocole d'application inscrite dans la conférence peut modifier cette entrée. Si ce paramètre n'est pas présent lors de la création de l'entrée, la valeur "public" est prise par défaut. S'il n'est pas présent à d'autres occasions, la valeur reste inchangée. Seul le propriétaire éventuel peut changer les droits de modification. Si une entité non propriétaire tente de changer les droits de modification, ce changement ne sera pas effectué, cependant toute modification de la valeur du paramètre sera exécutée si le demandeur possède le droit de modification. Dans la primitive de confirmation, ce paramètre indique l'état en vigueur des droits de modification, indépendamment de la valeur qu'il avait dans la primitive de demande. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Résultat*: indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "référentiel plein", "type incohérent" ou "demandeur non valide".



T0819670-94

**Figure 7-25 – Séquence de primitives – GCC-Registry-Set-Parameter**

#### 7.4.4.4 GCC-Registry-Retrieve-Entry (recherche d'entrée par le référentiel)

La primitive de demande GCC-Registry-Retrieve-Entry peut être émise par une entité de protocole d'application afin d'examiner le contenu d'une entrée de référentiel donnée. Cette primitive peut être émise à tout instant. Elle fournit le contenu de l'entrée et indique également si l'entrée est un identificateur de canal, un identificateur de jeton, un paramètre ou si l'entrée est vide. Le Tableau 7-37 donne les paramètres et les types de primitives de ce service. La Figure 7-26 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-37 – Types de primitives et paramètres – GCC-Registry-Retrieve-Entry**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de référentiel	M	M(=)
Élément de référentiel		C
Propriétaire		C
Droits de modification		C
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Clé de référentiel*: index de stockage de l'élément dans la base de données.

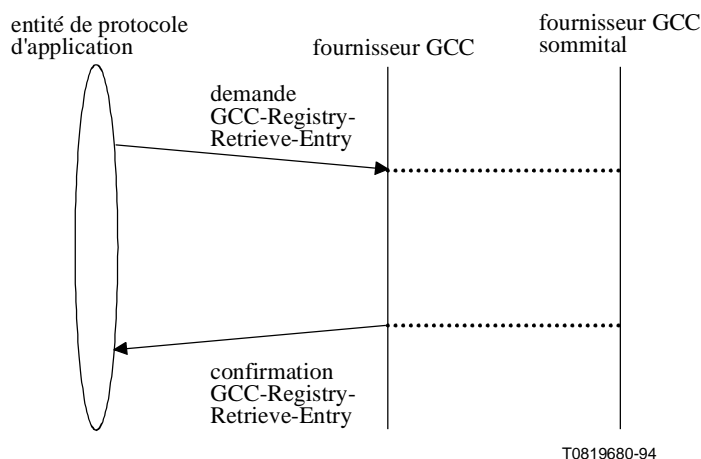
*Élément de référentiel*: identificateur de canal, identificateur de jeton, valeur de paramètre ou néant si l'entrée est vacante. Ce paramètre indique à la fois le type et, le cas échéant, la valeur. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Propriétaire*: paramètre indiquant le propriétaire actuel de cette entrée de référentiel. Si le propriétaire existe, ce paramètre indique l'identificateur du nœud sur lequel réside le propriétaire et l'identificateur de l'entité de protocole d'application propriétaire. Ce paramètre indique également si l'entrée n'a pas de propriétaire. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Droits de modification*: ce paramètre qui n'existe que pour des entrées du type "paramètre" indique la classe des entités de protocole d'application autorisées à modifier la valeur de cette clé de référentiel. Les valeurs possibles sont: "propriétaire", "session" ou "public". Le positionnement sur "propriétaire" spécifie que seul le propriétaire peut modifier cette entrée (tant que l'entrée a

effectivement un propriétaire). Le positionnement sur "session" spécifie que toute entité de protocole d'application faisant partie de la même session de protocole d'application que le propriétaire peut modifier cette entrée. Le positionnement sur "public" spécifie que toute entité de protocole d'application inscrite dans la conférence peut modifier cette entrée. Si ce paramètre n'est pas présent lors de la création de l'entrée, la valeur de public est prise par défaut. S'il n'est pas présent à d'autres occasions, la valeur reste inchangée.

*Résultat*: indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "entrée non trouvée" ou "conférence non valide".



**Figure 7-26 – Séquence de primitives – GCC-Registry-Retrieve-Entry**

#### 7.4.4.5 GCC-Registry-Delete-Entry (suppression d'entrée par le référentiel)

La primitive de demande GCC-Registry-Delete-Entry peut être émise par une entité de protocole d'application afin de supprimer une entrée de référentiel. Seule l'entité de protocole d'application propriétaire d'une entrée de référentiel a le droit de la supprimer (à moins que le dernier propriétaire ne se soit déconnecté de la conférence et qu'aucun nouveau propriétaire n'ait été attribué). Cette primitive ne peut être émise que d'un nœud conventionnel. Si l'entrée de référentiel a été désignée au moyen de la primitive de demande GCC-Registry-Monitor (supervision du référentiel) comme devant être supervisée, une primitive de demande GCC-Registry-Delete-Entry qui réussit induit une primitive d'indication GCC-Monitor-Indication avec un paramètre "élément de référentiel" de valeur néant. Cette indication est envoyée aux entités de protocole d'application de tous les nœuds ayant activé la supervision de cette entrée de référentiel pour signaler que l'entrée a été supprimée. Le Tableau 7-38 donne les paramètres et les types de primitives de ce service. La Figure 7-27 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

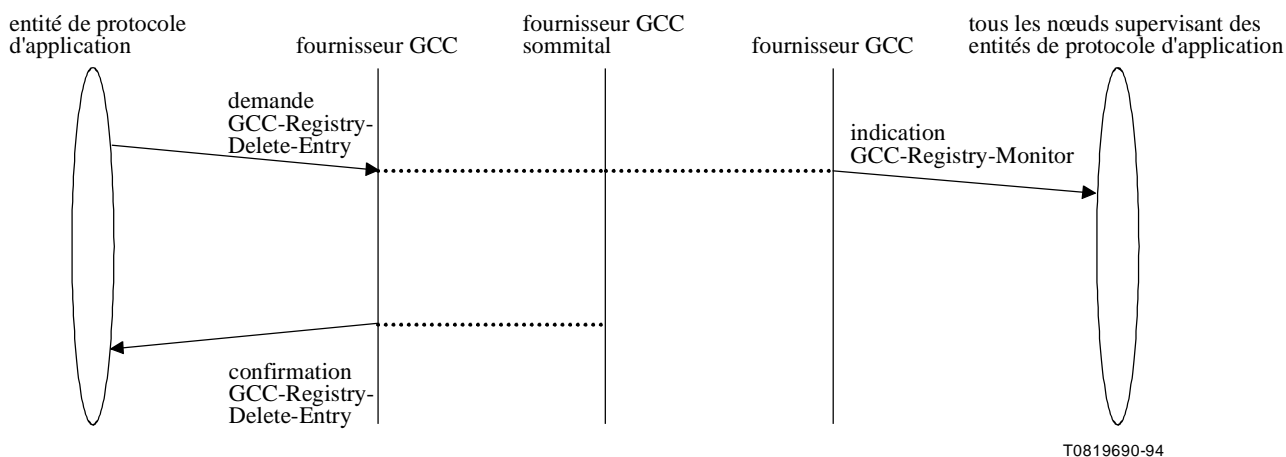
**Tableau 7-38 – Types de primitives et paramètres – GCC-Registry-Delete-Entry**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Clé de référentiel	M	M(=)
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Clé de référentiel*: index de stockage de l'élément dans la base de données.

*Résultat:* indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "index déjà propriété d'un autre fournisseur GCC".



**Figure 7-27 – Séquence de primitives – GCC-Registry-Delete-Entry**

#### 7.4.4.6 GCC-Registry-Monitor (supervision du référentiel)

Le service GCC fournit un procédé permettant de superviser d'une manière continue des entrées de référentiel de tous les types afin de déterminer si elles ont été changées par modification, suppression, changement de propriétaire ou – dans le cas d'une entrée du type "paramètre" – par changement de droits de modification, sans qu'il soit nécessaire de les scruter en permanence. La primitive de demande GCC-Registry-Monitor peut être émise par une entité de protocole d'application pour mettre en service ou hors service la supervision d'une entrée de référentiel donnée. Lorsque la supervision est en service, l'entité de protocole d'application ayant fait la demande reçoit, par l'intermédiaire d'une primitive d'indication GCC-Registry-Monitor, une notification de tous les changements faits pour cette entrée. Une indication est générée à la suite de toute modification du contenu de l'entrée de référentiel ou de sa suppression. Seules des entrées de référentiel qui existent peuvent être supervisées. Si une entrée supprimée est créée de nouveau, la demande de supervision doit être renouvelée pour recommencer la supervision. Le Tableau 7-39 donne les paramètres et les types de primitives de ce service. La Figure 7-28 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

NOTE – Des réalisations spécifiques de fournisseur GCC peuvent choisir de ne pas conserver la trace des entités de protocole d'application ayant mis en service ou hors service la supervision de chaque entrée. Dans ce cas, les entités de protocole d'application peuvent recevoir des indications GCC-Registry-Monitor concernant des entrées pour lesquelles elles n'ont pas demandé de supervision ou des entrées pour lesquelles elles ont mis la supervision explicitement hors service.

**Tableau 7-39 – Types de primitives et paramètres – GCC-Registry-Monitor**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Indicateur de supervision	M		M(=)
Clé de référentiel	M	M	M(=RQ)
Élément de référentiel		M	
Propriétaire		M	
Droits de modification		C	

Résultat			M
----------	--	--	---

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Indicateur de supervision*: TRUE (vrai) si les indications sur le contenu du référentiel doivent être fournies, FALSE (faux) pour supprimer les indications.

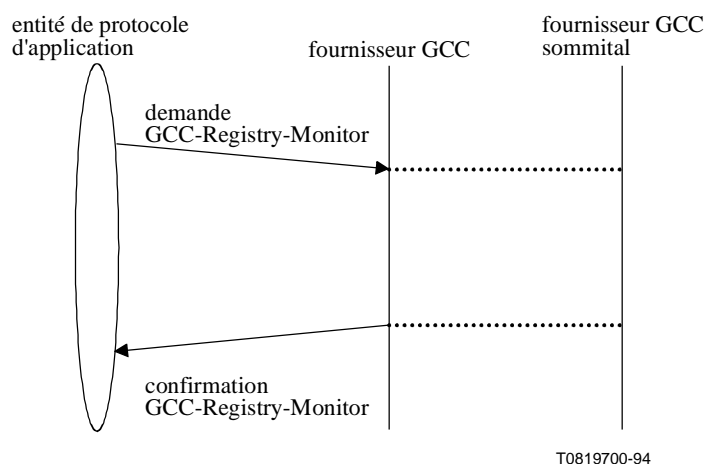
*Clé de référentiel*: index d'une entrée de référentiel dans la base de données. Pour une primitive d'indication, il s'agit de l'entrée qui a été modifiée, pour une primitive de demande/confirmation il s'agit de l'entrée à superviser.

*Élément de référentiel*: identificateur de canal, identificateur de jeton ou valeur de paramètre, ou vide indiquant une suppression. Le type éventuel et la valeur éventuelle sont indiqués par ce paramètre.

*Propriétaire*: paramètre indiquant le propriétaire actuel de cette entrée de référentiel. Si le propriétaire existe, ce paramètre indique l'identificateur du nœud sur lequel réside le propriétaire et l'identificateur de l'entité de protocole d'application propriétaire. Ce paramètre indique également si l'entrée n'a pas de propriétaire. Ce paramètre n'est pas présent si la confirmation est une condition d'erreur générée localement.

*Droits de modification*: ce paramètre qui n'existe que pour des entrées du type "paramètre" indique la classe des entités de protocole d'application autorisées à modifier la valeur de cette entrée de référentiel. Ce paramètre peut prendre les valeurs "propriétaire", "session" ou "public". Le positionnement sur "propriétaire" spécifie que seul le propriétaire peut modifier cette entrée tant que l'entrée a effectivement un propriétaire. Le positionnement sur "session" spécifie que toute entité de protocole d'application faisant partie de la même session de protocole d'application que le propriétaire peut modifier l'entrée. Le positionnement sur "public" spécifie que toute entité de protocole d'application inscrite dans la conférence peut modifier l'entrée. Si ce paramètre n'est pas présent lors de la création de l'entrée, la valeur de public est prise par défaut. S'il n'est pas présent à d'autres occasions, la valeur reste inchangée.

*Résultat*: indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "l'entrée n'existe pas" ou "conférence non valide".



**Figure 7-28 – Séquence de primitives – GCC-Registry-Monitor**

#### 7.4.4.7 GCC-Registry-Allocate-Handle (attribution de poignée par le référentiel)

La primitive de demande GCC-Registry-Allocate-Handle peut être émise par une entité de protocole d'application située dans un nœud conventionnel pour demander qu'une valeur numérique ou une liste de valeurs, globalement non ambiguë dans le domaine d'une conférence, soit fournie à

cette entité de protocole d'application. Les poignées sont attribuées selon une suite numérique croissante dans l'ordre dans lequel les demandes sont reçues par le fournisseur GCC sommital. Les blocs de poignées sont également attribués dans un ordre numérique croissant. En conséquence, seule la première poignée d'un bloc est renvoyée si le nombre de poignées est supérieur à un. Le Tableau 7-40 donne les paramètres et les types de primitives de ce service. La Figure 7-29 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-40 – Types de primitives et paramètres – GCC-Registry-Allocate-Handle**

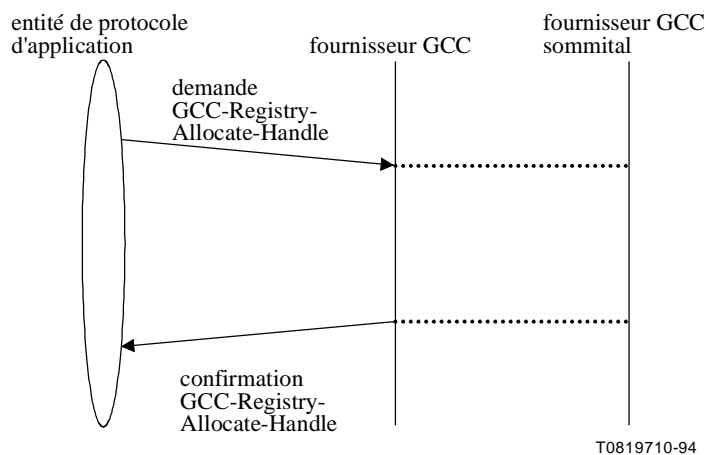
Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Nombre de poignées	M	M(=)
Première poignée		M
Résultat		M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Nombre de poignées*: nombre de poignées dont l'attribution est demandée et qui sont renvoyées dans la primitive de confirmation. Le domaine de ce paramètre va de 1 à 1024.

*Première poignée*: valeur entière de 32 bits sans signe. Si le nombre de poignées demandées est égal à 1, il s'agit de la valeur de la poignée attribuée. Si le nombre de poignées est supérieur à un, l'ensemble des poignées attribuées occupe des valeurs contiguës (modulo  $2^{32}$ ) qui vont de la valeur de la première poignée à la valeur (première poignée + nombre de poignées) mod  $2^{32}$ .

*Résultat*: indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "pas de poignées disponibles" ou "trop de poignées demandées".



**Figure 7-29 – Séquence de primitives – GCC-Registry-Allocate-Handle**

## 7.5 Présidence de conférence

Le service GCC fournit un procédé permettant à un nœud conventionnel de devenir le président d'une conférence. Le service GCC utilise un jeton pour déterminer si une conférence est présidée ou non. Le contrôleur nodal d'un nœud peut émettre une primitive de demande GCC-Conductor-Assign (assignation de président) afin d'acquérir le jeton de présidence. La conférence est placée en mode présidé lorsqu'un nœud quelconque a réussi à s'emparer du jeton de présidence. Le contrôleur nodal ainsi que les entités de protocole d'application de tous les nœuds sont informés par une primitive



d'indication GCC-Conductor-Assign lorsqu'une conférence passe en mode présidé. Cette indication mentionne également quel est le nœud qui est devenu le président.

La possibilité de fonctionnement d'une conférence en mode présidé est déterminée lors de la création de la conférence. Si l'indicateur "conférence pouvant être présidée" a été positionné, la conférence peut être placée en mode présidé. Dans le cas contraire, toute tentative de placer la conférence en mode présidé est rejetée.

Un nœud peut abandonner la présidence et faire revenir la conférence en mode non présidé par l'émission d'une primitive de demande GCC-Conductor-Release (abandon de la présidence). Le contrôleur nodal ainsi que les entités de protocole d'application de tous les nœuds sont informés lorsqu'une conférence bascule en mode non présidé par une primitive d'indication GCC-Conductor-Release.

Un nœud qui est à un instant donné le président d'une conférence peut transmettre directement la présidence à un autre nœud par l'émission d'une primitive de demande GCC-Conductor-Give (cession de présidence) qui spécifie le nœud récepteur souhaité, sans mettre la conférence en mode non présidé pendant la transition. Si le récepteur accepte la présidence, le contrôleur nodal ainsi que les entités de protocole d'application de tous les nœuds sont informés de la transition par une primitive d'indication GCC-Conductor-Assign qui mentionne également quel est le nœud qui est le nouveau président. Un nœud peut demander explicitement au président que la présidence lui soit transmise en lui envoyant une primitive de demande GCC-Conductor-Please (demande de présidence). Le président en exercice peut choisir de donner la présidence au nœud demandeur ou peut ignorer la demande.

Le contrôleur nodal et toutes les entités de protocole d'application d'un nœud quelconque peuvent également demander, par l'émission d'une primitive de demande GCC-Conductor-Inquire (interrogation de présidence), quel est le nœud, s'il existe, qui est actuellement en possession du jeton de présidence.

Quand une entité de protocole d'application reçoit l'information qu'une conférence à laquelle elle participe est en mode présidé, elle commencera immédiatement à agir selon le comportement en mode présidé prescrit par la spécification du protocole d'application correspondant. Quand une entité de protocole d'application est informée qu'une conférence à laquelle elle participe est en mode non présidé, elle commencera immédiatement à agir selon le comportement en mode non présidé prescrit par la spécification du protocole d'application correspondant. Une spécification normale de protocole d'application peut indiquer, par exemple, qu'une entité de protocole d'application doit demander une autorisation à l'entité de protocole d'application homologue avant d'exécuter toute action dans le mode présidé alors que cette autorisation n'est pas nécessaire dans le mode non présidé. Toutefois la question du comportement défini par les spécifications de protocole d'application en mode présidé ou non présidé se situe en dehors du domaine d'application de la présente Recommandation.

### **7.5.1 Description de services abstraits**

La liste qui suit récapitule les primitives définies dans le présent sous-paragraphe et en résume les fonctions:

- GCC-Conductor-Assign (*assignation de président*) – Utilisée par un contrôleur nodal situé dans un nœud conventionnel pour demander la présidence de la conférence. Lorsqu'un nœud devient le président, la primitive d'indication correspondante est utilisée pour annoncer ce fait à tous les contrôleurs nodaux de la conférence ainsi qu'à toutes les entités de protocole d'application de tous les nœuds de la conférence.
- GCC-Conductor-Release (*abandon de la présidence*) – Utilisée par un contrôleur nodal situé dans un nœud conventionnel pour abandonner la présidence de la conférence. Lorsqu'un nœud quelconque abandonne la présidence, la primitive d'indication correspondante est utilisée pour annoncer ce fait à tous les contrôleurs nodaux de la

conférence ainsi qu'à toutes les entités de protocole d'application de tous les nœuds de la conférence.

- GCC-Conductor-Please (*demande de présidence*) – Utilisée par un contrôleur nodal situé dans un nœud conventionnel pour demander au président en exercice de lui donner la présidence de la conférence.
- GCC-Conductor-Give (*cession de présidence*) – Utilisée par un contrôleur nodal situé dans un nœud conventionnel pour transmettre la présidence de la conférence à un nœud indiqué.
- GCC-Conductor-Inquire (*interrogation de présidence*) – Utilisée par un contrôleur nodal ou une entité de protocole d'application pour déterminer si la conférence est actuellement présidée et le cas échéant connaître l'identificateur de nœud du président en exercice.
- GCC-Conductor-Permission-Ask (*demande d'autorisation du président*) – Utilisée par un contrôleur nodal pour demander au président, pour les entités de protocole d'application de ce nœud, l'autorisation d'exécuter des actions exigeant l'accord du président lorsque la conférence est en mode présidé.
- GCC-Conductor-Permission-Grant (*octroi d'autorisation du président*) – Utilisée par un contrôleur nodal du nœud président pour indiquer quel est le sous-ensemble des nœuds d'une conférence qui ont reçu une autorisation de mode présidé.

#### 7.5.1.1 GCC-Conductor-Assign (assignation de président)

Dans le mode non présidé, le contrôleur nodal de tout nœud conventionnel peut émettre la primitive de demande GCC-Conductor-Assign afin de devenir président. Le paramètre résultat de la primitive de confirmation GCC-Conductor-Assign reçue indique si le demandeur est devenu le président ou non. Une primitive de demande GCC-Conductor-Assign qui a réussi sera suivie de l'émission d'indications GCC-Conductor-Assign destinées à tous les contrôleurs nodaux ainsi qu'à toutes les entités de protocole d'application de tous les nœuds de la conférence pour indiquer que la conférence est devenue présidée et donner l'information d'identificateur du président. La primitive d'indication correspondante peut également être envoyée à une entité de protocole d'application lors de son inscription dans la conférence, afin de l'informer que la conférence se trouve en mode présidé. L'ordre de succession des indications GCC-Conductor-Assign et GCC-Conductor-Release (abandon de présidence) représente l'ordre effectif des transitions de la présidence. Le Tableau 7-41 donne les paramètres et les types de primitives de ce service. La Figure 7-30 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

NOTE – Si le président se déconnecte de la conférence pour une cause quelconque, la conférence revient en mode non présidé jusqu'à ce qu'un autre nœud émette une demande pour devenir le président.

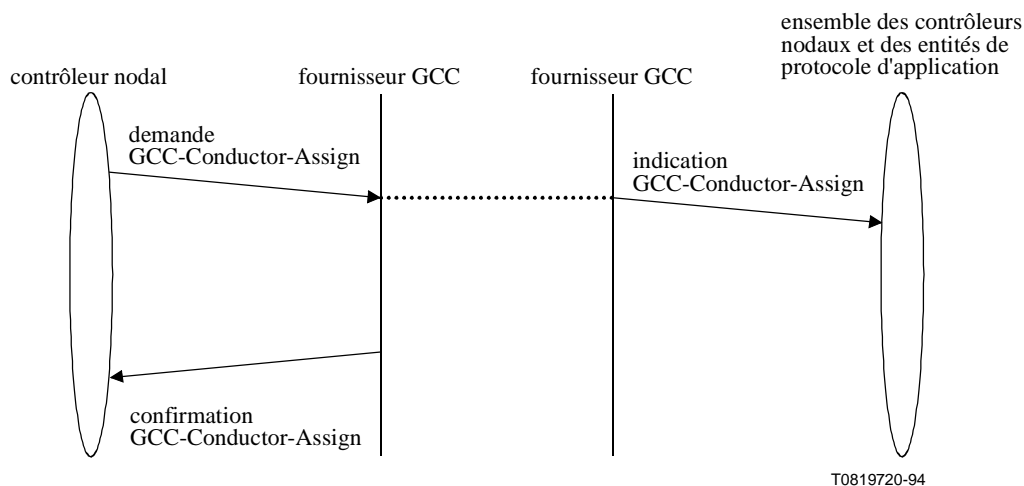
**Tableau 7-41 – Types de primitives et paramètres – GCC-Conductor-Assign**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Identificateur du nœud demandeur		M	
Résultat			M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Identificateur de nœud demandeur*: l'identificateur du nœud demandeur.

*Résultat*: indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "conférence ne pouvant pas être présidée" ou "jeton déjà attribué".



**Figure 7-30 – Séquence de primitives – GCC-Conductor-Assign**

### 7.5.1.2 GCC-Conductor-Release (abandon de présidence)

Le contrôleur nodal du nœud président émet la primitive de demande GCC-Conductor-Release afin de faire passer son statut de président à celui de participant ordinaire. Dès que le président a demandé l'abandon de la présidence, tous les contrôleurs nodaux de la conférence ainsi que toutes les entités de protocole d'application inscrites sur tous les nœuds de la conférence sont informés par la primitive d'indication GCC-Conductor-Release du changement de mode opératoire. L'ordre de succession des indications GCC-Conductor-Assign et GCC-Conductor-Release correspond à l'ordre effectif des transitions de la présidence. Un participant qui émet une primitive GCC-Conductor-Release sans être le président recevra du service GCC une primitive de confirmation GCC-Conductor-Release contenant un résultat négatif et une description de la cause. Dans ce cas, aucune indication ne sera envoyée aux autres participants. L'abandon de la présidence peut aussi être déclenchée par le service GCC, par exemple lorsque le président s'est déconnecté. La primitive d'indication correspondante peut également être envoyée à une entité de protocole d'application lors de son inscription afin de lui signaler que la conférence se trouve en mode non présidé. Le Tableau 7-42 donne les paramètres et les types de primitives de ce service. Les Figures 7-31 et 7-32 indiquent les séquences de primitives dans chacune des deux situations.

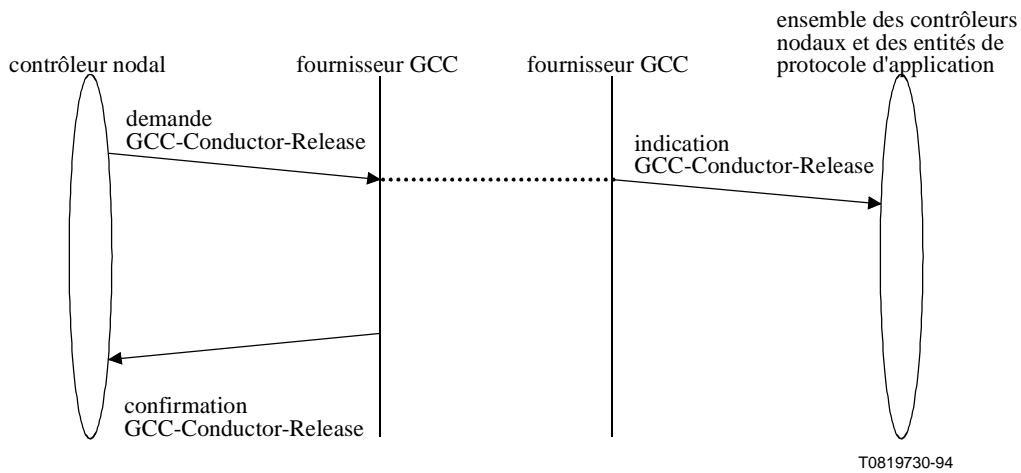
**Tableau 7-42 – Types de primitives et paramètres – GCC-Conductor-Release**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Résultat			M

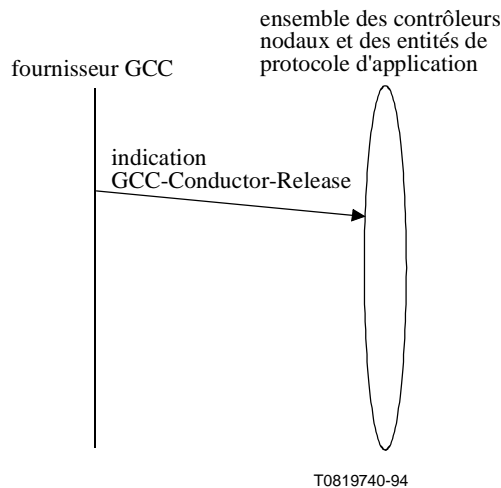
*Identificateur de conférence:* identificateur de la conférence à laquelle se réfère la primitive.

*Indicateur d'acceptation/rejet:* indicateur signalant si la primitive de demande GCC-Conductor-Release a été acceptée ou rejetée.

*Résultat:* indication si la demande a été acceptée ou rejetée avec la cause éventuelle du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "non propriétaire du jeton".



**Figure 7-31 – Séquence de primitives – GCC-Conductor-Release (lancée par l'utilisateur)**



**Figure 7-32 – Séquence de primitives – GCC-Conductor-Release (lancée par le service GCC)**

### 7.5.1.3 GCC-Conductor-Please (demande de présidence)

La primitive de demande GCC-Conductor-Please peut être émise par un contrôleur nodal situé dans un nœud conventionnel afin de demander au président en exercice de transférer la présidence au nœud demandeur. La primitive d'indication GCC-Conductor-Please est routée vers le nœud président en exercice. Le président en exercice peut alors décider de transférer la présidence au demandeur au moyen de la primitive GCC-Conductor-Give. La primitive de confirmation correspondante est uniquement une confirmation locale indiquant que la demande a été prise en charge par le fournisseur GCC local. Le Tableau 7-43 donne les paramètres et les types de primitives de ce service. La Figure 7-33 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

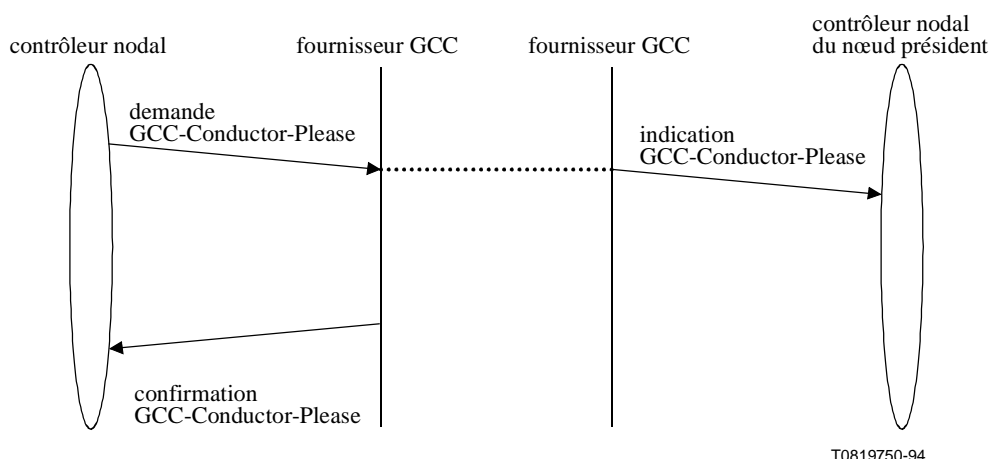
**Tableau 7-43 – Types de primitives et paramètres – GCC-Conductor-Please**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Identificateur du nœud demandeur		M	
Résultat			M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Identificateur du nœud demandeur*: identificateur du nœud demandeur.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "pas en mode présidé".



T0819750-94

**Figure 7-33 – Séquence de primitives – GCC-Conductor-Please**

NOTE – Cette primitive peut être employée pour garantir qu'un nœud donné devient le président de la conférence sans qu'un autre nœud ait la possibilité de prendre la présidence. La procédure consiste, pour l'organisateur de la conférence, à créer d'abord une conférence verrouillée puis à en prendre la présidence au moyen de la primitive GCC-Conductor-Assign (assignation de président). Une fois que l'organisateur est devenu le président de la conférence, il peut permettre l'entrée d'autres nœuds dans la conférence soit en déverrouillant celle-ci au moyen de la primitive GCC-Conference-Unlock (déverrouillage de conférence) et/ou en ajoutant directement les autres nœuds. Une fois que le nœud prévu comme président est entré dans la conférence, la présidence peut lui être transmise directement au moyen de la primitive GCC-Conductor-Give.

#### 7.5.1.4 GCC-Conductor-Give (cession de présidence)

La primitive de demande GCC-Conductor-Give peut être émise par un contrôleur nodal afin de transférer la présidence vers un nœud donné. Le président précédent conserve la présidence si celle-ci n'est pas acceptée par le récepteur envisagé. Le Tableau 7-44 donne les paramètres et les types de primitives de ce service. La Figure 7-34 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-44 – Types de primitives et paramètres – GCC-Conductor-Give**

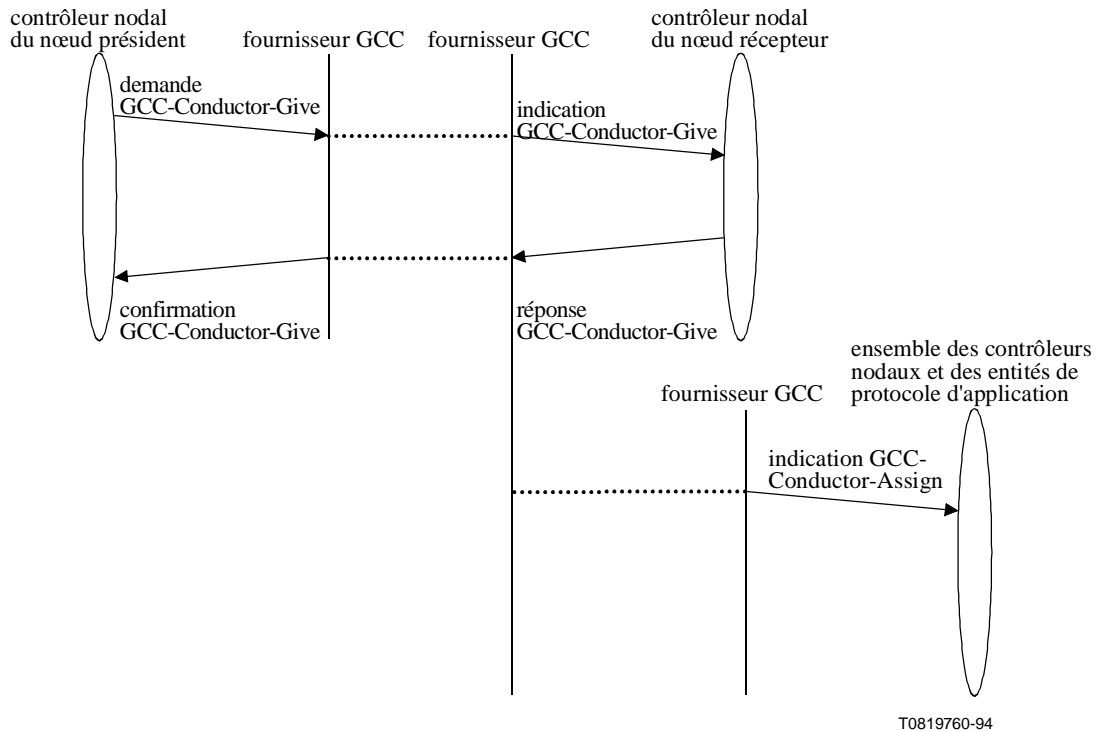
Paramètre	Demande	Indication	Réponse	Confirmation
Identificateur de conférence	M	M	M(=IN)	M(=RQ)
Identificateur du nœud récepteur	M			M(=)

Résultat			M	M(=)
----------	--	--	---	------

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Identificateur du nœud récepteur*: identificateur du nœud auquel sera transférée la présidence.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide", "non président" ou "cession non acceptée".



**Figure 7-34 – Séquence de primitives – GCC-Conductor-Give**

### 7.5.1.5 GCC-Conductor-Inquire (interrogation de présidence)

La primitive de demande GCC-Conductor-Inquire peut être émise à tout instant par un contrôleur nodal ou une entité de protocole d'application pour déterminer si la conférence est présidée, pour connaître l'identité du président éventuel ou pour déterminer si le nœud demandeur a reçu l'autorisation de mode présidé. Le Tableau 7-45 donne les paramètres et les types de primitives de ce service. La Figure 7-35 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-45 – Types de primitives et paramètres – GCC-Conductor-Inquire**

Paramètre	Demande	Confirmation
Identificateur de conférence	M	M(=)
Indicateur de mode présidé/non présidé		M
Identificateur du nœud président		C
Indicateur d'autorisation		C
Résultat		M

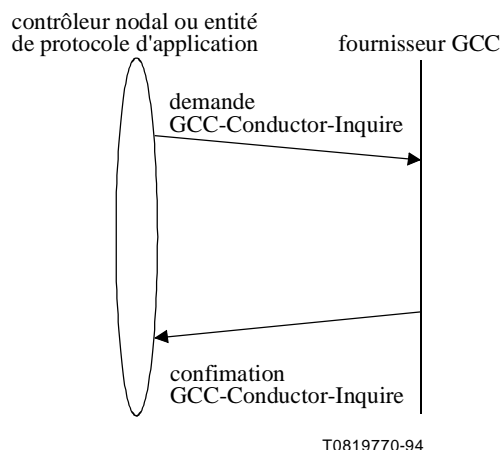
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Indicateur de mode présidé/non présidé*: indicateur signalant si la conférence est actuellement en mode présidé ou non.

*Identificateur du nœud président*: identificateur du nœud qui est actuellement le président. Omis si le mode en vigueur est non présidé.

*Indicateur d'autorisation*: si le mode est présidé, cet indicateur signale si le nœud local a reçu ou non l'autorisation de mode présidé.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



T0819770-94

**Figure 7-35 – Séquence de primitives – GCC-Conductor-Inquire**

### 7.5.1.6 GCC-Conductor-Permission-Ask (demande d'autorisation du président)

La primitive de demande GCC-Conductor-Permission-Ask peut être émise par un contrôleur nodal pour demander au président en exercice d'accorder ou de retirer aux entités de protocole d'application du nœud demandeur l'autorisation d'exécuter toute action nécessitant une autorisation du président. La définition actions spécifiques nécessitant une telle autorisation relève de la spécification du protocole d'application concerné. La primitive d'indication GCC-Conductor-Permission-Ask est routée vers le nœud président en exercice. L'ordre chronologique des indications GCC-Conductor-Permission-Ask provenant d'un nœud donné correspond à l'ordre effectif des demandes émanant de ce nœud. Le président en exercice peut choisir d'accorder ou de retirer au demandeur l'autorisation de mode présidé au moyen de la primitive GCC-Conductor-Permission-Grant (octroi d'autorisation du président). Il peut aussi choisir d'ignorer la demande. La primitive de confirmation correspondante est uniquement une confirmation locale indiquant que la demande a été prise en charge par le fournisseur GCC local. Aucune confirmation n'est donnée directement par le président. Le Tableau 7-46 donne les paramètres et les types de primitives de ce service. La Figure 7-36 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-46 – Types de primitives et paramètres – GCC-Conductor-Permission-Ask**

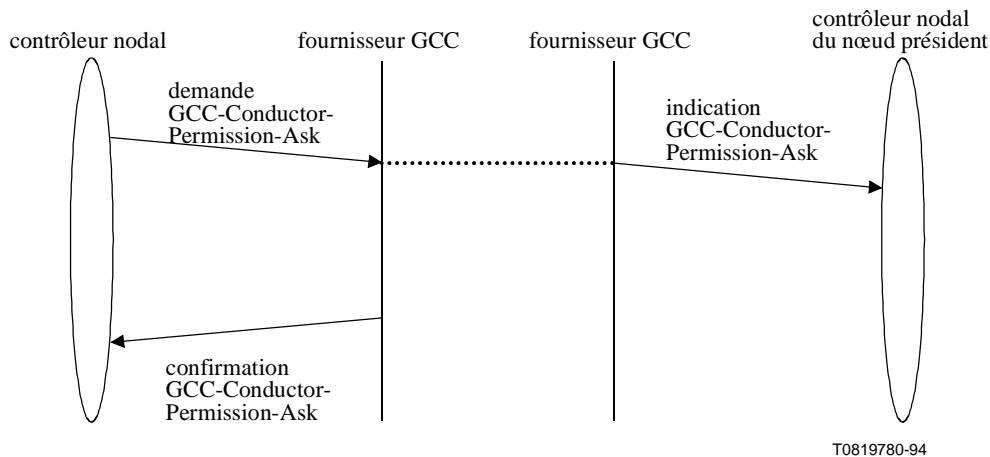
Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Indicateur d'accord/abandon	M	M(=)	M(=)
Identificateur du nœud demandeur		M	
Résultat			M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Indicateur d'accord/abandon*: indicateur signalant si le demandeur souhaite recevoir l'autorisation de mode présidé ou l'abandonner.

*Identificateur du nœud demandeur*: identificateur du nœud demandeur.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "pas en mode présidé".



**Figure 7-36 – Séquence de primitives – GCC-Conductor-Permission-Ask**

#### 7.5.1.7 GCC-Conductor-Permission-Grant (octroi d'autorisation du président)

La primitive de demande GCC-Conductor-Permission-Grant peut être émise par le contrôleur nodal du nœud président pour accorder ou retirer l'autorisation de mode présidé à un ou plusieurs nœuds d'une conférence. La primitive d'indication GCC-Conductor-Permission-Grant correspondante est diffusée à l'ensemble des nœuds de la conférence. Elle indique quels sont les nœuds qui possèdent actuellement l'autorisation de mode présidé et quels sont ceux qui en ont fait la demande et l'attendent encore. Cette dernière liste peut être fournie dans l'ordre selon lequel le président suppose que les autorisations vont finalement être accordées. Cette indication est fournie, pour chaque nœud, au contrôleur nodal ainsi qu'à toutes les entités de protocole d'application qui y sont inscrites. L'ordre chronologique des indications GCC-Conductor-Permission-Grant correspond à l'ordre effectif des demandes du président, c'est-à-dire que l'indication la plus récente s'applique. Une autorisation de mode présidé est généralement fournie en réponse à une primitive GCC-Conductor-Permission-Ask mais peut également être envoyée spontanément par le président. Le nœud présidé est supposé détenir l'autorisation de mode présidé figurant ou non d'une manière explicite dans la liste d'autorisations accordées aux nœuds. On peut supposer qu'aucun nœud ne possède d'autorisation lorsqu'une conférence passe pour la première fois en mode présidé, ceci étant vrai même si la conférence était antérieurement en mode présidé. Si la présidence change de main à la suite d'une opération GCC-Conductor-Give réussie, les autorisations restent dans l'état indiqué lors de la dernière diffusion par le président précédent. Lorsqu'un nouveau nœud entre dans une conférence, ce nœud supposera qu'il ne possède aucune autorisation. Le contrôleur nodal du nœud président peut rediffuser dans ce cas la liste des autorisations en émettant de nouveau une primitive de demande GCC-Conductor-Permission-Grant, de manière à informer le nouveau nœud sur l'état des autorisations des autres nœuds de la conférence. La primitive de confirmation correspondante est uniquement une confirmation locale indiquant que la demande a été prise en charge par le fournisseur GCC local. Le Tableau 7-47 donne les paramètres et les types de primitives de ce service. La Figure 7-37 indique la séquence d'événements correspondant à l'utilisation de cette primitive.



**Tableau 7-47 – Types de primitives et paramètres –  
GCC-Conductor-Permission-Grant**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Liste de nœuds ayant reçu l'autorisation	M	M(=)	
Liste de nœuds en attente d'autorisation	O	O(=)	
Indicateur d'autorisation		M	
Résultat			M

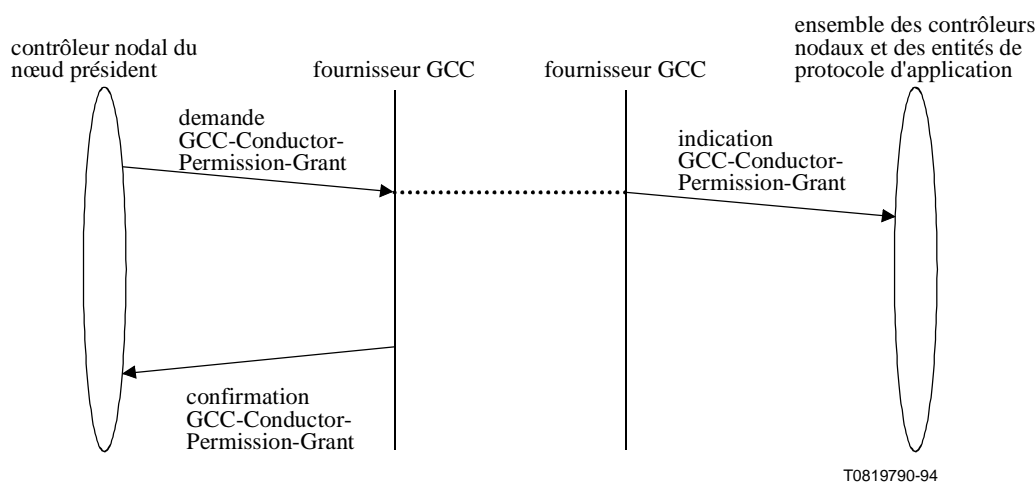
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Liste de nœuds ayant reçu l'autorisation*: liste d'identificateurs des nœuds, à raison d'un pour chaque nœud, auxquels le président a accordé l'autorisation de mode présidé. Cette liste est vide si aucun nœud n'a reçu l'autorisation de mode présidé.

*Liste de nœuds en attente d'autorisation*: liste ordonnée d'identificateurs des nœuds, à raison d'un pour chaque nœud, pour lesquels le président considère avoir accordé l'autorisation de mode présidé mais qui ne l'ont pas encore reçu. Le président peut trier cette liste dans l'ordre selon lequel il suppose que les autorisations vont être accordées. Dans ce cas, la première entrée de la liste est considérée comme le prochain nœud devant probablement recevoir une autorisation.

*Indicateur d'autorisation*: cet indicateur signale si le nœud local figure dans la liste des nœuds pour lesquels une autorisation a été accordée. Son utilité principale est de permettre aux entités de protocole d'application de déterminer leur mode opératoire sans avoir à faire une recherche dans les listes renvoyées.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "non président".



**Figure 7-37 – Séquence de primitives – GCC-Conductor-Permission-Grant**

## 7.6 Fonctions diverses

### 7.6.1 Description de services abstraits

La liste qui suit récapitule les primitives définies dans le présent sous-paragraphe et en résume les fonctions:

- GCC-Conference-Time-Remaining (*temps restant d'une conférence*) – Permet au contrôleur nodal de l'organisateur de conférence de notifier à tous les nœuds qu'une conférence limitée dans le temps est prévue pour se terminer à un instant donné.
- GCC-Conference-Time-Inquire (*information de temps sur une conférence*) – Permet au contrôleur nodal d'un nœud quelconque de déterminer combien de temps reste à courir dans une conférence limitée dans le temps.
- GCC-Conference-Extend (*extension du temps d'une conférence*) – Permet au contrôleur nodal d'un nœud quelconque de demander à l'organisateur d'étendre la conférence limitée dans le temps au-delà de la durée allouée.
- GCC-Conference-Assistance (*assistance de conférence*) – Utilisée pour demander à un opérateur de conférence une forme d'assistance non spécifiée.
- GCC-Text-Message (*message de texte*) – Utilisée pour envoyer un message de texte quelconque à un ensemble d'autres nœuds afin de le présenter à l'utilisateur ou aux utilisateurs de ces nœuds.

#### 7.6.1.1 GCC-Conference-Time-Remaining (temps restant d'une conférence)

La primitive de demande GCC-Conference-Time-Remaining peut être utilisée par un contrôleur nodal pour annoncer à tous les nœuds le temps qui reste à courir dans une conférence limitée dans le temps. Elle peut être envoyée à un nœud donné sans être diffusée aux autres nœuds. Cette demande est prévue pour être émise par l'organisateur, mais peut aussi être envoyée par d'autres nœuds. Généralement cette primitive est envoyée une seule fois, vers la fin de la conférence, pour indiquer que la conférence est proche de son terme. Le Tableau 7-48 donne les paramètres et les types de primitives de ce service. La Figure 7-38 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-48 – Types de primitives et paramètres –  
GCC-Conference-Time-Remaining**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Temps restant	M	M(=)	
Identificateur de nœud	O	O(=)	
Identificateur du nœud origine		M	
Résultat			M

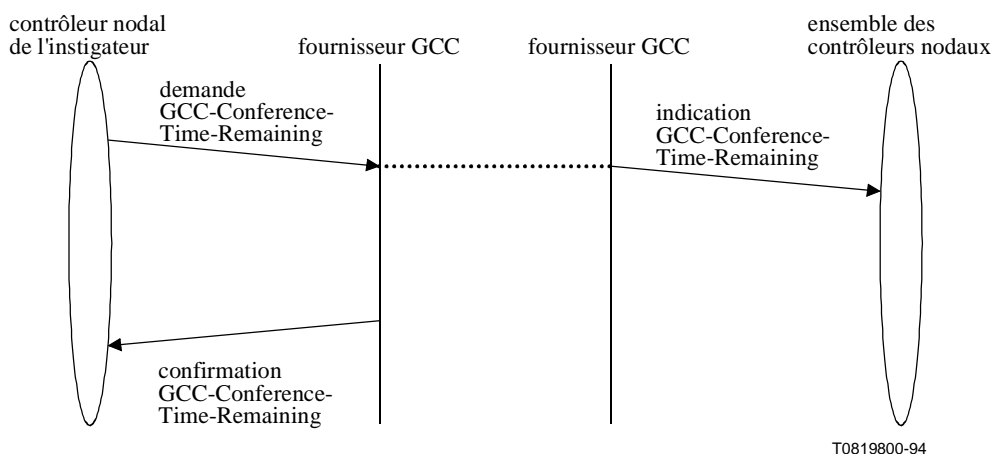
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Temps restant*: indication du temps restant de la conférence en nombre entier de secondes.

*Identificateur de nœud*: paramètre optionnel indiquant que l'information de temps restant ne s'applique qu'au nœud indiqué. S'il n'est pas fourni, le temps restant s'applique à tous les nœuds de la conférence.

*Identificateur du nœud origine*: identificateur du nœud qui a émis la demande.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



T0819800-94

**Figure 7-38 – Séquence de primitives – GCC-Conference-Time-Remaining**

**7.6.1.2 GCC-Conference-Time-Inquire (information de temps sur une conférence)**

La primitive de demande GCC-Conference-Time-Inquire peut être utilisée par un nœud quelconque pour demander à l'organisateur le temps restant à courir dans la conférence. La réception de la primitive d'indication correspondante par l'organisateur a pour résultat, si celui-ci prend en charge cette primitive, la diffusion de la valeur du temps restant à tous les nœuds de la conférence au moyen de la primitive GCC-Conference-Time-Remaining (temps restant d'une conférence). Le Tableau 7-49 donne les paramètres et les types de primitives de ce service. La Figure 7-39 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-49 – Types de primitives et paramètres – GCC-Conference-Time-Inquire**

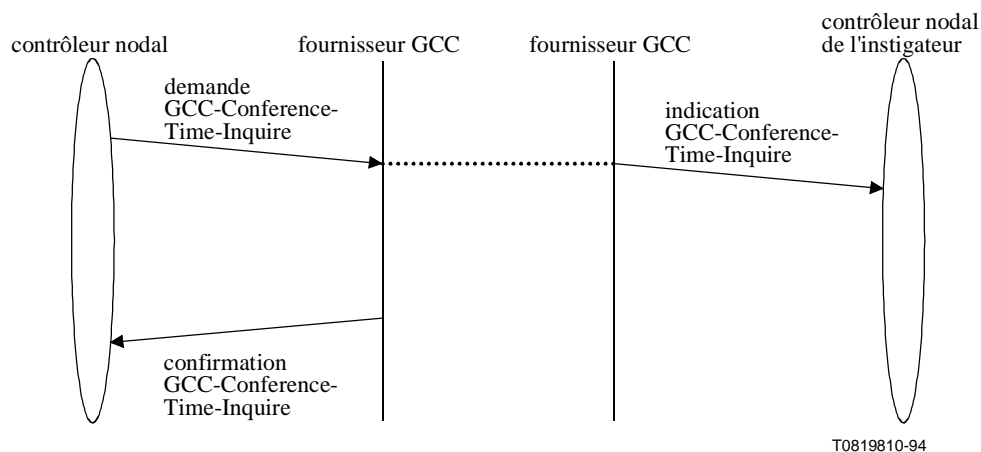
Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Indicateur de temps spécifique conférence ou nœud	M	M(=)	
Identificateur du nœud demandeur		M	
Résultat			M

*Identificateur de conférence:* identificateur de la conférence à laquelle se réfère la primitive.

*Indicateur de temps spécifique conférence ou nœud:* indicateur signalant que la demande est faite pour trouver le temps restant pour la conférence dans sa totalité ou, s'il est différent, le temps restant pour le nœud demandeur. Ce paramètre peut être ignoré si l'organisateur ne prend en charge qu'un instant unique pour la fin de la conférence.

*Identificateur du nœud demandeur:* identificateur d'utilisateur du nœud demandeur.

*Résultat:* indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



T0819810-94

**Figure 7-39 – Séquence de primitives – GCC-Conference-Time-Inquire**

### 7.6.1.3 GCC-Conference-Extend (extension de temps d'une conférence)

La primitive de demande GCC-Conference-Extend peut être utilisée par le contrôleur nodal d'un nœud pour demander à l'organisateur d'ajouter un temps supplémentaire à une conférence limitée dans le temps. La réception de la primitive d'indication correspondante par l'organisateur a pour résultat, si celui-ci prend en charge cette primitive, la diffusion de la nouvelle valeur du temps restant à tous les nœuds de la conférence au moyen de la primitive GCC-Conference-Time-Remaining (temps restant d'une conférence), même s'il n'y a pas eu effectivement d'extension du temps. Il n'est pas exigé que le temps effectivement ajouté par l'organisateur soit égal au temps demandé. Le Tableau 7-50 donne les paramètres et les types de primitives de ce service. La Figure 7-40 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-50 – Types de primitives et paramètres – GCC-Conference-Extend**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Temps demandé	M	M(=)	M(=)
Indicateur de temps spécifique conférence ou nœud	M	M(=)	
Identificateur du nœud demandeur		M	
Résultat			M

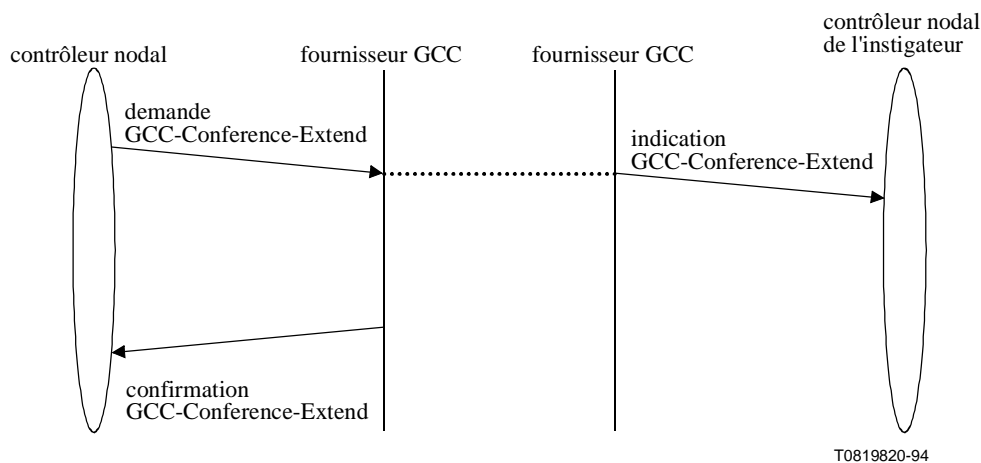
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Temps demandé*: paramètre indiquant le temps, en nombre entier de secondes, devant être ajouté pour étendre la conférence.

*Indicateur de temps spécifique conférence ou nœud*: indicateur signalant si la demande est faite pour le temps restant de la totalité de la conférence ou uniquement pour le nœud demandeur. Ce paramètre peut être ignoré si l'organisateur ne prend en charge qu'un instant unique pour la fin de la conférence.

*Identificateur du nœud demandeur*: identificateur du nœud demandeur.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès", "conférence non valide" ou "non président".



**Figure 7-40 – Séquence de primitives – GCC-Conference-Extend**

#### 7.6.1.4 GCC-Conference-Assistance (assistance de conférence)

La primitive de demande GCC-Conference-Assistance fournit un moyen simple pour demander à un opérateur de conférence de fournir une certaine forme d'assistance. L'envoi d'une demande GCC-Conference-Assistance a pour résultat la diffusion d'une primitive d'indication GCC-Conference-Assistance au contrôleur nodal de tous les nœuds d'une conférence donnée qui prennent en charge cette primitive. La réponse prévue pour cette primitive n'est pas spécifiée et se trouve en dehors du domaine d'application de la présente Recommandation. Le Tableau 7-51 donne les paramètres et les types de primitives de ce service. La Figure 7-41 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-51 – Types de primitives et paramètres – GCC-Conference-Assistance**

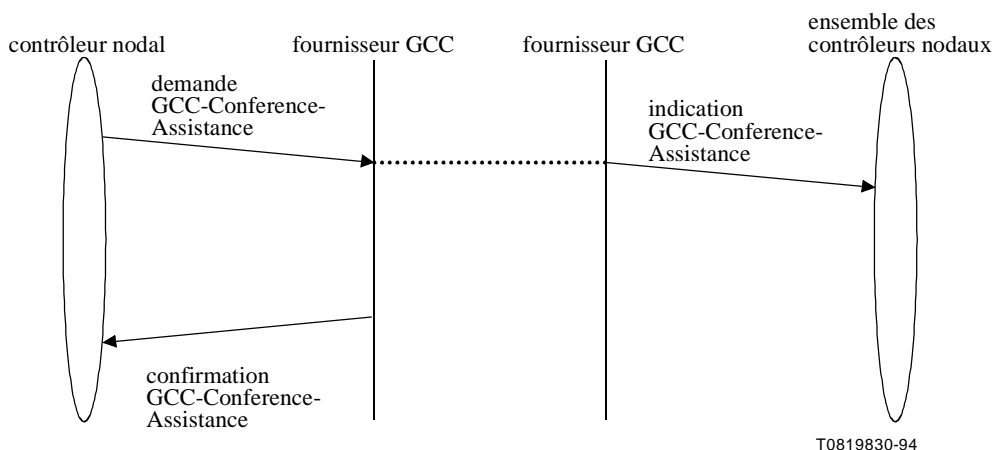
Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Données utilisateur	O	O(=)	
Identificateur du nœud origine		M	
Résultat			M

*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Données utilisateur*: données utilisateur non spécifiées.

*Identificateur du nœud origine*: identificateur de nœud de l'origine de la demande d'assistance.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



**Figure 7-41 – Séquence de primitives – GCC-Conference-Assistance**

### 7.6.1.5 GCC-Text-Message (message de texte)

La primitive de demande GCC-Text-Message fournit un moyen simple pour communiquer des messages de texte non spécifiés. L'envoi d'une primitive de demande GCC-Text-Message a pour résultat la diffusion d'une primitive d'indication GCC-Text-Message au contrôleur nodal de tous les nœuds d'une conférence donnée qui prennent en charge cette primitive ou l'envoi de l'indication à un nœud unique. L'action prévue pour cette primitive est la présentation du message de texte aux participants de la conférence par un moyen non spécifié par la présente Recommandation. Le Tableau 7-52 donne les paramètres et les types de primitives de ce service. La Figure 7-42 indique la séquence d'événements correspondant à l'utilisation de cette primitive.

**Tableau 7-52 – Types de primitives et paramètres – GCC-Text-Message**

Paramètre	Demande	Indication	Confirmation
Identificateur de conférence	M	M	M(=RQ)
Message de texte	M	M(=)	
Identificateur du nœud destinataire	O		
Identificateur du nœud origine		M	
Résultat			M

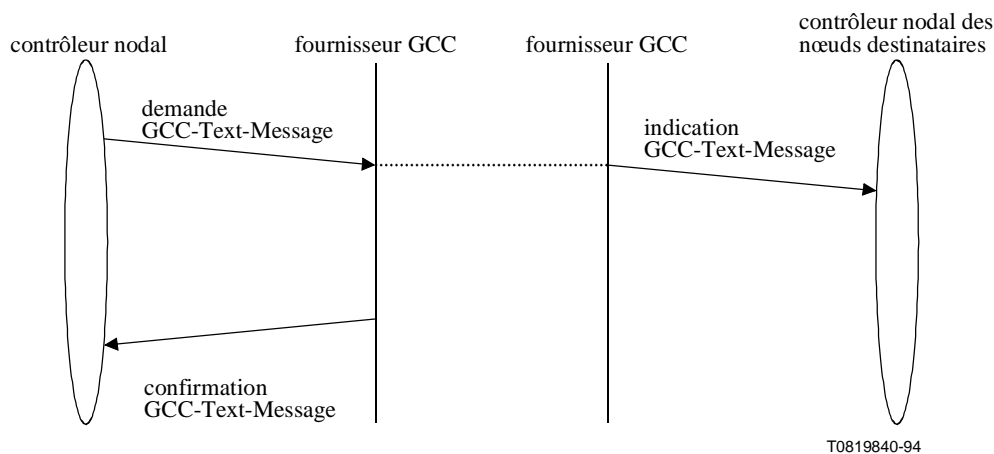
*Identificateur de conférence*: identificateur de la conférence à laquelle se réfère la primitive.

*Message de texte*: texte de message unicode.

*Identificateur du nœud destinataire*: identificateur de nœud d'un nœud unique destiné à recevoir le message. Si aucun nœud n'est indiqué, le message est diffusé à tous les nœuds de la conférence.

*Identificateur du nœud origine*: identificateur de nœud de la source du message de texte.

*Résultat*: indication si la demande a été acceptée ou rejetée avec le motif éventuel du rejet. Elle contient un des résultats possibles suivants: "succès" ou "conférence non valide".



T0819840-94

**Figure 7-42 – Séquence de primitives – GCC-Text-Message**

## 8 Spécification du protocole GCC

### 8.1 Fonctionnement général

Au niveau de chaque nœud, le fournisseur GCC, qui constitue l'application de commande du service MCS, communique avec ce service MCS à travers le point de contrôle MCSAP. Au moment de l'initialisation, le fournisseur GCC établira par un moyen local, sa liaison de communication avec le service MCS à travers un point de contrôle MCSAP.

Une conférence est toujours établie par la création d'un domaine MCS. Il existe une correspondance biunivoque entre les conférences et les domaines MCS. La création effective d'un domaine MCS se fait par un moyen local. La création des conférences et l'entrée dans des conférences se font par les primitives MCS-Connect-Provider (connexion de fournisseur).

Les primitives de demande qui se réfèrent à une conférence utilisent un identificateur de conférence. Le fournisseur GCC local destinataire d'une telle demande examinera si le nœud fait partie de la conférence mentionnée. Si tel est le cas, le fournisseur GCC traite la primitive comme décrit ci-dessous. Dans le cas contraire, la demande est rejetée et la primitive de confirmation correspondante est émise en indiquant "conférence non valide" comme cause du rejet. Toute primitive de réponse émise qui se réfère à une conférence non valide est ignorée par le fournisseur GCC (on l'erreur peut être traitée par un moyen local non spécifié).

La communication entre fournisseurs GCC se fait au moyen d'unités de données de protocole GCC (GCCPDU, *GCC protocol data unit*). Les GCCPDU sont transmises soit au moyen de l'une des deux primitives MCS de service de données MCS-Send-Data (envoi de données) ou MCS-Uniform-Send-Data (envoi uniforme de données) comme indiqué dans le Tableau 9-4, soit au moyen de primitives MCS-Connect-Provider (connexion de fournisseur) pour les GCCPDU utilisées pendant l'établissement de la connexion.

Toutes les GCCPDU appartiennent à l'une des trois catégories demande, réponse ou indication. Les PDU de demande sont celles qui exigent en retour une PDU de réponse. Les PDU d'indication sont celles qui n'exigent pas de réponse ou, dans certains cas, celles pour lesquelles la réponse est fournie d'une manière indirecte. Il existe, pour les PDU de demande qui ne correspondent pas à une fonction obligatoire, une PDU de réponse générique, *FunctionNotSupportedResponse* (réponse de fonction non prise en charge). Ceci permet à un fournisseur GCC de répondre au nœud demandeur sans avoir à connaître le format spécifique de la PDU de réponse. La PDU *FunctionNotSupportedResponse* contient intégralement la PDU de demande afin de permettre au demandeur de déterminer à quoi correspond une telle réponse. La PDU

FunctionNotSupportedResponse sera envoyée avec le même niveau de priorité que celui de la PDU de demande reçue.

NOTE – Il n'y a pas de correspondance stricte entre la terminologie utilisée (demande, réponse et indication) et les termes "request", "response" et "indication" utilisés dans la dénomination des primitives. La terminologie a été choisie de manière à être en rapport avec le but primaire des types de PDU, compte tenu des fonctions qu'elles exécuteront. C'est pourquoi la terminologie "confirmation" n'est pas nécessaire pour décrire les PDU: les PDU de réponse sont celles qui sont issues d'une primitive de réponse et induisent comme résultat une primitive de confirmation.

Le protocole du service GCC inclut la prise en charge d'extensions non normalisées. Un fournisseur GCC qui ne connaît pas une PDU nonStandardRequest (demande non normalisée) enverra en retour une PDU FunctionNotSupportedResponse (réponse de fonction non prise en charge). Un fournisseur GCC peut ignorer des PDU nonStandardResponse (réponse non normalisée) et nonStandardIndication (indication non normalisée).

## 8.2 Etablissement et fin de conférence

### 8.2.1 Création de conférence

Un fournisseur GCC qui reçoit la primitive de demande GCC-Conference-Create (création d'une conférence) émettra une primitive de demande MCS-Connect-Provider avec les paramètres donnés par le Tableau 8-1. Le fournisseur GCC local attribuera l'identificateur de conférence qui sera utilisé localement comme sélecteur de domaine MCS. L'identificateur de conférence est présent comme paramètre "sélecteur de domaine appelant". Il est également géré par le fournisseur GCC local comme un moyen d'identifier cette conférence dans des primitives ultérieures.

Si la combinaison des paramètres "nom de conférence" et "modificateur de nom de conférence" (utilisant soit la forme numérique, soit la forme textuelle du nom) est identique à celle d'une conférence dont fait déjà partie le nœud local, la demande est au contraire rejetée immédiatement par l'émission d'une primitive de réponse GCC-Conference-Create donnant comme résultat "le nom de conférence existe déjà". Dans le cas contraire, le fournisseur GCC mémorisera le nom de conférence et le modificateur de nom de conférence pour une utilisation ultérieure dans des procédures de réponse à une interrogation de la conférence, d'entrée dans la conférence ou pour lancer une invitation à la conférence. Le nom de conférence est également utilisé dans la PDU ConferenceCreateRequest (demande de création de conférence).

**Tableau 8-1 – Paramètres de la primitive MCS-Connect-Provider pour la PDU ConferenceCreateRequest**

Paramètre	Contenu
Adresse appelante	De la primitive de demande
Sélecteur de domaine appelant	Identificateur de conférence choisi par le fournisseur GCC
Adresse appelée	De la primitive de demande
Sélecteur de domaine appelé	NULL (néant)
Indicateur de direction	Montée
Paramètres de domaine	De la primitive de demande
Qualité de service	De la primitive de demande
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceCreateRequest (voir le Tableau 8-2)

Le paramètre "données utilisateur" de la primitive de demande MCS-Connect-Provider contient une structure constituée d'un identificateur d'objet identifiant – conformément à la présente



Recommandation – la PDU qui fait partie de la structure, suivie de cette PDU elle-même. Cette structure est définie en détail au § 9.6. Le contenu de la PDU est donné par le Tableau 8-2.

**Tableau 8-2 – GCCPDU ConferenceCreateRequest**

Contenu	Origine	Destination
Nom de conférence	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Mot de passe de l'organisateur (optionnel)	Primitive de demande	Primitive d'indication
Mot de passe (optionnel)	Primitive de demande	Primitive d'indication
Indicateur de conférence verrouillée	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Indicateur de conférence listée	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Indicateur de conférence pouvant être présidée	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Méthode de fin	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Liste de privilèges du président (optionnel)	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Liste de privilèges de conférence en mode présidé (optionnel)	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Liste de privilèges de conférence en mode non présidé (optionnel)	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Description de conférence (optionnel)	Primitive de demande	Primitive d'indication et fournisseur GCC destinataire
Identificateur de l'appelant (optionnel)	Primitive de demande	Primitive d'indication
Mode de conférence (optionnel)	Primitive de demande	Primitive d'indication
Données utilisateur (optionnel)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit la primitive d'indication MCS-Connect-Provider contenant un identificateur d'objet T.124 et l'unité PDU ConferenceCreateRequest, un fournisseur GCC produira une primitive d'indication GCC-Conference-Create avec les paramètres tels qu'ils sont donnés dans l'unité PDU ConferenceCreateRequest. Il enverra cette primitive au point de contrôle GCCSAP. Le fournisseur GCC attribue un identificateur de conférence – une chaîne de caractères localement non ambiguë – qui figurera également dans la primitive. Si la création de conférence réussit, le fournisseur GCC utilisera cet identificateur comme un moyen d'identifier par la suite la conférence dans des primitives. Si le service GCC ne dispose pas de ressources suffisantes pour créer une nouvelle conférence, il peut produire automatiquement la réponse négative sans produire la primitive d'indication GCC-Conference-Create. Sinon, lorsqu'il reçoit du point de contrôle GCCSAP une primitive de réponse GCC-Conference-Create positive, le fournisseur GCC, qui est devenu le fournisseur GCC sommital pour cette conférence, enverra une primitive de demande MCS-Attach-User (rattachement d'utilisateur). Lorsqu'il reçoit la confirmation MCS-Attach-User qui contient l'identificateur de nœud attribué, le fournisseur GCC se branchera sur le canal utilisateur GCC par l'émission d'une primitive de demande MCS-Channel-Join (branchement sur canal). Le fournisseur GCC se branchera également sur le ou les canaux de diffusion GCC appropriés par l'émission d'une ou deux primitives de demande MCS-Channel-Join. De telles demandes peuvent être émises en succession sans attendre la confirmation de la demande précédente. Il peut être nécessaire d'indiquer localement au fournisseur MCS qu'un nouveau

domaine vient d'être créé avant d'effectuer les branchements. Tout échange nécessaire à cet effet est considéré comme un sujet local non couvert par la présente Recommandation.

Il existe trois sortes de canaux de diffusion GCC, diffusant chacun des ensembles informationnels similaires mais différents. Les versions antérieures du protocole de commande GCC utilisaient un canal de diffusion unique appelé "GCC-Broadcast-Channel" (canal de diffusion GCC). Sur ce canal, les nœuds conventionnels plus anciens, qui existaient avant l'introduction des catégories nodales, recevaient des messages de commande GCC et des mises à jour complètes de répertoire. Pour prendre en charge tous les nœuds GCC pouvant participer à une conférence, il est impératif que les nœuds prennent en charge le canal de diffusion GCC original, bien qu'aucun nœud ne puisse y être joint (à l'exception peut-être des nœuds d'extrémité, qui sont anonymes). Un deuxième canal de diffusion GCC, appelé "GCC-Conventional-Broadcast-Channel" (canal de diffusion conventionnel GCC), est utilisé par les nœuds qui n'interprètent pas les catégories nodales. Toutes les unités PDU de commande GCC et toutes les mises à jour delta de répertoire qui sont produites lorsque des nœuds conventionnels joignent, quittent ou modifient un enregistrement de répertoire sont diffusées par ce canal. Tous les nœuds (sauf ceux des anciens protocoles) doivent entrer dans ce canal de diffusion. Les nœuds anonymes peuvent ignorer les mises à jour delta de répertoire reçues sur ce canal mais doivent traiter les messages de commande reçus. Le dernier canal de diffusion GCC est appelé "GCC-Counted-Broadcast-Channel" (canal de diffusion compté GCC). Sur ce canal, le fournisseur sommital diffuse des mises à jour delta de répertoire qui sont produites lorsque des nœuds comptés joignent, quittent, ou modifient un enregistrement de répertoire. Seuls les nœuds conventionnels entrent dans ce canal de diffusion GCC, ce qui réduit l'intensité du trafic dans le réseau lorsqu'un grand nombre de nœuds comptés sont en train de participer à la conférence.

Si la primitive de réponse GCC-Conference-Create contient un modificateur de nom de conférence, le fournisseur GCC, devenu fournisseur GCC sommital, mémorisera ce modificateur pour une utilisation ultérieure dans des procédures de réponse à une interrogation de la conférence, d'entrée dans la conférence ou d'invitation à une conférence.

Le fournisseur GCC générera une primitive de réponse MCS-Connect-Provider (connexion de fournisseur) comportant un résultat qui indique le succès ou "utilisateur rejeté" selon que la primitive de réponse GCC-Conference-Create a indiqué un succès ou un échec. Le paramètre "données utilisateur" contient un identificateur d'objet T.124 ainsi que la PDU ConferenceCreateResponse (réponse de création de conférence). Le contenu de la primitive MCS-Connect-Provider est donné par le Tableau 8-3. Si la création d'une conférence réussit, le fournisseur GCC du nœud recevant la primitive d'indication MCS-Connect-Provider deviendra le fournisseur GCC sommital pour la conférence.

**Tableau 8-3 – Paramètres de réponse de la primitive MCS-Connect-Provider pour la PDU ConferenceCreateResponse**

Paramètre	Contenu
Paramètres de domaine	De la primitive de réponse
Qualité de service	De la primitive de réponse
Résultat	Comme spécifié dans [UIT-T T.122]
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceCreateResponse (voir le Tableau 8-4)

Le contenu de la PDU ConferenceCreateResponse (réponse de création de conférence) est donné par le Tableau 8-4. Le fournisseur GCC qui est à l'origine de la PDU fournira le paramètre "identificateur de nœud". Cet identificateur d'utilisateur a été attribué par le service MCS en réponse à la primitive de demande MCS-Attach-User envoyée par le fournisseur GCC. Le paramètre

"marque", attribué par le fournisseur GCC d'origine d'une manière localement non ambiguë, est utilisé pour identifier la PDU UserIDIndication (indication d'identificateur utilisateur) envoyée en retour. Le paramètre résultat contient une information spécifique d'erreur issue directement du paramètre résultat de la primitive de réponse GCC-Conference-Create. Si ce dernier n'indique pas le succès, le paramètre résultat de la primitive de réponse MCS-Connect-Provider est positionné sur "utilisateur rejeté".

**Tableau 8-4 – GCCPDU ConferenceCreateResponse**

Contenu	Origine	Destination
Identificateur de nœud	Fournisseur GCC sommital	Fournisseur GCC destinataire
Marque	Fournisseur GCC sommital	Fournisseur GCC destinataire
Résultat	Primitive de réponse	Primitive de confirmation
Données utilisateur (optionnel)	Primitive de réponse	Primitive de confirmation

Lorsqu'il reçoit l'unité PDU ConferenceCreateResponse indiquant un résultat positif, le fournisseur GCC émettra en premier lieu une primitive de demande MCS-Attach-User. Lorsqu'il reçoit la confirmation MCS-Attach-User contenant l'identificateur de nœud attribué, le fournisseur GCC se branchera sur le canal de l'identificateur de nœud en émettant une demande MCS-Channel-Join. Le fournisseur GCC doit également se brancher sur le canal GCC-Conventional-Broadcast-Channel ainsi que sur le canal GCC-Counted-Broadcast-Channel en émettant deux demandes MCS-Channel-Join (les nœuds d'organisateur sont toujours considérés comme étant conventionnels, voir ci-dessus l'explication relative aux canaux de diffusion). Le fournisseur GCC peut également se brancher sur le canal GCC-Convener-Channel (canal d'organisateur) s'il utilise l'une quelconque des fonctions qui exigent la présence de ce canal. Les primitives de demande MCS-Channel-Join (branchement sur un canal) peuvent être émises à la suite sans attendre la confirmation de la précédente. Une fois que le fournisseur GCC a reçu une confirmation MCS-Channel-Join pour chacune des demandes de branchement de canal, il enverra une unité PDU UserIDIndication au fournisseur GCC sommital. Pour ce faire, il enverra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de nœud du fournisseur GCC sommital, une priorité de données maximale et l'unité PDU dans le champ de données. Le contenu de l'unité PDU UserIDIndication est donné par le Tableau 8-5. Le paramètre "marque" reçoit la valeur du paramètre correspondant reçu dans l'unité PDU ConferenceCreateResponse.

**Tableau 8-5 – GCCPDU UserIDIndication**

Contenu	Origine	Destination
Marque	Fournisseur GCC origine	Fournisseur GCC destinataire

Le fournisseur GCC générera ensuite une primitive de confirmation GCC-Conference-Create qu'il enverra au point de contrôle GCCSAP. Cette primitive contiendra les paramètres "nom de conférence" de la primitive de demande d'origine, "modificateur de nom de conférence" (s'il existe) et "résultat" de la PDU reçue ainsi que l'identificateur de conférence attribué localement. Si la PDU reçue a indiqué un échec ou si le fournisseur GCC reçoit une primitive d'indication MCS-Disconnect-Provider (déconnexion de fournisseur) pour cette connexion avant d'avoir envoyé une confirmation GCC-Conference-Create positive, la primitive de confirmation GCC-Conference-Create est émise immédiatement avec une indication de résultat négatif, les demandes de connexion d'utilisateur et de canal ne sont pas émises et la PDU UserIDIndication (indication d'identificateur d'utilisateur) n'est pas envoyée. Le paramètre résultat de la PDU ainsi que le paramètre résultat indiqué par la primitive de confirmation MCS-Connect-Provider, ou le paramètre "cause" de la primitive d'indication MCS-Disconnect-Provider, sont utilisés pour générer

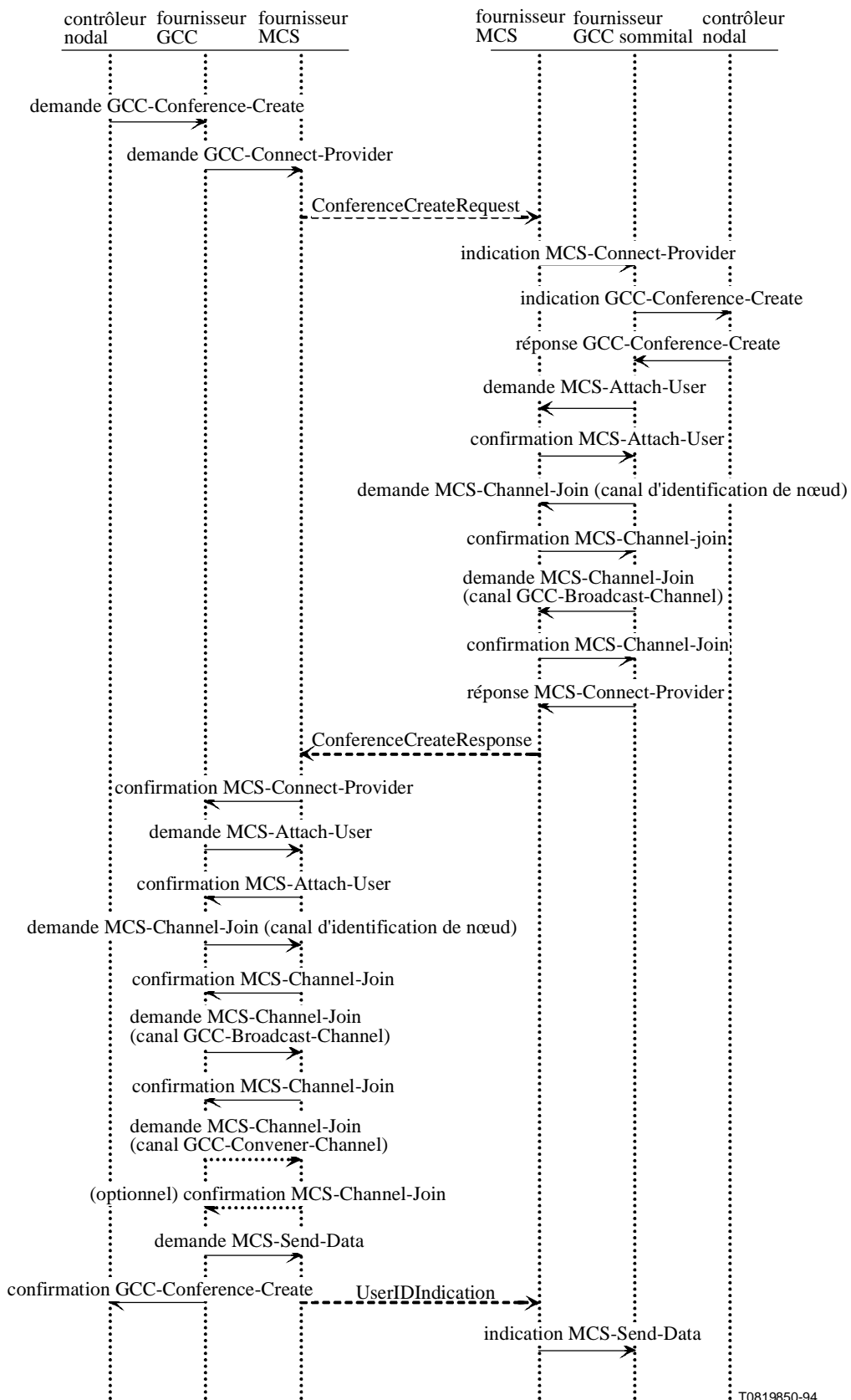
le résultat renvoyé dans la primitive de confirmation GCC-Conference-Create. Si le paramètre résultat de la confirmation MCS-Connect-Provider est "utilisateur rejeté", le paramètre résultat de la PDU est utilisé pour déterminer le résultat renvoyé. Sinon, le paramètre résultat de la primitive d'indication MCS-Connect-Provider est utilisé directement.

Lorsqu'il reçoit l'indication UserIDIndication, le nœud sur lequel la conférence a été créée comparera le paramètre "marque" avec sa liste de marques en attente. Si une concordance est trouvée, le nœud mémorise l'identificateur d'utilisateur du nœud origine (extrait de la primitive d'indication MCS-Send-Data) dans sa base de données contenant les identificateurs de nœud des nœuds qui lui sont directement subordonnés. La PDU sera ignorée si la marque ne coïncide avec aucune marque en attente.

Si le fournisseur GCC sommital qui vient d'être créé pour cette conférence reçoit une primitive d'indication MCS-Disconnect-Provider pour la connexion en cours d'établissement à un moment quelconque de la création de la conférence (allant de l'instant d'émission de la primitive d'indication GCC-Conference-Create à l'instant de réception de l'indication UserIDIndication), il enverra au point de contrôle GCCSAP une primitive d'indication GCC-Conference-Terminate (fin de conférence) donnant comme cause une demande de fin normale si la cause fournie par la primitive d'indication MCS-Disconnect-Provider est "demandé par l'utilisateur" et une fin anormale dans les autres cas. Il cessera ensuite tout traitement de la connexion. Le fournisseur GCC cessera tout traitement d'établissement de cette connexion et n'exécutera aucune autre action si la primitive d'indication MCS-Disconnect-Provider est reçue avant l'envoi de la primitive d'indication GCC-Conference-Create au point de contrôle GCCSAP.

La Figure 8-1 présente la succession des événements dans le cas d'une création de conférence réussie.

NOTE – Si une conférence est créée sur initiative locale plutôt que par une demande d'un autre nœud, le fournisseur GCC du nœud local prendra en charge à la place du fournisseur GCC sommital la succession d'événements indiquée par la Figure 8-1 à partir de la primitive de demande MCS-Attach-User. Cela signifie qu'il s'attachera au service MCS en envoyant une primitive de demande MCS-Attach-User et en attendra la confirmation, puis se branchera sur son canal de l'identificateur de nœud et sur les canaux de diffusion GCC appropriés. Il peut, facultativement, se brancher sur le canal GCC-Convener-Channel (canal d'organisateur) car le nœud local est également l'organisateur dans le cas présent. Dans ce cas, si un autre nœud entre dans la conférence en indiquant qu'il est l'organisateur et fournit un mot de passe d'organisateur correct, le fournisseur GCC sommital peut choisir d'accepter la connexion en cédant les privilèges d'organisateur au nouveau nœud. Il faut également noter dans ce cas, qu'il est supposé que les paramètres du domaine MCS sont positionnés au moment de la création plutôt qu'au moment de l'établissement de la première connexion.



T0819850-94

**Figure 8-1 – Création d'une conférence**

## 8.2.2 Interrogation de conférences

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Query émettra une primitive MCS-Connect-Provider (connexion de fournisseur) avec les paramètres donnés par le Tableau 8-6.

**Tableau 8-6 – Paramètres de la primitive de demande MCS-Connect-Provider pour la PDU ConferenceQueryRequest**

Paramètre	Contenu
Adresse appelante	De la primitive de demande
Sélecteur de domaine appelant	NULL (néant)
Adresse appelée	De la primitive de demande
Sélecteur de domaine appelé	NULL (néant)
Indicateur de direction	Montée
Paramètres de domaine	Paramètres par défaut du domaine
Qualité de service	Paramètres par défaut de qualité de service
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceQueryRequest (voir le Tableau 8-7)

Le paramètre "données utilisateur" de la primitive de demande MCS-Connect-Provider contient une structure constituée d'un identificateur d'objet identifiant (conformément à la présente Recommandation) la PDU qui fait partie de la structure, suivie de cette PDU elle-même. Cette structure est définie en détail au § 9.6. Le contenu de la PDU est donné par le Tableau 8-7.

**Tableau 8-7 – GCCPDU ConferenceQueryRequest**

Contenu	Origine	Destination
Type de nœud	Primitive de demande	Primitive d'indication
Indicateur d'asymétrie (conditionnel)	Primitive de demande	Primitive d'indication
Données utilisateur	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit la primitive d'indication MCS-Connect-Provider contenant un identificateur d'objet T.124 et la PDU ConferenceQueryRequest (demande d'interrogation de conférence), le fournisseur GCC générera une primitive d'indication GCC-Conference-Query contenant les paramètres spécifiés dans la PDU ConferenceQueryRequest. Il enverra cette primitive au point de contrôle GCCSAP. Lorsqu'il reçoit une primitive de réponse GCC-Conference-Query du point de contrôle GCCSAP, le fournisseur GCC enverra une primitive de réponse MCS-Connect-Provider qui est rejetée (signifiant qu'aucune conférence n'est établie) mais contient la PDU ConferenceQueryResponse (réponse à une interrogation de conférence) dans le champ "données utilisateur". Les paramètres de la primitive de réponse MCS-Connect-Provider sont donnés par le Tableau 8-8.

**Tableau 8-8 – Paramètres de réponse de la primitive MCS-Connect-Provider pour la PDU ConferenceQueryResponse**

Paramètre	Contenu
Paramètres de domaine	Paramètres par défaut du domaine
Qualité de service	Paramètres par défaut de qualité de service
Résultat	Utilisateur rejeté
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceQueryResponse (voir le Tableau 8-9)

L'unité PDU ConferenceQueryResponse est donnée par le Tableau 8-9. La liste de descripteurs de conférence contient une entrée pour chaque conférence listée à laquelle appartient le nœud faisant l'objet de la demande. La liste de descripteurs de conférence est produite par le fournisseur GCC qui est à l'origine de l'unité PDU ConferenceQueryResponse. Le descripteur reçoit les informations suivantes: le nom de conférence de chaque conférence à laquelle appartient le fournisseur GCC, le modificateur de nom de conférence géré localement pour chaque conférence (le cas échéant), la description de conférence (le cas échéant), l'indicateur de verrouillage/déverrouillage, l'indicateur "mot de passe en clair exigé", le champ d'adresse réseau (s'il existe), l'indicateur de conférence par défaut et le mode de conférence. Le champ d'adresse réseau, s'il existe, est extrait du paramètre "adresse locale de réseau" de la primitive de demande GCC-Conference-Create, de la primitive de réponse GCC-Conference-Create, de la primitive de demande GCC-Conference-Join ou de la primitive de réponse GCC-Conference-Invite. Le paramètre *Résultat* indiquera un succès si la demande d'interrogation peut être satisfaite, même si la liste de conférences est vide, ou le motif "utilisateur rejeté" si cela est indiqué dans la primitive de réponse.

Si le fournisseur GCC traitant de la primitive d'indication MCS-Connect-Provider reçoit une primitive d'indication MCS-Disconnect-Provider (déconnexion de fournisseur) pour la même connexion à tout instant au cours du processus, le fournisseur GCC cessera tout traitement et n'exécutera aucune autre action.

**Tableau 8-9 – GCCPDU ConferenceQueryResponse**

Contenu	Origine	Destination
Type de nœud	Primitive de demande	Primitive de confirmation
Indicateur d'asymétrie (conditionnel)	Primitive de réponse	Primitive de confirmation
Liste de descripteurs de conférence	Fournisseur GCC origine	Primitive de confirmation
Résultat	Primitive de réponse	Primitive de confirmation
Données utilisateur (optionnel)	Primitive de réponse	Primitive de confirmation

Lorsqu'il reçoit une primitive de réponse MCS-Connect-Provider contenant un identificateur d'objet T.124 et la PDU ConferenceQueryResponse, un fournisseur GCC générera une primitive de réponse GCC-Conference-Query qu'il enverra au point de contrôle GCCSAP. Le contenu de la primitive de confirmation sera déduit des paramètres de la PDU ConferenceQueryResponse. En particulier, le paramètre résultat est dupliqué à partir du paramètre résultat de la PDU reçue. Le paramètre résultat de la primitive MCS-Connect-Provider est ignoré car il contient "utilisateur rejeté" même en cas de succès.

Si la PDU reçue a indiqué un résultat négatif ou si le fournisseur GCC reçoit une primitive d'indication MCS-Disconnect-Provider (déconnexion de fournisseur) concernant cette connexion avant d'avoir envoyé une primitive de confirmation GCC-Conference-Query positive, elle envoie immédiatement la primitive de confirmation GCC-Conference-Query indiquant un résultat négatif.

Le paramètre résultat de la PDU ainsi que le paramètre résultat renvoyé par la confirmation MCS-Connect-Provider (ou le paramètre "cause" de la primitive MCS-Disconnect-Provider) sont utilisés pour générer le résultat renvoyé par la primitive de confirmation GCC-Conference-Query. Le paramètre résultat de la PDU est utilisé si le paramètre résultat de la confirmation MCS-Connect-Provider est "utilisateur rejeté". Sinon le paramètre résultat de la primitive MCS-Connect-Provider est utilisé directement.

### 8.2.3 Entrée dans une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Join (addition à une conférence) émettra une primitive MCS-Connect-Provider (connexion de fournisseur) avec les paramètres donnés par le Tableau 8-10. Le fournisseur GCC local attribuera l'identificateur de conférence qui sera utilisé comme sélecteur de domaine MCS local associé à la conférence créée. L'identificateur de conférence sera envoyé comme sélecteur de domaine appelant de la primitive de demande MCS-Connect-Provider (connexion de fournisseur).

**Tableau 8-10 – Paramètres de la primitive de demande MCS-Connect-Provider pour la PDU ConferenceJoinRequest**

Paramètre	Contenu
Adresse appelante	De la primitive de demande
Sélecteur de domaine appelant	Identificateur de conférence choisi par le fournisseur GCC
Adresse appelée	De la primitive de demande
Sélecteur de domaine appelé	NULL (néant)
Indicateur de direction	Montée
Paramètres de domaine	De la primitive de demande
Qualité de service	De la primitive de demande
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceJoinRequest (voir le Tableau 8-11)

Le paramètre "données utilisateur" de la primitive de demande MCS-Connect-Provider contient une structure constituée d'un identificateur d'objet identifiant – conformément à la présente Recommandation – la PDU qui fait partie de la structure, suivie de cette PDU elle-même. Cette structure est définie en détail au § 9.6. Le contenu de la PDU est donné par le Tableau 8-11. Le paramètre "marque" du nœud entrant ne sera pas utilisé. Le nom de conférence et le modificateur de nom de conférence proviennent de la primitive de demande. Le modificateur de nom de conférence de la PDU provient du paramètre "modificateur de nom de conférence du nœud appelé" de la primitive. Le modificateur de nom de conférence du nœud appelant, issu également de la primitive de demande, est géré comme l'identificateur de la conférence au niveau du fournisseur GCC local pour une utilisation dans les procédures de réponse à une demande d'entrée dans une conférence ou une demande d'interrogation de conférence.



**Tableau 8-11 – GCCPDU ConferenceJoinRequest**

Contenu	Origine	Destination
Nom de conférence (conditionnel)	Primitive de demande	Fournisseur GCC recevant la primitive d'indication MCS-Connect-Provider
Modificateur de nom de conférence (optionnel)	Primitive de demande	Fournisseur GCC recevant la primitive d'indication MCS-Connect-Provider
Marque (conditionnel)	Fournisseur GCC recevant la primitive d'indication MCS-Connect-Provider	Fournisseur GCC sommital
Mot de passe (optionnel)	Primitive de demande	Primitive d'indication
Identificateur d'organisateur (optionnel)	Primitive de demande	Primitive d'indication
Identificateur de l'appelant (optionnel)	Primitive de demande	Primitive d'indication
Catégorie nodale (optionnel)	Primitive de demande	Primitive d'indication
Données utilisateur (optionnel)	Primitive de demande	Primitive d'indication

L'action exécutée par le fournisseur GCC qui reçoit la primitive d'indication MCS-Connect-Provider contenant un identificateur d'objet T.124 et la PDU ConferenceJoinRequest (demande d'entrée dans une conférence) diffère s'il est ou non le fournisseur GCC sommital de la conférence dans laquelle doit se faire l'entrée. Le fournisseur GCC détermine celle-ci en comparant le nom de conférence et le modificateur de nom de conférence éventuel avec la liste des conférences dont fait partie ce nœud. Le nom de conférence reçu dans la PDU ConferenceJoinRequest peut être sous la forme numérique ou la forme texte. Le fournisseur GCC tentera de trouver une concordance entre le nom indiqué et la partie correspondante des noms des conférences existantes. Pour qu'une conférence soit en concordance, à la fois le nom de conférence et le nom de modificateur de conférence doivent être identiques à ceux d'une conférence existante. Si le modificateur de conférence n'est pas donné dans la demande, le fournisseur GCC recherchera une concordance avec une conférence n'ayant pas non plus de modificateur de nom.

Si le fournisseur GCC recevant la primitive d'indication MCS-Connect-Provider n'est pas le fournisseur GCC sommital de la conférence dans laquelle souhaite entrer le nœud demandeur mais en fait partie, le fournisseur GCC routera la PDU ConferenceJoinRequest vers le fournisseur GCC sommital de la conférence en question. Pour ce faire, il émettra une PDU de primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC sommital, une priorité de données maximale et la PDU dans le champ de données. Dans ce cas, il attribuera un identificateur localement non ambigu et le placera dans le paramètre "marque" de la PDU. Ce nombre est utilisé pour identifier la PDU de réponse correspondante quand elle est renvoyée par le fournisseur GCC sommital. Il peut omettre les paramètres "nom de conférence" et "modificateur de nom de conférence" de la PDU reçue car le fournisseur GCC sommital ne les utilise pas.

Lorsque le fournisseur GCC sommital reçoit cette PDU et que la conférence n'est pas verrouillée, il générera une primitive d'indication GCC-Conference-Join avec les paramètres fournis dans la PDU ConferenceJoinRequest, ainsi que l'identificateur de la conférence correspondante. Il enverra cette primitive au point de contrôle GCCSAP. Il est à noter que les paramètres "nom de conférence" et "modificateur de nom de conférence" sont utilisés uniquement par le nœud recevant la primitive d'indication MCS-Connect-Provider et non par le fournisseur GCC sommital. Si la conférence a été verrouillée, le fournisseur GCC sommital enverra par contre une PDU ConferenceJoinRequest au

générateur de la primitive MCS-Send-Data qui contenait la PDU ConferenceJoinRequest (pas au demandeur initial). Pour ce faire il émettra une primitive de demande MCS-Send-Data avec en paramètre l'identificateur de ce nœud, une priorité de données maximale et la PDU dans le champ de données. Le code résultat de la PDU indiquera "conférence verrouillée". Sinon, lorsqu'il reçoit la primitive de réponse GCC-Conference-Join, le fournisseur GCC enverra une PDU ConferenceJoinResponse (réponse à une entrée en conférence) au générateur de la primitive MCS-Send-Data qui contenait la PDU ConferenceJoinRequest (pas au demandeur initial). Pour ce faire il émet une primitive de requête MCS-Send-Data avec en paramètre l'identificateur de ce nœud, une priorité de données maximale et la PDU dans le champ de données. Le contenu de la PDU ConferenceJoinRequest est donné dans le Tableau 8-13. L'identificateur de nœud n'est pas présent dans cette partie de la réponse mais le paramètre "identificateur de nœud sommital", le paramètre "marque" et les paramètres associés avec le profil de conférence sont présents ainsi que le paramètre résultat. Le paramètre "marque" possède la même valeur que le paramètre correspondant de la PDU ConferenceJoinRequest reçue.

Lorsqu'il reçoit la primitive d'indication MCS-Send-Data contenant cette PDU, le fournisseur GCC examinera sa liste locale de demandes d'entrée en conférence en attente pour lesquelles il doit envoyer la primitive de réponse MCS-Connect-Provider. Si le paramètre "marque" reçu a permis d'identifier la connexion en attente dans la liste locale, le fournisseur GCC générera une primitive de réponse MCS-Connect-Provider. Le résultat fourni par cette réponse est soit positif, soit égal à "utilisateur rejeté" selon que le paramètre résultat de la PDU reçue indique un succès ou un échec. Le paramètre "données utilisateur" contient un identificateur d'objet T.124 ainsi que la PDU ConferenceJoinResponse. Le paramètre "identificateur de nœud" de la PDU ConferenceJoinResponse contient à cet instant la valeur de l'identificateur du nœud auquel le nœud entrant est connecté directement. Le contenu de la primitive MCS-Connect-Provider est donné par le Tableau 8-12.

Si le récepteur de la primitive d'indication MCS-Connect-Provider initiale est le fournisseur GCC sommital et si la conférence n'est pas verrouillée, celui-ci générera une primitive d'indication GCC-Conference-Join en utilisant les paramètres indiqués dans la PDU ConferenceJoinRequest ainsi que l'identificateur de la conférence désignée. Il enverra cette primitive au point de contrôle GCCSAP. Si la conférence est verrouillée, le fournisseur GCC sommital générera une primitive de réponse MCS-Connect-Provider contenant une primitive de réponse GCC-Conference-Join avec le code résultat "conférence verrouillée". La PDU contiendra également le code résultat "conférence verrouillée". Dans le cas contraire, lorsque le fournisseur GCC reçoit la primitive de réponse GCC-Conference-Join, il générera une primitive de réponse MCS-Connect-Provider contenant la PDU ConferenceJoinResponse avec un résultat qui est soit positif soit égal à "utilisateur rejeté" selon que le paramètre résultat de la primitive de réponse GCC-Conference-Join indique un succès ou une erreur.

Si le récepteur de l'indication initiale MCS-Connect-Provider ne fait actuellement partie d'aucune conférence ou ne prend pas en charge la primitive d'indication GCC-Conference-Join, il peut rejeter la demande connexion en émettant immédiatement une primitive de réponse MCS-Connect-Provider contenant le code résultat "utilisateur rejeté" et une PDU ConferenceJoinResponse avec un code cause indiquant "conférence non valide".

Si le récepteur de l'indication initiale MCS-Connect-Provider ne fait pas partie de la conférence définie par les paramètres "nom de conférence" et "modificateur de nom de conférence", il rejettera la demande par l'émission d'une primitive de réponse MCS-Connect-Provider donnant comme cause "pas de tel domaine".

**Tableau 8-12 – Paramètres de réponse de la primitive MCS-Connect-Provider pour la PDU ConferenceJoinResponse**

Paramètre	Contenu
Paramètres de domaine	Tels que reçus précédemment du fournisseur GCC et stockés
Qualité de service	Telle que reçue précédemment du fournisseur GCC et stockée
Résultat	Comme spécifié dans [UIT-T T.122]
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceJoinResponse (voir le Tableau 8-13)

Le Tableau 8-13 donne le contenu de la PDU ConferenceJoinResponse. Le paramètre "identificateur de nœud", qui est l'identificateur d'utilisateur attribué par le service MCS en réponse à une primitive de demande MCS-Attach-User (rattachement d'utilisateur) émise par le fournisseur GCC, est fourni par le fournisseur GCC du nœud connecté directement au nœud entrant dans la conférence. Le paramètre "marque" reçoit du fournisseur GCC du nœud connecté directement au nœud entrant une valeur localement non ambiguë qui sera utilisée pour identifier la PDU UserIDIndication (indication d'identificateur utilisateur) renvoyée par la suite. La valeur de ce paramètre peut être la même que celle reçue du fournisseur GCC sommital puisque ce paramètre a été attribué à l'origine par ce nœud et envoyé dans la PDU ConferenceJoinRequest. Dans ce cas, il est nécessaire que les marques employées pour l'un ou l'autre usage soient localement non ambiguës. Les autres paramètres de la PDU sont déterminés par le fournisseur GCC sommital. Ils comprennent l'identificateur du nœud sommital, qui est l'identificateur de nœud du fournisseur GCC sommital, ainsi que les paramètres associés au profil de conférence. Le fournisseur GCC sommital fixe également la valeur du paramètre "alias de nom de conférence". Ce paramètre est présent si le nom de conférence est donné à la fois sous les formes numérique et texte. Dans ce cas, l'alias de nom de conférence reçoit la valeur de celle des formes du nom de conférence qui ne figure pas dans la PDU ConferenceJoinRequest. Le paramètre résultat contient une information d'échec spécifique du service GCC si le paramètre résultat du message MCS-Connect-Provider indique "utilisateur rejeté". Cette information provient du paramètre résultat de la PDU de primitive de réponse GCC-Conference-Create.

**Tableau 8-13 – GCCPDU ConferenceJoinResponse**

Contenu	Origine	Destination
Identificateur de nœud (conditionnel)	Fournisseur GCC du nœud connecté directement au nœud entrant	Fournisseur GCC destinataire
Identificateur du nœud sommital	Fournisseur GCC sommital	Fournisseur GCC destinataire
Marque	Fournisseur GCC sommital	Fournisseur GCC destinataire
Alias du nom de conférence (conditionnel)	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Indicateur de mot de passe en clair requis	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Indicateur de conférence verrouillée	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire

**Tableau 8-13 – GCCPDU ConferenceJoinResponse**

Contenu	Origine	Destination
Indicateur de conférence listée	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Indicateur de conférence pouvant être présidée	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Méthode de fin	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Liste de privilèges du président (optionnel)	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Liste de privilèges de conférence en mode présidé (optionnel)	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Liste de privilèges de conférence en mode non présidé (optionnel)	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Description de conférence (optionnel)	Fournisseur GCC sommital	Primitive de confirmation et fournisseur GCC destinataire
Mot de passe (optionnel)	Primitive de demande	Primitive d'indication
Catégorie nodale (optionnel)	Primitive de demande	Primitive d'indication
Résultat	Primitive de réponse	Primitive de confirmation
Données utilisateur (optionnel)	Primitive de réponse	Primitive de confirmation

Lorsqu'il reçoit l'unité PDU ConferenceJoinResponse indiquant un résultat positif, un fournisseur GCC émettra en premier lieu une primitive de demande MCS-Attach-User (rattachement d'utilisateur). Après avoir reçu la confirmation MCS-Attach-User qui contient l'identificateur de nœud attribué, le fournisseur GCC se branchera sur le canal correspondant à l'identificateur de nœud en émettant une demande MCS-Channel-Join. Le fournisseur GCC doit également se brancher sur le ou les canaux de diffusion GCC appropriés en émettant une ou deux demandes MCS-Channel-Join. Tous les nœuds sont appelés à entrer dans le canal GCC-Conventional-Broadcast-Channel. Toutes les unités PDU de commande GCC et toutes les mises à jour delta de répertoire qui sont produites lorsque des nœuds conventionnels joignent, quittent ou modifient un enregistrement de répertoire sont diffusées par ce canal. Les nœuds anonymes peuvent ignorer les mises à jour delta de répertoire reçues sur ce canal mais doivent traiter les messages de commande reçus. Les nœuds conventionnels doivent également entrer dans le canal GCC-Counted-Broadcast. Sur ce canal, le fournisseur sommital diffuse des mises à jour delta de répertoire qui sont produites lorsque des nœuds comptés joignent, quittent, ou modifient un enregistrement de répertoire. Etant donné que seuls les nœuds conventionnels entrent dans le canal GCC-Counted-Broadcast, l'utilisation de ce canal réduit l'intensité du trafic dans le réseau lorsqu'un grand nombre de nœuds comptés sont en train de participer à la conférence. A noter que les nœuds utilisant d'anciens protocoles entrent dans le canal de diffusion GCC à ce point.

Si la demande GCC-Conference-Join d'origine comporte le mot de passe de l'organisateur, indiquant par là que le nœud doit reprendre son rôle d'organisateur, le fournisseur GCC peut aussi rejoindre le canal GCC-Convener-Channel (s'il prend en charge au moins l'une des fonctions qui nécessitent l'utilisation de ce canal) en émettant une demande MCS-Channel-Join. Une fois que le fournisseur GCC a reçu une confirmation MCS-Channel-Join pour chaque demande d'adjonction à un canal (les demandes successives peuvent être lancées avant d'avoir reçu les confirmations

précédentes), il enverra une PDU UserIDIndication au fournisseur GCC du nœud directement connecté en envoyant une demande MCS-Send-Data spécifiant l'identificateur du nœud directement connecté conformément à la PDU ConferenceJoinResponse reçue, en spécifiant la priorité maximale et en inscrivant l'unité PDU dans le champ de données. Le contenu de l'unité PDU UserIDIndication est indiqué au Tableau 8-5. Le paramètre d'étiquette recevra la valeur du paramètre correspondant reçu dans l'unité PDU ConferenceJoinResponse.

Le fournisseur GCC générera ensuite une primitive de confirmation GCC-Conference-Join qu'il enverra au point de contrôle GCCSAP. Cette primitive contiendra les paramètres "résultat" de la PDU reçue ainsi que l'identificateur de conférence attribué localement. Si la PDU reçue a indiqué un résultat négatif, ou si le fournisseur GCC reçoit une primitive d'indication MCS-Disconnect-Provider pour cette connexion avant d'avoir envoyé une primitive de confirmation GCC-Conference-Join positive, la primitive de confirmation GCC-Conference-Join sera émise immédiatement pour indiquer un résultat négatif. Les demandes de connexion utilisateur ou canal ne sont pas émises et la PDU UserIDIndication n'est pas envoyée dans ce cas. Le paramètre résultat de la PDU ainsi que le paramètre résultat fourni par la confirmation MCS-Connect-Provider (ou le paramètre "cause" de la primitive MCS-Disconnect-Provider) sont utilisés pour générer le résultat donné dans la primitive de confirmation GCC-Conference-Join. Si le paramètre résultat de la confirmation MCS-Connect-Provider est "utilisateur rejeté", le paramètre résultat de la PDU est utilisé pour déterminer le résultat renvoyé. Sinon, le paramètre résultat de la primitive d'indication MCS-Connect-Provider est utilisé directement.

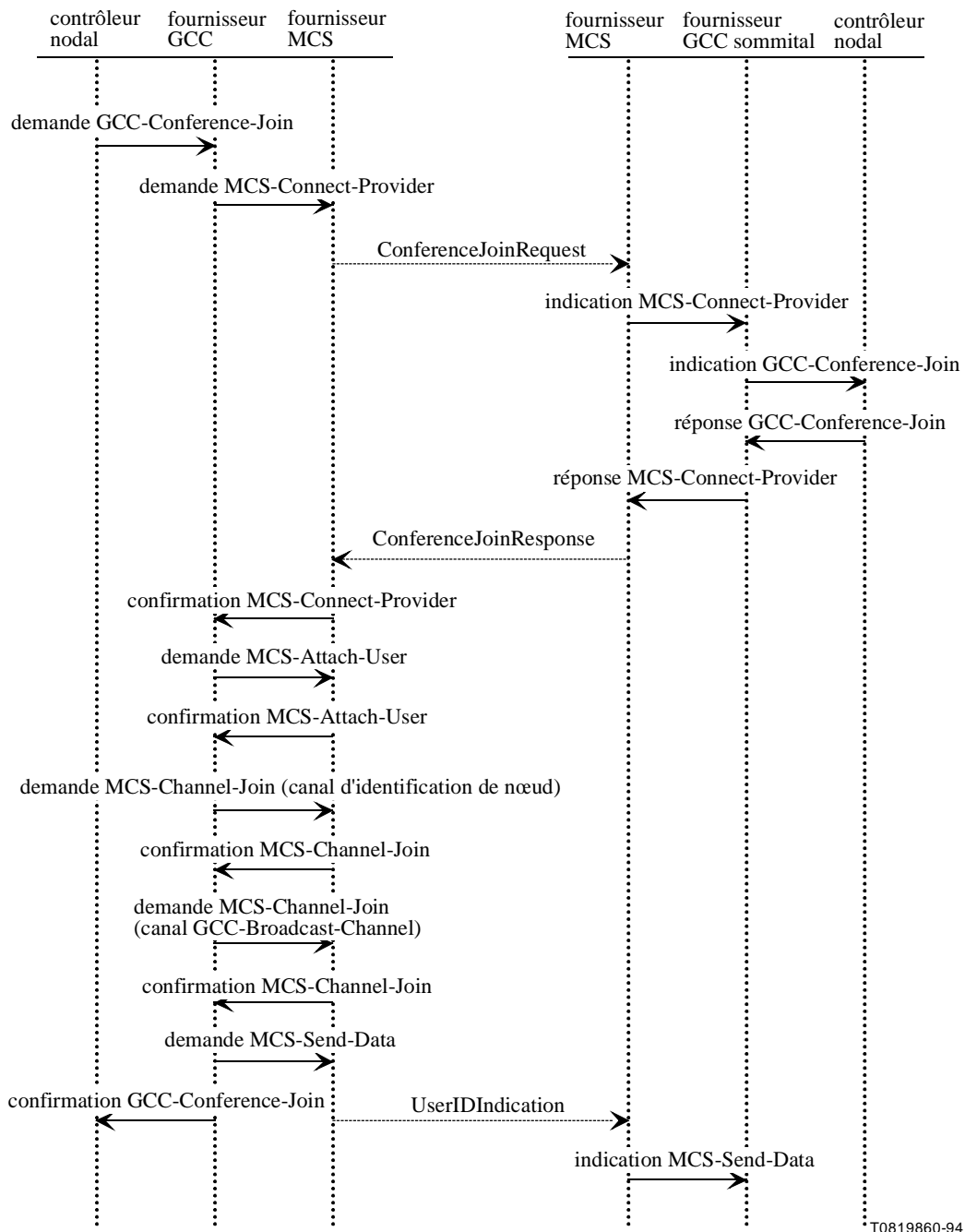
L'alias du nom de conférence reçu éventuellement dans la PDU ConferenceJoinResponse est concaténé avec le nom de conférence contenu dans la PDU ConferenceJoinRequest pour former le nom de conférence complet renvoyé dans la primitive de confirmation GCC-Conference-Join. Ce nom de conférence complet sera géré par le fournisseur GCC local comme l'identificateur de conférence utilisé par les procédures de réponse à une demande d'entrée en conférence, de demande d'interrogation de conférence et d'invitation dans une conférence.

Lorsqu'il reçoit une indication UserIDIndication, le nœud directement connecté au nœud entrant comparera le paramètre "marque" avec ceux de sa liste de nœuds de la conférence dont il attend une telle PDU. Si la marque coïncide avec l'une de ceux-ci, il mémorisera l'identificateur du nœud origine extrait de la primitive d'indication MCS-Send-Data dans sa base de données des nœuds qui lui sont directement subordonnés. La PDU sera ignorée si la marque ne coïncide pas avec une marque en attente.

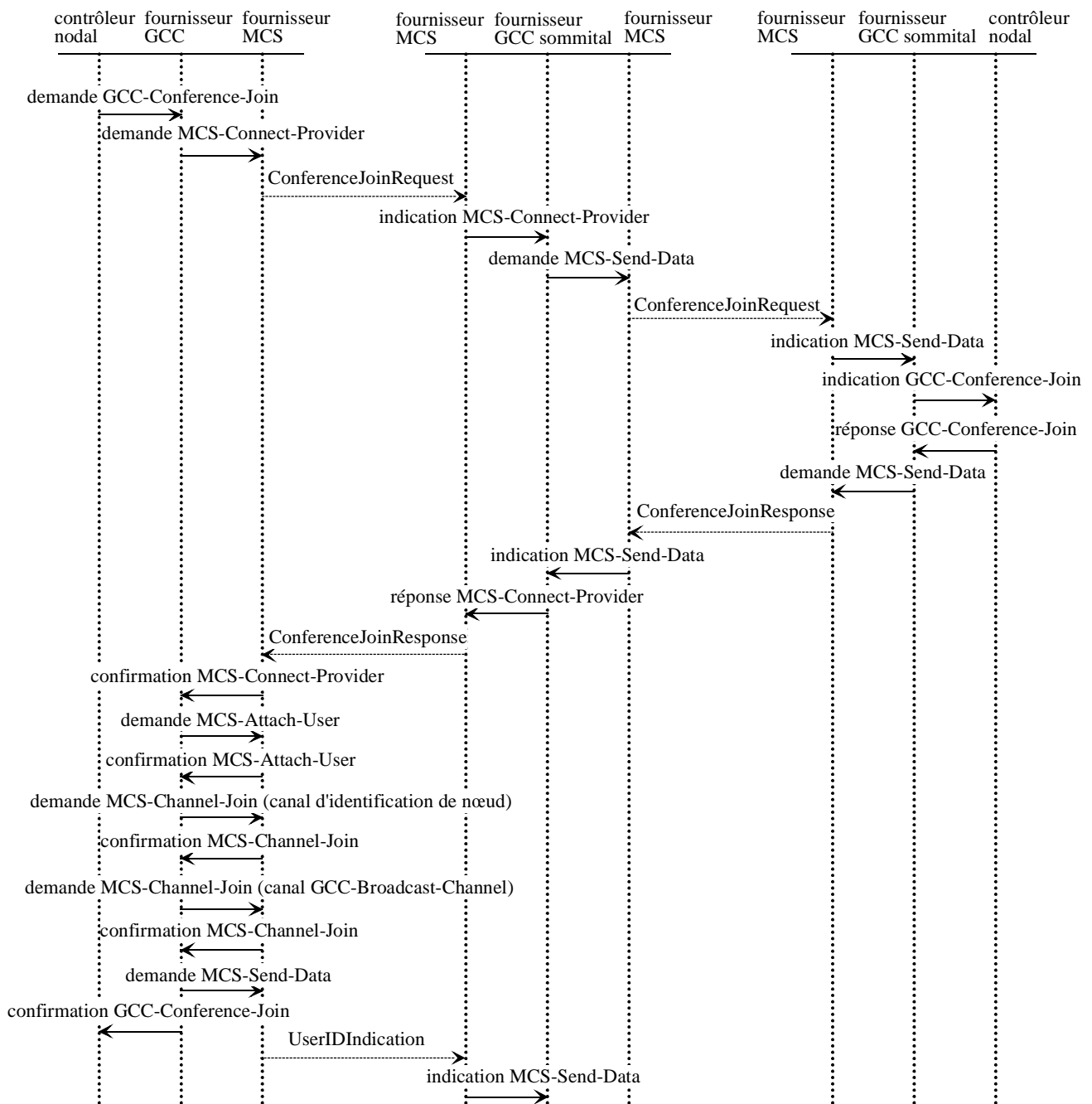
Si le fournisseur GCC sommital reçoit, comme résultat d'une opération d'entrée qui a réussi, une indication UserIDIndication dans laquelle le paramètre "mot de passe de l'organisateur" figure dans la PDU ConferenceJoinRequest, il mémorisera l'identificateur du nœud entrant figurant dans l'indication UserIDIndication comme l'organisateur de conférence valide. Il utilisera par la suite cet identificateur pour vérifier des demandes dont l'utilisation est réservée à l'organisateur.

Si le fournisseur GCC du nœud connecté directement au nœud entrant reçoit, à tout moment du processus d'entrée, une primitive d'indication MCS-Disconnect-Provider concernant la connexion en cours d'établissement, il cessera tout traitement et n'exécutera aucune autre action.

La Figure 8-2 présente la succession d'événements se déroulant lors d'une entrée en conférence d'un nœud connecté directement au fournisseur GCC sommital. Le cas où le nœud entrant n'est pas connecté directement au fournisseur GCC sommital est présenté dans la Figure 8-3.



**Figure 8-2 – Entrée dans une conférence en cas de connexion directe au fournisseur GCC sommital**



T0819870-94

**Figure 8-3 – Entrée dans une conférence dans le cas sans connexion directe au fournisseur GCC sommital**

#### 8.2.4 Invitation d'un nœud dans une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Invite (invitation dans une conférence) émettra une primitive de demande MCS-Connect-Provider (connexion de fournisseur) avec les paramètres donnés par le Tableau 8-14.

**Tableau 8-14 – Paramètres de la primitive de demande MCS-Connect-Provider pour la PDU ConferenceInviteRequest**

Paramètre	Contenu
Adresse appelante	De la primitive de demande
Sélecteur de domaine appelant	Identificateur de conférence de la primitive de demande
Adresse appelée	De la primitive de demande
Sélecteur de domaine appelé	NULL (néant)
Indicateur de direction	Descente
Paramètres de domaine	Tels que reçus précédemment du fournisseur GCC et stockés
Qualité de service	Telle que reçue précédemment du fournisseur GCC et stockée
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceInviteRequest (voir le Tableau 8-15)

Le paramètre "données utilisateur" de la primitive de demande MCS-Connect-Provider contient une structure constituée d'un identificateur d'objet identifiant – conformément à la présente Recommandation – la PDU qui fait partie de la structure, suivie de cette PDU elle-même. Cette structure est définie en détail au § 9.6. Le contenu de la PDU est donné par le Tableau 8-15. Le paramètre "nom de conférence" est le nom de la conférence indiquée dans le paramètre "identificateur de conférence" de la primitive de demande, tel qu'il est mémorisé dans le profil de conférence local. Le paramètre "identificateur de nœud" sera livré par le fournisseur GCC origine – il s'agit de l'identificateur d'utilisateur attribué par le service MCS en réponse à la primitive de demande MCS-Attach-User (rattachement d'utilisateur) envoyée par le fournisseur GCC. L'identificateur de nœud sommital est l'identificateur de nœud du fournisseur GCC sommital mémorisé précédemment par le fournisseur GCC du nœud invitant. Le paramètre "marque" est attribué par le fournisseur GCC d'origine d'une manière localement non ambiguë. Il est utilisé pour identifier la PDU UserIDIndication (indication d'identificateur d'utilisateur) envoyée en retour.

**Tableau 8-15 – GCCPDU ConferenceInviteRequest**

Contenu	Origine	Destination
Nom de conférence	Fournisseur GCC origine	Fournisseur GCC destinataire et indication
Identificateur du nœud	Fournisseur GCC origine	Fournisseur GCC destinataire
Identificateur du nœud sommital	Fournisseur GCC origine	Fournisseur GCC destinataire
Marque	Fournisseur GCC origine	Fournisseur GCC destinataire
Indicateur de mot de passe en clair exigé	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Indicateur de conférence verrouillée	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Indicateur de conférence listée	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Indicateur de conférence pouvant être présidée	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Méthode de fin	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Liste de privilèges du président (optionnel)	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire



**Tableau 8-15 – GCCPDU ConferenceInviteRequest**

Contenu	Origine	Destination
Liste de privilèges de conférence en mode présidé (optionnel)	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Liste de privilèges de conférence en mode non présidé (optionnel)	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Descripteur de conférence (optionnel)	Fournisseur GCC origine	Primitive d'indication et fournisseur GCC destinataire
Identificateur de l'appelant (optionnel)	Primitive de demande	Primitive d'indication
Catégorie nodale	Primitive de demande	Primitive d'indication
Données utilisateur (optionnel)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit la primitive d'indication MCS-Connect-Provider contenant un identificateur d'objet T.124 et la PDU ConferenceInviteRequest (demande d'invitation à une conférence), le fournisseur GCC générera une primitive d'indication GCC-Conference-Invite qu'il enverra au point de contrôle GCCSAP. L'identificateur de conférence de cette primitive sera attribué localement par le fournisseur GCC et utilisé comme sélecteur de domaine MCS local. Si le service GCC ne dispose pas des ressources nécessaires pour l'entrée dans la conférence, il peut générer automatiquement la réponse négative sans générer d'indication GCC-Conference-Invite. Sinon le fournisseur GCC générera, lorsqu'il reçoit la primitive de réponse GCC-Conference-Invite, une primitive de réponse MCS-Connect-Provider contenant un résultat qui est soit succès, soit "utilisateur rejeté" selon que le paramètre résultat de la primitive de réponse GCC-Conference-Invite a indiqué un succès ou un échec. Ce résultat sera également placé dans la PDU ConferenceInviteResponse. Le paramètre "données utilisateur" contient l'identificateur d'objet T.124 ainsi que la PDU ConferenceInviteResponse. Le contenu de la primitive MCS-Connect-Provider est donné par le Tableau 8-16.

En cas de réponse positive, le fournisseur GCC gèrera un identificateur local de la conférence utilisé pour la réponse à une demande d'entrée en conférence ou une demande d'interrogation de conférence ainsi que pour la création d'une demande d'invitation à la conférence. Il utilisera à cet effet le paramètre "nom de conférence" fourni sous la forme numérique et éventuellement texte fourni dans la PDU ConferenceInviteRequest ainsi que le paramètre "modificateur de nom de conférence" fourni éventuellement dans la réponse. Si le modificateur de nom de conférence n'est pas présent, seul le nom de conférence sera géré à cette fin.

Après l'envoi de la primitive de réponse MCS-Connect-Provider, si l'invitation est acceptée, le fournisseur GCC émettra une primitive de demande MCS-Attach-User. Lorsqu'il reçoit la confirmation MCS-Attach-User contenant l'identificateur de nœud qui a été attribué, le fournisseur GCC se branche sur le canal correspondant à l'identificateur de nœud en émettant une demande MCS-Channel-Join. Il doit également se brancher sur le ou les canaux de diffusion GCC appropriés en émettant une ou deux demandes MCS-Channel-Join. Tous les nœuds sont appelés à entrer dans le canal GCC-Conventional-Broadcast-Channel. Toutes les unités PDU de commande GCC et toutes les mises à jour delta de répertoire qui sont produites lorsque des nœuds conventionnels joignent, quittent ou modifient un enregistrement de répertoire sont diffusées par ce canal. Les nœuds anonymes peuvent ignorer les mises à jour delta de répertoire reçues sur ce canal mais doivent traiter les messages de commande reçus. Les nœuds conventionnels doivent également entrer dans le canal GCC-Counted-Broadcast. Sur ce canal, le fournisseur sommital diffuse des mises à jour delta de répertoire qui sont produites lorsque des nœuds comptés joignent, quittent, ou modifient un enregistrement de répertoire. Etant donné que seuls les nœuds conventionnels entrent dans le canal GCC-Counted-Broadcast, l'utilisation de ce canal réduit l'intensité du trafic dans le

réseau lorsqu'un grand nombre de nœuds comptés sont en train de participer à la conférence. A noter que les nœuds utilisant d'anciens protocoles entrent dans le canal de diffusion GCC à ce point.

Une fois que le fournisseur GCC a reçu une primitive de confirmation MCS-Channel-Join pour chacune des demandes MCS-Channel-Join, il enverra une PDU UserIDIndication (indication d'identificateur utilisateur) au fournisseur GCC du nœud connecté directement. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre l'identificateur du nœud connecté directement tel qu'il est fourni par la PDU ConferenceInviteRequest reçue, une priorité de données maximale et la PDU dans le champ de données. Le contenu de la PDU UserIDIndication est donné par le Tableau 8-5. Le paramètre "marque" reçoit la valeur du paramètre correspondant de la PDU ConferenceInviteRequest reçue.

Le fournisseur GCC du nœud invité enverra au point de contrôle GCCSAP une primitive d'indication GCC-Conference-Terminate (fin de conférence) s'il reçoit une primitive d'indication MCS-Disconnect-Provider (déconnexion de fournisseur) pour la connexion en cours d'établissement à un moment quelconque du processus d'entrée dans la conférence allant de l'instant d'émission de la primitive d'indication GCC-Conference-Invite à l'instant de réception de l'indication UserIDIndication. Il indiquera comme cause une demande de fin normale si la cause fournie par la primitive d'indication MCS-Disconnect-Provider est "demandé par l'utilisateur" et une fin anormale dans les autres cas. Le fournisseur GCC sommital cessera ensuite tout traitement de cet établissement de connexion. Si la primitive d'indication MCS-Disconnect-Provider est reçue avant l'envoi de la primitive d'indication GCC-Conference-Invite au point de contrôle GCCSAP, le fournisseur GCC cessera tout traitement de cet établissement de connexion et n'exécutera aucune autre action.

**Tableau 8-16 – Paramètres de réponse de la primitive MCS-Connect-Provider pour la PDU ConferenceInviteResponse**

Paramètre	Contenu
Paramètres de domaine	De la primitive de réponse
Qualité de service	De la primitive de réponse
Résultat	Comme spécifié dans [UIT-T T.122]
Données utilisateur	Identificateur d'objet T.124
	PDU ConferenceInviteResponse (voir le Tableau 8-17)

Le contenu de la PDU ConferenceInviteResponse est donné par le Tableau 8-17.

**Tableau 8-17 – GCCPDU ConferenceInviteResponse**

Contenu	Origine	Destination
Résultat	Primitive de réponse	Primitive de confirmation
Données utilisateur (optionnel)	Primitive de réponse	Primitive de confirmation

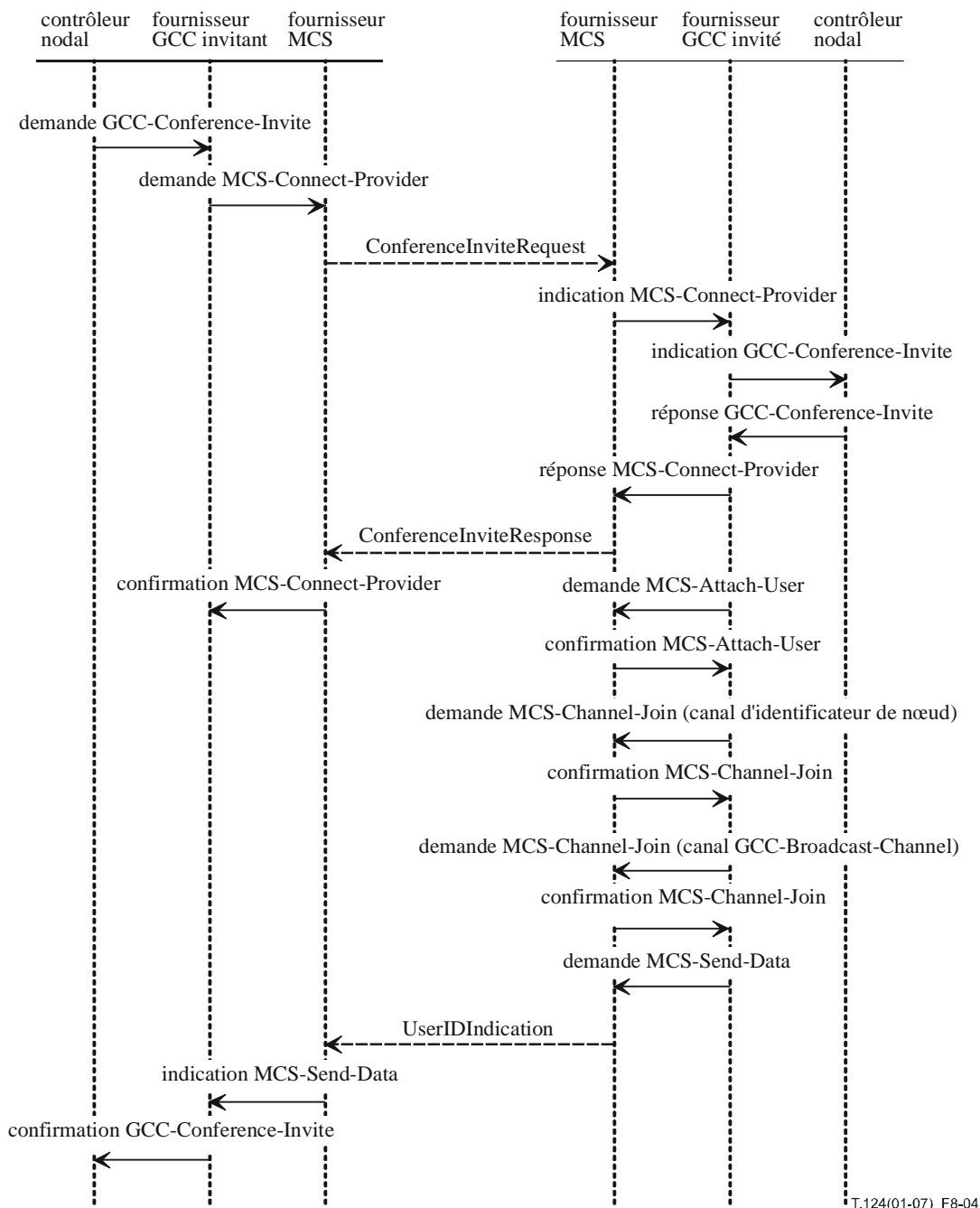
Le fournisseur GCC qui reçoit la PDU ConferenceInviteResponse mémorisera le paramètre "données utilisateur" si le paramètre résultat de la primitive MCS-Connect-Provider indique un succès. Il utilisera la valeur mémorisée pour remplir la primitive de confirmation GCC-Conference-Invite lorsqu'elle sera générée à la suite de la réception de l'indication UserIDIndication en attente. Si le paramètre résultat indique un échec ou si le fournisseur GCC reçoit pour cette connexion une primitive d'indication MCS-Disconnect-Provider avant d'avoir envoyé une primitive de confirmation GCC-Conference-Invite positive, le fournisseur GCC générera immédiatement une primitive de confirmation GCC-Conference-Invite qu'il enverra au point de contrôle GCCSAP. Le

paramètre résultat de la PDU ainsi que le paramètre résultat renvoyé par la confirmation MCS-Connect-Provider (ou le paramètre "cause" de la primitive MCS-Disconnect-Provider) sont utilisés pour générer le résultat renvoyé par la primitive de confirmation GCC-Conference-Invite. Si le paramètre résultat de la confirmation est "utilisateur rejeté", le paramètre résultat de la PDU est utilisé pour déterminer le résultat renvoyé. Sinon le paramètre résultat de la confirmation MCS-Connect-Provider est utilisé directement.

Lorsqu'il reçoit une indication UserIDIndication, le nœud connecté directement au nœud invité comparera le paramètre "marque" avec sa liste de conférences pour lesquelles il attend l'envoi de cette PDU. Si une concordance est trouvée, le nœud mémorise l'identificateur du nœud origine (extrait de la primitive d'indication MCS-Send-Data) dans sa base de données contenant les identificateurs des nœuds qui lui sont directement subordonnés. La PDU sera ignorée si la marque ne coïncide avec aucune marque en attente.

Si la marque correspond à une invitation de conférence en attente, le fournisseur GCC générera lors de la réception de l'unité UserIDIndication une primitive de confirmation GCC-Conference-Invite qu'il enverra au point de contrôle GCCSAP avec un paramètre résultat positif.

La Figure 8-4 donne la succession d'événements pour une invitation dans une conférence qui réussit.



**Figure 8-4 – Invitation d'un nœud dans une conférence**

### 8.2.5 Demande d'addition d'un nœud dans une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Add (addition à une conférence) enverra au fournisseur GCC sommital une PDU ConferenceAddRequest (demande d'addition dans une conférence). Pour ce faire il émet une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC sommital, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceAddRequest est donné par le Tableau 8-18. Les contenus sont remplis à partir des paramètres fournis par la primitive de demande GCC-Conference-Add. Le paramètre "marque" est assigné par le fournisseur GCC demandeur d'une manière localement unique. Ce paramètre est utilisé pour identifier l'unité de PDU de réponse quand elle sera renvoyée.

**Tableau 8-18 – GCCPDU ConferenceAddRequest**

Contenu	Origine	Destination
Adresse réseau	Primitive de demande	Primitive d'indication
Nœud demandeur	Fournisseur GCC demandeur	Primitive d'indication
Marque	Fournisseur GCC demandeur	Fournisseur GCC destinataire
MCU chargée de l'addition (optionnel)	Primitive de demande	Fournisseur GCC sommital
Catégorie nodale (optionnel)	Primitive de demande	Primitive d'indication
Données utilisateur (optionnel)	Primitive de demande	Primitive d'indication
Adresse réseau V2 (voir l'Annexe B)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU ConferenceAddRequest le fournisseur GCC sommital vérifiera en premier lieu si le nœud demandeur possède le privilège requis pour ajouter un nœud, compte tenu du profil de conférence. Si ce n'est pas le cas il rejettera la demande en envoyant au nœud demandeur une unité PDU ConferenceAddResponse indiquant comme résultat "demandeur non valide". Le contenu de l'unité PDU ConferenceAddResponse est donné par le Tableau 8-19. Si le demandeur possède un privilège suffisant, le fournisseur GCC sommital examinera ensuite si le paramètre "unité MCU chargée de l'addition" est présent. Si ce n'est pas le cas ou si l'identificateur de MCU chargée de l'addition est identique à l'identificateur de nœud du fournisseur GCC sommital, le fournisseur GCC sommital, s'il prend en charge la fonction d'addition dans une conférence, produira une primitive d'indication GCC-Conference-Add qu'il enverra au point de contrôle GCCSAP. Les paramètres de la primitive sont remplis à partir du contenu de l'unité PDU reçue. Si le nœud ne prend pas en charge cette fonction, il peut rejeter la demande par l'envoi d'une unité PDU FunctionNotSupportedResponse (réponse de fonction non prise en charge) contenant l'unité ConferenceAddRequest reçue.

Si le paramètre facultatif "MCU chargée de l'addition" est présent et contient une valeur autre que l'identificateur de nœud du fournisseur GCC sommital, le fournisseur GCC sommital enverra une unité PDU ConferenceAddRequest à l'unité MCU chargée de l'addition. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de nœud de l'unité MCU chargée de l'addition, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU est identique à celui de l'unité PDU reçue. Lorsqu'il reçoit l'unité PDU ConferenceAddRequest et s'il prend en charge la fonction GCC-Conference-Add, un nœud qui n'est pas le fournisseur GCC sommital vérifiera en premier lieu que l'unité PDU a été reçue du fournisseur GCC sommital en examinant l'identificateur d'utilisateur de la primitive d'indication MCS-Send-Data reçue. Si l'unité PDU provient du fournisseur GCC sommital, il produira une primitive d'indication GCC-Conference-Add qu'il enverra au point de contrôle GCCSAP. Les paramètres de la primitive sont remplis à partir du contenu de l'unité PDU reçue. Si l'identificateur d'utilisateur de l'unité PDU reçue ne coïncide pas avec l'identificateur de nœud du fournisseur GCC sommital, l'unité PDU reçue est ignorée et aucune autre action n'est exécutée. Si le nœud ne prend pas en charge cette fonction, il peut rejeter la demande par l'envoi d'une unité PDU FunctionNotSupportedResponse contenant l'unité ConferenceAddRequest reçue dans la réponse.

Lorsque le fournisseur GCC reçoit une primitive de réponse GCC-Conference-Add, il enverra au nœud demandeur une PDU ConferenceAddResponse (réponse d'addition dans une conférence) par l'émission d'une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre l'identificateur du nœud demandeur, une priorité de données élevée et la PDU dans le champ de données. Dans le cas où la demande a été routée par le fournisseur GCC sommital vers un autre

nœud, la réponse est envoyée au nœud origine indiqué par le paramètre "nœud demandeur" de la PDU de demande. Le Tableau 8-19 donne le contenu de la PDU ConferenceAddResponse. Les contenus sont remplis à partir des paramètres fournis par la primitive de réponse GCC-Conference-Add. Le paramètre "marque" doit avoir la même valeur que le paramètre correspondant de la PDU ConferenceAddRequest.

**Tableau 8-19 – GCCPDU ConferenceAddResponse**

Contenu	Origine	Destination
Marque	Fournisseur GCC origine	Fournisseur GCC destinataire
Résultat	Primitive de réponse	Primitive de confirmation
Données utilisateur (optionnel)	Primitive de réponse	Primitive de confirmation

Lorsqu'il reçoit la PDU ConferenceAddResponse, le fournisseur GCC générera une primitive de confirmation GCC-Conference-Add qu'il enverra au point de contrôle GCCSAP. Les paramètres de la primitive sont obtenus à partir des paramètres de la PDU reçue.

### 8.2.6 Verrouillage d'une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Lock (verrouillage de conférence) enverra au fournisseur GCC sommital une PDU ConferenceLockRequest (demande de verrouillage de conférence). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC sommital, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceLockRequest est donné par le Tableau 8-20. Cette PDU n'a pas de paramètres.

**Tableau 8-20 – GCCPDU ConferenceLockRequest**

Contenu	Origine	Destination
-- Pas de paramètres --		

Lorsqu'il reçoit la PDU ConferenceLockRequest et s'il prend en charge la capacité de verrouillage de conférence, le fournisseur GCC sommital déterminera en premier lieu si le nœud demandeur a le privilège requis pour verrouiller la conférence, compte tenu des privilèges de verrouillage définis à la création de la conférence. Si tel est le cas, le fournisseur GCC sommital générera une primitive d'indication GCC-Conference-Lock qu'il enverra au point de contrôle GCCSAP. Le nœud origine indiqué dans la primitive est déterminé à partir de l'identificateur d'utilisateur du demandeur présent dans la primitive d'indication MCS-Send-Data. Lorsqu'il reçoit une primitive de réponse GCC-Conference-Lock, le fournisseur GCC enverra une PDU ConferenceLockResponse (réponse de verrouillage de conférence) par l'émission d'une primitive de demande MCS-Send-Data avec en paramètre le nœud origine, une priorité de données élevée et la PDU dans le champ de données. Le Tableau 8-21 donne le contenu de la PDU ConferenceLockResponse. Le paramètre résultat est généré à partir du résultat fourni par la primitive de réponse GCC-Conference-Lock.

Si le nœud demandeur n'a pas le privilège requis pour prendre en charge cette opération, l'indication GCC-Conference-Lock n'est pas générée et la demande est immédiatement rejetée par l'émission d'une PDU ConferenceLockResponse indiquant le résultat "demandeur non valide".

Le fournisseur GCC sommital préservera la correspondance entre l'ordre de réception des PDU ConferenceLockRequest et ConferenceUnlockRequest et l'ordre de transmission des primitives d'indication correspondantes vers le point de contrôle GCCSAP. Il préservera également la correspondance de l'ordre entre les primitives de réponse du point de contrôle GCCSAP et les PDU ConferenceLockResponse et ConferenceUnlockResponse.

Au niveau du nœud demandeur, l'ordre sera préservé entre les primitives de demande et les PDU correspondantes ConferenceLockRequest et ConferenceUnlockRequest transmises ainsi qu'entre les réponses ConferenceLockResponse et ConferenceUnlockResponse reçues et les primitives de confirmation correspondantes.

Si le fournisseur GCC sommital ne prend pas en charge la capacité de verrouillage de conférence, il générera dès la réception d'une PDU ConferenceLockRequest une PDU FunctionNotSupportedResponse (réponse de fonction non prise en charge) contenant la PDU ConferenceLockRequest.

**Tableau 8-21 – GCCPDU ConferenceLockResponse**

Contenu	Origine	Destination
Résultat	Primitive de réponse	Primitive de confirmation

Lorsqu'il reçoit la PDU ConferenceLockResponse, le fournisseur GCC générera une primitive de confirmation GCC-Conference-Lock qu'il enverra au point de contrôle GCCSAP. Le paramètre résultat de la primitive de confirmation est déduit du champ "résultat" de la PDU reçue.

Si le fournisseur GCC sommital envoie une réponse ConferenceLockResponse indiquant un résultat positif, il générera également une PDU ConferenceLockIndication (indication de verrouillage de conférence) qu'il enverra à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) indiquant l'identificateur de canal GCC-Broadcast-Channel (canal de diffusion), une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceLockIndication est donné par le Tableau 8-22. Cette PDU ne contient pas de paramètres.

**Tableau 8-22 – GCCPDU ConferenceLockIndication**

Contenu	Origine	Destination
-- Pas de paramètres --		

Lorsqu'il reçoit une PDU ConferenceLockIndication, un fournisseur GCC qui prend en charge cette indication optionnelle peut générer une primitive d'indication GCC-Conference-Lock-Report (compte rendu de verrouillage de conférence) et l'envoyer au point de contrôle GCCSAP. Il examinera auparavant l'identificateur d'utilisateur contenu dans la primitive d'indication MCS-Uniform-Send-Data reçue et le comparera à l'identificateur de nœud du fournisseur GCC sommital. La primitive GCC-Conference-Lock-Report ne peut être générée que si les identificateurs d'utilisateur reçus coïncident avec l'identificateur de nœud du fournisseur GCC sommital. Dans le cas contraire, la PDU reçue est ignorée et aucune autre action n'est exécutée.

Si le fournisseur GCC sommital apprend l'entrée d'un nouveau nœud dans une conférence en constatant sa présence dans le répertoire de conférence, alors que la conférence est verrouillée et qu'il est possible que la conférence ait été déverrouillée lorsque le nouveau nœud y est entré ou y a été invité, il générera une PDU ConferenceLockIndication (indication de verrouillage de conférence) qu'il enverra au nouveau nœud. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre l'identificateur nodal de ce nœud comme identificateur de canal, avec une priorité de données élevée et la PDU dans le champ de données. Comme autre possibilité, il peut envoyer cette PDU à tous les nœuds par l'émission d'une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) avec en paramètre l'identificateur du canal GCC-Broadcast-Channel comme identificateur de canal, une priorité de données élevée et la PDU dans le champ de données. Quoique l'indicateur signalant si la conférence est verrouillée ou non fasse partie de l'information fournie au nouveau nœud entrant pendant le processus d'établissement de la connexion, ceci garantit que si la conférence a quitté l'état verrouillé

depuis l'établissement de la connexion, ce nœud reçoit correctement la notification de ce changement. Le fournisseur GCC sommital n'a pas besoin d'envoyer cette PDU s'il est impossible que l'état de verrouillage de la conférence ait été changé depuis que le nœud est entré dans la conférence ou y a été invité, par exemple si l'état de verrouillage n'a pas changé depuis le début de la conférence.

### 8.2.7 Déverrouillage d'une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Unlock (déverrouillage de conférence) enverra au fournisseur GCC sommital une PDU ConferenceUnlockRequest (demande de déverrouillage de conférence). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC sommital, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceUnlockRequest est donné par le Tableau 8-23. Cette PDU n'a pas de paramètres.

**Tableau 8-23 – GCCPDU ConferenceUnlockRequest**

Contenu	Origine	Destination
-- Pas de paramètres --		

Lorsqu'il reçoit une PDU ConferenceUnlockRequest et s'il prend en charge la capacité de déverrouillage de conférence, le fournisseur GCC sommital déterminera en premier lieu si le nœud demandeur a le privilège requis pour déverrouiller la conférence compte tenu des privilèges de verrouillage/déverrouillage définis à la création de la conférence. Si tel est le cas, le fournisseur GCC sommital générera une primitive d'indication GCC-Conference-Unlock qu'il enverra au point de contrôle GCCSAP. Le nœud origine indiqué dans la primitive sera déterminé à partir de l'identificateur d'utilisateur du demandeur présent dans la primitive d'indication MCS-Send-Data. Lorsqu'il reçoit une primitive de réponse GCC-Conference-Unlock, le fournisseur GCC enverra une PDU ConferenceUnlockResponse (réponse de déverrouillage de conférence). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre comme identificateur de canal le nœud origine, une priorité de données élevée et la PDU dans le champ de données. Le Tableau 8-24 donne le contenu de la PDU ConferenceUnlockResponse. Le paramètre résultat est généré à partir du résultat fourni par la primitive de réponse GCC-Conference-Unlock.

Si le nœud demandeur n'a pas le privilège requis pour prendre en charge cette opération, l'indication GCC-Conference-Unlock n'est pas générée et la demande est immédiatement rejetée par l'émission d'une PDU ConferenceUnlockResponse indiquant le résultat "demandeur non valide".

Le fournisseur GCC sommital préservera la correspondance entre l'ordre de réception des PDU ConferenceLockRequest et ConferenceUnlockRequest et l'ordre de transmission des primitives d'indication correspondantes vers le point de contrôle GCCSAP. Il préservera également la correspondance de l'ordre entre les primitives de réponse du point de contrôle GCCSAP et les PDU ConferenceLockResponse et ConferenceUnlockResponse.

Au niveau du nœud demandeur, l'ordre sera préservé entre les primitives de demande et les PDU correspondantes ConferenceLockRequest et ConferenceUnlockRequest transmises ainsi qu'entre les réponses ConferenceLockResponse et ConferenceUnlockResponse reçues et les primitives de confirmation correspondantes.

Si le fournisseur GCC sommital ne prend pas en charge la capacité de déverrouillage de conférence, il générera dès la réception d'une PDU ConferenceUnlockRequest une PDU FunctionNotSupportedResponse (réponse de fonction non prise en charge) contenant la PDU ConferenceUnlockRequest.



**Tableau 8-24 – GCCPDU ConferenceUnlockResponse**

Contenu	Origine	Destination
Résultat	Primitive de réponse	Primitive de confirmation

Lorsqu'il reçoit la PDU ConferenceUnlockResponse, le fournisseur GCC générera une primitive de confirmation GCC-Conference-Unlock qu'il enverra au point de contrôle GCCSAP. Le paramètre résultat de la primitive de confirmation est déduit du champ "résultat" de la PDU reçue.

Si le fournisseur GCC sommital envoie une réponse ConferenceUnlockResponse indiquant un résultat positif, il générera également une PDU ConferenceUnlockIndication qu'il enverra à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) avec en paramètre le canal GCC-Broadcast-Channel comme identificateur de canal, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceUnlockIndication est donné par le Tableau 8-25. Cette PDU ne contient pas de paramètres.

**Tableau 8-25 – GCCPDU ConferenceUnlockIndication**

Contenu	Origine	Destination
-- Pas de paramètres --		

Lorsqu'il reçoit une PDU ConferenceUnlockIndication et qu'il prend en charge cette indication optionnelle, un fournisseur GCC peut générer une primitive d'indication GCC-Conference-Lock-Report et l'envoyer au point de contrôle GCCSAP. Il examinera auparavant l'identificateur d'utilisateur contenu dans la primitive d'indication MCS-Uniform-Send-Data reçue et le comparera à l'identificateur de nœud du fournisseur GCC sommital. La primitive GCC-Conference-Lock-Report ne peut être générée que si les identificateurs d'utilisateur reçus coïncident avec l'identificateur de nœud du fournisseur GCC sommital, sinon la PDU reçue est ignorée et aucune autre action n'est exécutée.

Si le fournisseur GCC sommital prend connaissance de l'entrée d'un nouveau nœud dans une conférence en constatant sa présence dans le répertoire de conférence, alors que la conférence est actuellement déverrouillée et qu'il est possible que la conférence ait été verrouillée lorsque le nouveau nœud y est entré ou y a été invité, le fournisseur GCC générera une PDU ConferenceUnlockIndication qu'il enverra au nouveau nœud par l'émission d'une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC de ce nœud comme identificateur de canal, une priorité de données élevée et la PDU dans le champ de données. Comme alternative, il peut envoyer cette PDU à tous les nœuds par l'émission d'une primitive de demande MCS-Uniform-Send-Data avec en paramètre l'identificateur du canal GCC-Broadcast-Channel (canal de diffusion), une priorité de données élevée et la PDU dans le champ de données. Quoique l'indicateur signalant si la conférence est verrouillée ou non fasse partie de l'information fournie au nouveau nœud entrant pendant le processus d'établissement de la connexion, ceci garantit que si la conférence a quitté l'état déverrouillé depuis l'établissement de la connexion, ce nœud reçoit correctement la notification de ce changement. S'il est impossible que l'état de verrouillage de la conférence ait été changé depuis que le nœud est entré dans la conférence ou y a été invité (par exemple si l'état de verrouillage n'a pas changé depuis le début de la conférence), le fournisseur GCC sommital n'a pas besoin d'envoyer cette PDU.

### 8.2.8 Déconnexion d'une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Disconnect (déconnexion d'une conférence) tentera en premier lieu d'éjecter les nœuds qui lui sont directement subordonnés dans la hiérarchie de connexion, si de tels nœuds existent. Il enverra à cet effet, pour

chacun de ces nœuds, une unité PDU ConferenceEjectUserIndication (indication d'éjection d'utilisateur d'une conférence) avec le paramètre "nœud à éjecter" contenant l'identificateur du nœud subordonné considéré et en donnant comme motif "déconnexion par un nœud supérieur". Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant dans son identificateur de canal à la fois le canal GCC-Broadcast-Channel (canal de diffusion prenant en charge les nœuds des anciens protocoles) et le canal GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données. Le contenu de l'unité PDU ConferenceEjectUserIndication est donné par le Tableau 8-31. Le fournisseur GCC attendra ensuite d'avoir reçu les indications MCS-Disconnect-Provider de chacune des connexions subordonnées. Il se déconnectera alors de la conférence en émettant d'abord une primitive de demande MCS-Detach-User (détachement d'utilisateur) suivie d'une primitive de demande MCS-Disconnect-Provider (déconnexion de fournisseur) émises dans la direction ascendante, qui est la seule connexion restante. Si pour une raison quelconque le fournisseur GCC n'a pas reçu les indications MCS-Disconnect-Provider de chacun des nœuds inférieurs dans un laps de temps raisonnable (déterminé localement), le fournisseur GCC peut entreprendre de rompre lui-même ces connexions par l'émission de demandes MCS-Disconnect-Provider envoyées sur chacune des connexions descendantes restantes suivies d'une primitive de demande MCS-Detach-User et d'une primitive de demande MCS-Disconnect-Provider émises dans la direction ascendante comme dans le cas normal. Dans les deux cas ainsi que dans le cas où il n'existerait pas de nœud subordonné, le fournisseur GCC produira ensuite une primitive de confirmation GCC-Conference-Disconnect qu'il enverra au point de contrôle GCCSAP. Le fournisseur GCC purgera ensuite la base de données de toute information associée à cette conférence.

Lorsqu'il reçoit une primitive d'indication MCS-Detach-User, tout fournisseur GCC de la conférence examinera l'identificateur d'utilisateur fourni dans cette indication et le comparera à sa liste d'identificateurs de nœuds contenue dans sa copie locale du répertoire de conférence. Si l'identificateur d'utilisateur correspond à l'un des identificateurs de nœuds et diffère de son propre identificateur de nœud, le fournisseur GCC générera une primitive d'indication GCC-Conference-Disconnect qu'il enverra au point de contrôle GCCSAP. Le paramètre de l'indication "nœud en cours de déconnexion" correspondra à l'identificateur d'utilisateur de l'indication reçue. Si le code motif de l'indication reçue est "demandé par l'utilisateur", le code motif dans la primitive d'indication GCC-Conference-Disconnect sera soit "demandé par l'utilisateur" ou "nœud éjecté" selon qu'une PDU ConferenceEjectUserIndication, contenant dans son champ "nœud à éjecter" le même identificateur d'utilisateur que la primitive d'indication MCS-Detach-User, a ou n'a pas été reçue précédemment. Sinon le motif donné est "inconnu".

### 8.2.9 Fin de conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Terminate (fin de conférence) enverra au fournisseur GCC sommital une PDU ConferenceTerminateRequest (demande de fin de conférence) par l'émission d'une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC sommital, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceTerminateRequest est donné par le Tableau 8-26. Le code "cause" est déduit du paramètre correspondant de la primitive de demande.

**Tableau 8-26 – GCCPDU ConferenceTerminateRequest**

Contenu	Origine	Destination
Cause	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit la PDU ConferenceTerminateRequest, le fournisseur GCC sommital déterminera en premier lieu si le nœud demandeur a le privilège requis pour terminer la conférence, compte tenu

des privilèges de fin définis à la création de la conférence. Si ce n'est pas le cas, la demande est rejetée et une réponse `ConferenceTerminateResponse` (réponse de fin de conférence) est renvoyée par l'émission d'une primitive de demande `MCS-Send-Data` (envoi de données) avec en paramètre l'identificateur de canal correspondant à l'identificateur d'utilisateur du demandeur, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU `ConferenceTerminateResponse` est donné par le Tableau 8-27. Dans ce cas le paramètre résultat reçoit comme motif du rejet "demandeur non valide".

Si le demandeur possède le privilège requis pour terminer la conférence, une réponse `ConferenceTerminateResponse` indiquant un résultat positif sera alors renvoyée au demandeur. En outre une indication `ConferenceTerminateIndication` (indication de fin de conférence) est envoyée à tous les nœuds de la conférence par l'émission d'une primitive de demande `MCS-Uniform-Send-Data` (envoi uniforme de données) spécifiant dans son identificateur de canal à la fois le canal `GCC-Broadcast-Channel` (canal de diffusion prenant en charge les nœuds des anciens protocoles) et le canal `GCC-Conventional-Broadcast-Channel`, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données. Le contenu de l'unité PDU `ConferenceTerminateIndication` (indication de fin de conférence) est donné par le Tableau 8-28. Le code "motif" est déduit du code "motif" de l'unité PDU de demande.

**Tableau 8-27 – GCCPDU ConferenceTerminateResponse**

Contenu	Origine	Destination
Résultat	Fournisseur GCC sommital	Primitive de confirmation

**Tableau 8-28 – GCCPDU ConferenceTerminateIndication**

Contenu	Origine	Destination
Résultat	Fournisseur GCC sommital	Primitive de confirmation

Lorsqu'il reçoit la PDU `ConferenceTerminateResponse`, le fournisseur GCC générera une primitive de confirmation `GCC-Conference-Terminate` qu'il enverra au point de contrôle `GCCSAP`. Le résultat indiqué dans la primitive est obtenu directement à partir du paramètre résultat de la PDU.

Lorsqu'il reçoit la PDU `ConferenceTerminateIndication` envoyée par le fournisseur GCC sommital, le fournisseur GCC attendra d'abord d'avoir reçu les indications `MCS-Disconnect-Provider` (déconnexion de fournisseur) sur chaque connexion qui lui est directement subordonnée. Il se déconnectera ensuite de la conférence par l'émission d'une primitive de demande `MCS-Disconnect-Provider` dirigée dans la direction ascendante qui est la seule connexion restante. Si pour une raison quelconque le fournisseur GCC n'a pas reçu les indications `MCS-Disconnect-Provider` de chacun des nœuds inférieurs dans un laps de temps raisonnable (déterminé localement), le fournisseur GCC peut entreprendre de rompre lui-même ces connexions par l'émission de demandes `MCS-Disconnect-Provider` envoyées sur chacune des connexions descendantes restantes suivies d'une primitive de demande `MCS-Disconnect-Provider` émise dans la direction ascendante comme dans le cas normal. Dans les deux cas ainsi que dans le cas où il n'existerait pas de nœud subordonné, le fournisseur GCC générera ensuite une primitive d'indication `GCC-Conference-Terminate` qu'il enverra au point de contrôle `GCCSAP`. La cause indiquée dans cette primitive provient directement du paramètre "cause" de la PDU.

Si le fournisseur GCC reçoit la primitive d'indication `MCS-Disconnect-Provider` du service MCS local correspondant à sa connexion MCS ascendante, ce qui indique que la connexion MCS a été interrompue à la suite d'une condition anormale survenue au sein du service MCS, le fournisseur GCC générera une primitive d'indication `GCC-Conference-Terminate` qu'il enverra au point de contrôle `GCCSAP`. Le code cause indiquera une fin anormale.

Si le fournisseur GCC sommital reçoit une indication de déconnexion [par une PDU ou par une primitive d'indication MCS-Detach-User (déconnexion d'utilisateur)] dont le résultat est tel que le nœud local reste le seul nœud listé dans le répertoire de conférence, le fournisseur GCC vérifiera le profil de conférence pour déterminer si la conférence se termine manuellement ou automatiquement. Si la fin est manuelle, aucune autre action n'est exécutée. Si la fin est automatique, le fournisseur GCC indiquera au nœud local que la conférence a été terminée en générant une primitive d'indication GCC-Conference-Terminate et l'émettra vers le point de contrôle GCCSAP. Le code cause indiquera qu'il n'y a plus de nœuds dans une conférence se terminant automatiquement. Le fournisseur GCC purgera ensuite la base de données de toute information associée à cette conférence.

### 8.2.10 Ejection d'un nœud d'une conférence

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Eject-User (éjection d'un utilisateur d'une conférence) comparera en premier lieu le paramètre "nœud à éjecter" avec les identificateurs des nœuds qui lui sont directement subordonnés, s'il en existe. Si le nœud à éjecter n'est pas directement subordonné, le fournisseur GCC enverra une unité PDU ConferenceEjectUserRequest (demande d'éjection d'un utilisateur d'une conférence) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConferenceEjectUserRequest est donné par le Tableau 8-29. Les paramètres de l'unité PDU sont obtenus à partir de la primitive de demande.

Si le nœud à éjecter est un nœud situé directement subordonné du nœud local, le fournisseur GCC enverra par contre une unité PDU ConferenceEjectUserIndication (indication d'éjection d'un utilisateur d'une conférence) à tous les nœuds en indiquant l'identificateur du nœud à éjecter dans l'unité PDU et le motif tel qu'il est indiqué dans la primitive de demande. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Data spécifiant dans son identificateur de canal à la fois le canal GCC-Broadcast-Channel (canal de diffusion prenant en charge les nœuds des anciens protocoles) et le canal GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données. Le contenu de l'unité PDU ConferenceEjectUserIndication est donné par le Tableau 8-31. Le fournisseur GCC peut ensuite attendre d'avoir reçu une primitive d'indication MCS-Disconnect-Provider (déconnexion de fournisseur) de la connexion correspondant au nœud éjecté. Si pour une raison quelconque, le fournisseur GCC n'a pas reçu de primitive d'indication MCS-Disconnect-Provider du nœud éjecté dans un laps de temps raisonnable (déterminé localement), le fournisseur GCC peut entreprendre de rompre lui-même cette connexion par l'émission d'une primitive de demande MCS-Disconnect-Provider vers le nœud éjecté. Le fournisseur GCC produira ensuite une unité PDU de confirmation GCC-Conference-Eject-User indiquant un résultat positif.

**Tableau 8-29 – GCCPDU ConferenceEjectUserRequest**

Contenu	Origine	Destination
Nœud à éjecter	Primitive de demande	Fournisseur GCC sommital
Cause	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit l'unité PDU ConferenceEjectUserRequest, le fournisseur GCC sommital déterminera en premier lieu si le nœud demandeur a le privilège requis pour éjecter un utilisateur, compte tenu des privilèges d'éjection d'utilisateur définis à la création de la conférence. Si ce n'est pas le cas, la demande est rejetée et une unité PDU ConferenceEjectUserResponse (réponse d'éjection d'un utilisateur d'une conférence) est renvoyée au demandeur par l'émission d'une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le

champ de données. Le contenu de l'unité PDU ConferenceEjectUserResponse est donné par le Tableau 8-30. Dans ce cas le paramètre résultat reçoit comme motif du rejet "demandeur non valide".

Si le demandeur possède le privilège requis pour éjecter un utilisateur, une primitive ConferenceEjectUserIndication sera alors diffusée à tous les nœuds de la conférence par l'émission d'une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant dans son identificateur de canal à la fois le canal GCC-Broadcast-Channel (canal de diffusion prenant en charge les nœuds des anciens protocoles) et le canal GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données. Le contenu de l'unité PDU ConferenceEjectUserIndication est donné par le Tableau 8-31. Le paramètre "motif" de cette PDU est déduit de l'unité PDU de demande. Si le nœud à éjecter est valide, une réponse est renvoyée au demandeur indiquant un résultat positif. S'il n'est pas possible d'éjecter le nœud demandé, une réponse contenant un résultat négatif est renvoyée.

**Tableau 8-30 – GCCPDU ConferenceEjectUserResponse**

Contenu	Origine	Destination
Nœud à éjecter	Fournisseur GCC sommital	Primitive de confirmation
Résultat	Fournisseur GCC sommital	Primitive de confirmation

Lorsqu'il reçoit une unité PDU ConferenceEjectUserResponse, un fournisseur GCC produira une primitive de confirmation GCC-Conference-Eject-User qu'il enverra au point de contrôle GCCSAP. Le contenu de la primitive de confirmation sera déduit de l'unité PDU ConferenceEjectUserResponse.

**Tableau 8-31 – GCCPDU ConferenceEjectUserIndication**

Contenu	Origine	Destination
Nœud à éjecter	Fournisseur GCC sommital	Primitive de confirmation
Cause	Fournisseur GCC sommital	Primitive d'indication

Lorsqu'il reçoit l'unité PDU ConferenceEjectUserIndication, le fournisseur GCC comparera le paramètre "nœud à éjecter" avec son propre identificateur de nœud. En cas d'égalité, il comparera ensuite l'identificateur d'utilisateur donné dans la primitive d'indication MCS-Send-Data (envoi de données) avec l'identificateur de nœud du fournisseur GCC sommital et avec celui du nœud qui est son supérieur direct dans la hiérarchie de connexion. Si l'identificateur du nœud origine est égal à l'un de ceux-ci, il se déconnectera immédiatement de la conférence en essayant en premier lieu d'éjecter les nœuds qui lui sont directement subordonnés, s'ils existent. Il émettra à cet effet, pour chacun de ces nœuds, une unité PDU ConferenceEjectUserIndication avec le paramètre "nœud à éjecter" contenant l'identificateur du nœud subordonné donné et en donnant comme cause "éjection par un nœud supérieur". Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data spécifiant dans son identificateur de canal à la fois le canal GCC-Broadcast-Channel (canal de diffusion prenant en charge les nœuds des anciens protocoles) et le canal GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données. Le fournisseur GCC attendra ensuite d'avoir reçu les indications MCS-Disconnect-Provider de chaque connexion subordonnée. Il se déconnectera ensuite de la conférence par l'émission d'une primitive de demande MCS-Detach-User suivie d'une primitive de demande MCS-Disconnect-Provider dans la direction ascendante, qui est la seule connexion restante. Si pour une raison quelconque, le fournisseur GCC n'a pas reçu les indications MCS-Disconnect-Provider de chacun des nœuds inférieurs dans un laps de temps raisonnable (déterminé localement), le fournisseur GCC peut entreprendre de rompre lui-même ces

connexions par l'émission de demandes MCS-Disconnect-Provider dirigées vers chaque connexion inférieure restante, suivies d'abord d'une demande MS-Detach-User puis d'une demande MCS-Disconnect-Provider dirigée vers la connexion supérieure comme dans le cas normal. Dans les deux cas ainsi que dans le cas où il n'existerait pas de nœud subordonné, le fournisseur GCC générera ensuite une primitive d'indication GCC-Conference-Eject-User qu'il enverra au point de contrôle GCCSAP. Si la PDU reçue contient un identificateur d'utilisateur qui ne coïncide pas avec celui du fournisseur GCC sommital ou celui du nœud situé immédiatement au-dessus dans la hiérarchie de connexion, la PDU sera ignorée et aucune autre action ne sera exécutée.

Si le fournisseur GCC destinataire est une MCU supérieure directe du nœud, il peut d'une manière optionnelle déconnecter le nœud à éjecter de la conférence par l'émission d'une primitive de demande MCS-Disconnect-Provider pour la connexion MCS correspondante. Avant de prendre une telle mesure, il vérifiera que l'identificateur d'utilisateur indiqué par la PDU reçue est le même que l'identificateur de nœud du fournisseur GCC sommital.

Lors de la réception d'une PDU ConferenceEjectUserIndication envoyée soit par le fournisseur GCC sommital, soit par l'identificateur de nœud figurant dans le registre de conférence au-dessus du nœud à éjecter, tous les nœuds à l'exception du nœud à éjecter feront une notice concernant cet événement qu'ils consulteront ultérieurement en cas de réception d'une primitive d'indication MCS-Detach-User concernant le nœud à éjecter. A ce moment, comme spécifié au § 8.2.8, un nœud générera une primitive d'indication GCC-Conference-Disconnect (déconnexion d'une conférence) et l'enverra au point de contrôle GCCSAP. La cause indiquée sera "nœud éjecté".

### 8.2.11 Transfert de nœuds entre conférences

Un fournisseur GCC qui reçoit une primitive de demande GCC-Conference-Transfer (transfert de conférence) enverra au fournisseur GCC sommital une PDU ConferenceTransferRequest (demande de transfert de conférence). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre le canal de l'identificateur de nœud du fournisseur GCC sommital, une priorité de données élevée et la PDU dans le champ de données. Le contenu de la PDU ConferenceTransferRequest est donné par le Tableau 8-32. Tous les paramètres de cette PDU sont obtenus directement à partir de la primitive de requête.

**Tableau 8-32 – GCCPDU ConferenceTransferRequest**

Contenu	Origine	Destination
Nom de conférence	Primitive de demande	Fournisseur GCC sommital
Modificateur de nom de conférence (optionnel)	Primitive de demande	Fournisseur GCC sommital
Adresse réseau (optionnel)	Primitive de demande	Fournisseur GCC sommital
Nœuds chargés du transfert (optionnel)	Primitive de demande	Fournisseur GCC sommital
Mot de passe (optionnel)	Primitive de demande	Fournisseur GCC sommital
Adresse réseau V2 (voir l'Annexe B)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit l'unité PDU ConferenceTransferRequest, le fournisseur GCC sommital déterminera en premier lieu si le nœud demandeur a le privilège requis pour demander un transfert, compte tenu des privilèges de transfert définis à la création de la conférence. Si ce n'est pas le cas, la demande est rejetée et une réponse ConferenceTransferResponse est renvoyée au demandeur. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant dans son identificateur le canal Node-ID-Channel du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données de l'unité PDU. Le contenu de l'unité

PDU ConferenceTransferResponse est donné par le Tableau 8-33. Dans ce cas, le paramètre *Résultat* reçoit comme motif du rejet "demandeur non valide".

Si le demandeur possède le privilège requis pour demander un transfert, alors le fournisseur GCC sommital renverra au demandeur une unité PDU ConferenceTransferResponse, comme décrit ci-dessus, mais avec un paramètre résultat indiquant une réussite. Il diffusera ensuite à tous les nœuds de la conférence une unité PDU ConferenceTransferIndication (indication de transfert de conférence) par l'émission d'une primitive de demande MCS-Uniform-Send-Data spécifiant dans son identificateur de canal à la fois le canal GCC-Broadcast-Channel (canal de diffusion prenant en charge les nœuds des anciens protocoles) et le canal GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU sélectionnée dans le champ de données. Le contenu de l'unité PDU ConferenceTransferIndication est donné par le Tableau 8-34. Les paramètres de cette PDU sont obtenus à partir de l'unité PDU ConferenceTransferRequest reçue.

**Tableau 8-33 – GCCPDU ConferenceTransferResponse**

Contenu	Origine	Destination
Nom de conférence	Fournisseur GCC sommital	Primitive de confirmation
Modificateur de nom de conférence (optionnel)	Fournisseur GCC sommital	Primitive de confirmation
Nœuds chargés du transfert (optionnel)	Fournisseur GCC sommital	Primitive de confirmation
Résultat	Fournisseur GCC sommital	Primitive de confirmation

Lorsqu'il reçoit la PDU ConferenceTransferResponse, le fournisseur GCC générera une primitive de confirmation GCC-Conference-Transfer qu'il enverra au point de contrôle GCCSAP. Le contenu de la primitive est déduit de la PDU ConferenceTransferResponse.

**Tableau 8-34 – GCCPDU ConferenceTransferIndication**

Contenu	Origine	Destination
Nom de conférence	Fournisseur GCC sommital	Primitive d'indication
Modificateur de nom de conférence (optionnel)	Fournisseur GCC sommital	Primitive d'indication
Adresse réseau (optionnel)	Fournisseur GCC sommital	Primitive d'indication
Nœuds chargés du transfert (optionnel)	Fournisseur GCC sommital	Primitive d'indication
Mot de passe (optionnel)	Fournisseur GCC sommital	Primitive d'indication
Adresse réseau V2 (voir l'Annexe B)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit la PDU ConferenceTransferIndication, le fournisseur GCC qui prend en charge la primitive d'indication GCC-Conference-Transfer vérifiera la liste de nœuds de destination. Si l'identificateur du nœud local se trouve dans cette liste ou si la liste de nœuds de destination est NULL (néant), le fournisseur GCC vérifiera ensuite que l'identificateur d'utilisateur donné dans la primitive d'indication MCS-Uniform-Send-Data (envoi uniforme de données) coïncide avec l'identificateur de nœud du fournisseur GCC sommital. Si tel est le cas, il générera une primitive d'indication GCC-Conference-Transfer qu'il enverra au point de contrôle GCCSAP. Si le fournisseur GCC sommital ne prend pas en charge la primitive d'indication GCC-Conference-Transfer, si le nœud local ne figure pas sur la liste de nœuds de destination ou si la PDU reçue ne

provient pas du fournisseur GCC sommital, la PDU est ignorée et aucune autre action n'est exécutée.

### **8.3 Les répertoires d'application et de conférence**

#### **8.3.1 Protocole de répertoire original ou protocole de répertoire échelonné**

A un moment donné, l'échange d'informations de répertoire était le plus grand obstacle aux conférences échelonnées dans le cadre de la présente Recommandation. En raison des capacités ajoutées au protocole GCC pour prendre en charge les conférences échelonnées, à savoir les catégories nodales, le canal GCC-Conventional-Broadcast et le canal GCC-Counted-Broadcast, le protocole associé à l'échange de répertoires de conférence et de répertoires d'application a subi des modifications importantes. Pour assurer la rétrocompatibilité avec les nœuds appliquant les anciens protocoles, il est nécessaire de décrire dans des paragraphes distincts le protocole de répertoire original et le protocole de répertoire échelonné. Il importe que tous les nœuds T.124 prennent en charge aussi bien le protocole de répertoire original que le protocole de répertoire échelonné afin de garantir la rétrocompatibilité. On notera que la description du protocole de répertoire original donne toutes les informations sur le protocole de répertoire nécessaire pour prendre en charge les nœuds successoraux. Toutes les informations décrivant le protocole de répertoire échelonné sont contenues dans le sous-paragraphes qui suit la description du protocole de répertoire original (voir § 8.3.3).

#### **8.3.2 Protocole de répertoire original**

Le présent sous-paragraphes décrit en détail tous les aspects du protocole de répertoire original. Celui-ci n'est pas considéré comme étant échelonné en raison de l'application des mises à jour complètes de répertoire, qui sont propagées vers tous les nœuds d'une conférence chaque fois qu'un nœud joint, quitte ou modifie un enregistrement de répertoire. Les nœuds qui tentent de prendre en charge des nœuds successoraux doivent joindre le canal GCC-Broadcast-Channel original et mettre en œuvre le protocole détaillé ci-dessous. On notera que les nœuds qui ne prennent en charge que le protocole de répertoire original ne sont pas informés des catégories nodales et ne prennent donc en charge que les nœuds conventionnels.

##### **8.3.2.1 Aperçu général**

Les répertoires d'application et de conférence sont communiqués entre nœuds en utilisant le même ensemble de PDU. La PDU RosterUpdateIndication (indication de mise à jour de répertoire) est impliquée dans tous les aspects de cet échange. Elle est utilisée pour envoyer aux autres nœuds de la conférence des informations de répertoire partielles ou complètes.

Un nœud qui n'est pas le nœud sommital annonce initialement son information de répertoire lors de son entrée dans une conférence en envoyant une PDU RosterUpdateIndication au nœud qui est son supérieur immédiat dans la hiérarchie de connexion. Les mises à jour ultérieures d'une partie quelconque du répertoire d'application ou du répertoire de conférence sont annoncées par l'émission d'une PDU RosterUpdateIndication contenant la nouvelle information.

Lorsqu'un nœud met à jour sa partie de répertoire, ou l'annonce pour la première fois, cette information est propagée de nœud en nœud vers le haut de la hiérarchie jusqu'à ce qu'elle atteigne le fournisseur GCC sommital responsable de la constitution des répertoires complets d'application et de conférence ainsi que de leur distribution à l'ensemble des nœuds de la conférence.

Lorsque l'information de répertoire se propage vers le haut de la hiérarchie de connexion, chaque nœud intermédiaire est responsable de la constitution d'un sous-ensemble des répertoires complets d'application et de conférence. Ce sous-ensemble contient ce nœud ainsi que tous les nœuds qui lui sont subordonnés. Ceci signifie que le répertoire de conférence contient les enregistrements de nœud de tous ces nœuds, les entrées du répertoire d'application concernant chaque session de protocole d'application contiennent les enregistrements d'application de chaque nœud qui contient



des entités de protocole d'application homologues inscrites; la liste des capacités d'application pour chaque session de protocole d'application contient l'information condensée des capacités pour l'ensemble de tels nœuds. Lorsqu'il reçoit une indication RosterUpdateIndication d'un nœud subordonné, un nœud intermédiaire procède aux modifications appropriées sur son sous-ensemble de répertoires puis transmet cette information au nœud immédiatement supérieur de la hiérarchie de connexion par l'émission d'une PDU RosterUpdateIndication.

Si la mise à jour en provenance d'un nœud contient une modification qui a pour conséquence que la liste des capacités d'application d'au moins une des sessions de protocole d'application doit être recalculée, la nouvelle information du sous-ensemble ne peut être générée que si chacune des listes de capacités d'application de chaque nœud subordonné direct est connue individuellement, de manière à permettre une nouvelle application des règles de condensation des capacités à l'information mise à jour. Le fournisseur GCC de chaque nœud gèrera cette information au niveau local.

Lorsque les mises à jour ont atteint le fournisseur GCC sommital, celui-ci transmettra les nouvelles informations de répertoire à l'ensemble des nœuds de la conférence en diffusant une PDU RosterUpdateIndication à tous les nœuds.

L'envoi d'une information de répertoire au moyen d'une PDU RosterUpdateIndication peut se faire de trois manières. Le répertoire peut être envoyé sous la forme d'un rafraîchissement complet de la totalité des répertoires de conférence et d'application qui remplace toutes les entrées des répertoires. En second lieu, des parties de répertoires peuvent être envoyées, auquel cas le répertoire de conférence et/ou des parties du répertoire d'application associées à une ou plusieurs clés de session peuvent être rafraîchis. Dans ce cas, toutes les entrées des parties de répertoires transmises sont mises à jour, mais les parties non transmises restent inchangées. Finalement, des changements de parties du répertoire peuvent être transmis sous forme de mises à jour, auquel cas sont transmises uniquement les entrées particulières du répertoire (par exemple les enregistrements d'application individuels) qui ont été ajoutées, modifiées ou supprimées, toutes les autres entrées restant inchangées. Le fournisseur GCC sommital n'utilise que l'une des deux premières méthodes lorsqu'il diffuse une nouvelle information de répertoire. Lorsque le fournisseur GCC sommital diffuse une nouvelle information de répertoire à la suite d'un changement occasionné par l'entrée d'au moins un nœud dans la conférence, il diffusera cette information uniquement sous la forme d'un rafraîchissement complet, parce que le ou les nouveaux nœuds ne disposent pas d'une information de répertoire préexistante. Lors de la propagation de l'information vers le haut de la hiérarchie de connexion, l'une quelconque des trois méthodes peut être utilisée. Le choix de la méthode dépend de la taille de l'information nouvelle et sera fait de préférence dans le but de minimiser la taille des PDU et les durées de transmission.

NOTE – Le procédé décrit nécessite que tous les fournisseurs GCC d'une conférence mémorisent le sous-ensemble des répertoires de conférence et d'application aussi bien pour leur propre nœud que pour l'ensemble des nœuds qui leur sont directement subordonnés. En outre, diverses fonctions exigent que les fournisseurs GCC stockent également la totalité des répertoires de conférence et d'application diffusés par le fournisseur GCC sommital. Si pour une raison quelconque un fournisseur GCC ne peut pas stocker tout ou partie de cette information, le seul remède consiste à le déconnecter de la conférence et, si possible, à revenir dans la conférence à un moment ultérieur.

### **8.3.2.2 Entrée de nœuds dans une conférence**

Lorsqu'un nœud entre dans une conférence, soit par création de la conférence, par entrée ou sur invitation, le fournisseur GCC émettra une primitive d'indication GCC-Application-Permission-To-Enroll (autorisation d'inscription d'application) à destination du point d'accès GCCSAP de toutes les entités de protocole d'application qui ont indiqué localement leur présence au fournisseur GCC. Quand un fournisseur GCC prend connaissance d'une nouvelle entité de protocole d'application alors qu'il fait déjà partie d'une conférence, il émettra une primitive

d'indication GCC-Application-Permission-To-Enroll à destination du point d'accès GCCSAP correspondant pour indiquer l'existence de cette conférence.

Avant de prendre toute nouvelle action, le fournisseur GCC attendra qu'il ait reçu une primitive de demande GCC-Conference-Announce-Presence (annonce de présence de conférence) du point de contrôle GCCSAP ainsi qu'une primitive de demande GCC-Application-Enroll (inscription d'application) venant de chaque entité de protocole d'application qui a reçu une indication GCC-Application-Permission-To-Enroll. Certaines de ces demandes peuvent indiquer que l'entité de protocole d'application n'a pas l'intention de s'inscrire (l'indicateur d'inscription n'est pas positionné). Le répertoire d'application local est mis à jour à chaque fois qu'une primitive de demande GCC-Application-Permission-To-Enroll est reçue avec un indicateur d'inscription positionné.

Le fournisseur GCC attribuera également d'une manière locale un identificateur d'entité à chaque entité de protocole d'application inscrite. Cette information sera utilisée par le fournisseur GCC comme l'identificateur de l'entité de protocole d'application dans la PDU RosterUpdateIndication (indication de mise à jour de répertoire). L'identificateur d'entité de protocole d'application attribué et l'identificateur du nœud local figurent dans la primitive de confirmation GCC-Application-Enroll renvoyée à chaque entité de protocole d'application qui s'inscrit. L'identificateur est une valeur entière de 16 bits qui est non ambiguë pour toutes les entités de protocole d'application inscrites sur un nœud. Si une entité de protocole d'application résilie son inscription, la valeur de son identificateur de nœud ne sera pas réutilisée avant que toutes les autres valeurs non attribuées n'aient été utilisées.

Le fournisseur GCC examinera l'indicateur de capacité de présidence d'opération de chaque entité de protocole d'application inscrite dont l'indicateur d'activité est positionné. Les entités de protocole d'application inactives sont supposées ne pas posséder cette capacité. S'il existe plus d'une entité de protocole d'application possédant la capacité de présidence, le fournisseur GCC choisira l'une d'elles pour laquelle il positionnera l'indicateur dans l'information de répertoire d'application transmise. La règle de choix de l'entité de protocole d'application est un problème local non spécifié par la présente Recommandation. Une règle normale pourrait consister à choisir la première d'une telle entité de protocole d'application qui s'inscrit.

Lorsqu'il reçoit la primitive de demande GCC-Conference-Announce-Presence émise par le point de contrôle GCCSAP et une primitive de demande GCC-Application-Enroll émise par chacune des entités de protocole d'application, un fournisseur GCC qui n'est pas le fournisseur GCC sommital enverra une unité RosterUpdateIndication au fournisseur GCC immédiatement supérieur. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre l'identificateur du nœud destinataire, une priorité de données élevée et la PDU dans le champ de données. L'identificateur du nœud immédiatement supérieur est fourni au moment de la connexion à la conférence par les PDU ConferenceCreateResponse (réponse de création de conférence), ConferenceJoinResponse (réponse d'entrée en conférence) ou ConferenceInviteRequest (demande d'invitation dans une conférence) selon la façon dont s'est faite l'entrée dans la conférence. Cette valeur sera également présente comme paramètre "nœud supérieur" dans la partie répertoire de conférence de la PDU. Le Tableau 8-35 donne le contenu de la PDU RosterUpdateIndication. Dans ce cas, le répertoire est envoyé sous la forme d'un rafraîchissement complet comportant l'enregistrement local de nœud, les enregistrements d'application de toutes les entités de protocole d'application inscrites et les listes de capacités d'application pour toutes les sessions de protocole d'application correspondant aux entités de protocole d'application inscrites. S'il y a plus d'une entité de protocole d'application inscrite localement pour une session de protocole d'application donnée, le fournisseur GCC procédera à une condensation des listes de capacités d'application entre entités de protocole d'application homologues afin de produire la liste des capacités d'application fournie pour cette session dans la PDU RosterUpdateIndication. La condensation sera faite selon la procédure décrite au § 8.3.2.8. Le § 8.3.2.5 décrit l'action exécutée par un nœud qui reçoit une unité RosterUpdateIndication.

Un nœud fournisseur GCC sommital qui vient d'entrer dans une conférence et qui reçoit une primitive de demande GCC-Conference-Announce-Presence utilisera l'information contenue dans sa base de données de répertoire de conférence. Il peut toutefois attendre d'avoir reçu une PDU RosterUpdateIndication d'au moins un des nœuds de la conférence avant de transmettre à son tour une quelconque PDU. Lorsqu'il reçoit une telle indication, il mettra à jour sa base de données de répertoires de conférence et d'application puis diffusera la totalité des répertoires d'application et de conférence en envoyant une PDU RosterUpdateIndication à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) avec en paramètre l'identificateur de canal GCC-Broadcast-Channel (canal de diffusion), une priorité de données élevée et la PDU dans le champ de données. Si de nouveaux nœuds sont entrés dans la conférence, l'unité RosterUpdateIndication est envoyée sous la forme d'un rafraîchissement complet comportant l'enregistrement de nœud de tous les nœuds de la conférence ainsi que, pour chaque session de protocole d'application de la conférence, une liste des capacités d'application condensée, une liste des entités de protocole d'application homologues participant à la session et l'enregistrement d'application de chacune d'elles. Les règles utilisées pour générer les répertoires d'application et de conférence complets, y compris la liste des capacités d'application condensée, sont décrites au § 8.3.2.5.

**Tableau 8-35 – GCCPDU RosterUpdateIndication**

Contenu	Origine	Destination
Indicateur de mise à jour complète	Fournisseur GCC origine	Fournisseur GCC destinataire
Information de conférence (optionnel)	Fournisseur GCC origine indiqué dans la demande ou PDU reçue	Fournisseur GCC destinataire
Information d'application (optionnel)	Fournisseur GCC origine indiqué dans la demande ou PDU reçue	Fournisseur GCC destinataire

### 8.3.2.3 Inscription des entités de protocole d'application

Lorsqu'il reçoit une primitive de demande GCC-Application-Enroll (inscription d'application), un fournisseur GCC déterminera en premier lieu s'il existe déjà dans le répertoire d'application local une entrée pour la conférence mentionnée qui contient le point d'accès GCCSAP du demandeur. Si ce n'est pas le cas, et si l'indicateur d'inscription de la demande est positionné sur inscription, une nouvelle entrée est créée dans le répertoire d'application local avec l'information contenue dans la primitive de demande. Dans ce cas, le fournisseur GCC attribuera également un identificateur d'entité à l'entité de protocole d'application nouvellement inscrite, comme décrit au § 8.3.2.2. Si l'indicateur d'inscription est positionné sur résiliation, une primitive de confirmation GCC-Application-Enroll est immédiatement générée et envoyée comme confirmation au point d'accès GCCSAP du demandeur. Aucune autre action n'est exécutée dans ce cas.

Si l'entrée existe déjà dans le répertoire d'application local et si l'indicateur d'inscription de la demande est positionné sur inscription, le contenu de l'entrée existante est mis à jour pour tenir compte des nouvelles valeurs fournies. Si l'indicateur d'inscription de la demande est positionné sur résiliation, le fournisseur GCC supprimera l'entrée correspondante du répertoire d'application local.

Si le fournisseur GCC n'est pas le fournisseur GCC sommital de la conférence mentionnée, il enverra ensuite son nouveau répertoire d'application local au nœud immédiatement supérieur au moyen d'une PDU RosterUpdateIndication (indication de mise à jour de répertoire). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) avec en paramètre l'identificateur du nœud supérieur comme identificateur de canal, une priorité de données élevée et la PDU sélectionnée dans le champ de données. L'identificateur du nœud immédiatement supérieur est connu au moment de la connexion à la conférence à partir des PDU ConferenceCreateResponse (réponse de création de conférence), ConferenceJoinResponse (réponse d'entrée en conférence) ou

ConferenceInviteRequest (demande d'invitation dans une conférence) selon la façon dont s'est faite l'entrée dans la conférence. Le Tableau 8-35 donne le contenu de la PDU RosterUpdateIndication.

Si l'inscription s'est faite avant la transmission initiale d'une PDU RosterUpdateIndication, l'enregistrement nouveau ou modifié est transmis sous la forme d'un rafraîchissement complet comme décrit au § 8.3.2.2. Si une entité de protocole d'application a résilié son inscription avant la transmission initiale, son entrée dans le répertoire est supprimée et ne figurera plus dans les PDU transmises par la suite.

Si l'inscription s'est faite après la transmission initiale d'une PDU RosterUpdateIndication, l'enregistrement nouveau, modifié ou supprimé est traité comme une mise à jour du répertoire existant. Dans ce cas le fournisseur GCC peut choisir d'émettre de nouveau sa partie des répertoires de conférence et d'application dans leur totalité ou d'émettre l'information sous forme de mise à jour en n'envoyant que l'information concernant l'enregistrement modifié. La dernière méthode est préférable afin de minimiser la taille et la durée de transmission de la PDU. Si plus d'une modification a été faite à la suite de la réception de plusieurs primitives de demande GCC-Application-Enroll depuis la dernière mise à jour, elles peuvent être concaténées dans une PDU unique. Si seule l'information de l'enregistrement d'application a été modifiée depuis la dernière instance d'une session de protocole d'application, il n'est nécessaire de transmettre dans la PDU RosterUpdateIndication que l'enregistrement d'application modifié mais non la liste des capacités d'application. La liste des capacités d'application d'une session de protocole d'application nécessite une mise à jour dans les trois cas suivants: si le contenu concernant une entité de protocole d'application déjà inscrite a été modifié, si une entité de protocole d'application nouvellement inscrite fait partie d'une session de protocole d'application dont les entités de protocole d'application homologues sont déjà inscrites sur le nœud local ou sur un nœud subordonné de la hiérarchie de connexion ou si l'inscription d'un nœud est résiliée alors qu'il reste d'autres entités de protocole d'application homologues inscrites sur le nœud local ou sur un nœud subordonné de la hiérarchie de connexion. S'il existe plus d'une entité de protocole d'application homologue après une inscription ou une résiliation, le fournisseur GCC reconstituera la liste de capacités d'application condensée de cette session de protocole d'application avant de la placer dans la PDU RosterUpdateIndication. Cette condensation sera faite selon la procédure décrite au § 8.3.4.

Le fournisseur GCC examinera l'indicateur de capacité de présidence d'opération avant d'envoyer la PDU RosterUpdateIndication dans le cas où l'indicateur d'activité de l'entité de protocole d'application qui s'inscrit est positionné sur actif. Si ce dernier est positionné, le fournisseur GCC garantira que, pour toute session de protocole d'application, il n'existe pour ce nœud qu'une seule entité de protocole d'application figurant dans le répertoire d'application avec un indicateur de capacité de présidence d'opération positionné. Si le répertoire d'application ne contient pas actuellement d'entité de protocole d'application homologue ayant cet indicateur positionné, la nouvelle entité de protocole d'application qui s'inscrit peut être présente avec cet indicateur positionné dans la PDU RosterUpdateIndication émise. S'il y a déjà une entité de protocole d'application homologue de ce nœud figurant dans le répertoire d'application avec l'indicateur positionné, le fournisseur GCC peut ajouter la nouvelle entité de protocole d'application homologue sans positionner son indicateur. Il peut aussi, comme alternative, ajouter la nouvelle entité de protocole d'application homologue avec cet indicateur positionné et positionner sur FALSE (faux) l'indicateur de l'entité de protocole d'application qui avait précédemment la capacité de présidence d'opération. La règle du choix de l'entité de protocole d'application homologue à inclure dans le répertoire d'application avec cet indicateur positionné est un problème local qui se situe en dehors du domaine d'application de la présente Recommandation. Si l'indicateur d'activité/inactivité de l'entité de protocole d'application qui s'inscrit est positionné sur inactif, l'indicateur de capacité de présidence d'opération de la primitive de demande sera ignoré et le champ correspondant du répertoire mis à jour sera positionné sur FALSE (faux).

Dans le cas du fournisseur GCC sommital, la réussite de la modification d'une entrée du répertoire d'application local a pour résultat une modification directe du répertoire d'application de conférence

et la notification du nouveau répertoire d'application de conférence à l'ensemble des autres nœuds de la conférence selon la procédure décrite au § 8.3.2.5.

Dans tous les cas de réussite, le fournisseur GCC générera une primitive de confirmation GCC-Application-Enroll indiquant un résultat positif qu'il enverra au point d'accès GCCSAP du demandeur.

#### **8.3.2.4 Mise à jour d'une entrée du répertoire de conférence**

Si une primitive de demande GCC-Conference-Announce-Presence (annonce de présence de conférence) est reçue après la transmission d'un répertoire initial, le nouvel enregistrement de nœud contenu dans cette demande sera traité comme une mise à jour du répertoire de conférence. Comme dans le cas d'une mise à jour d'un enregistrement d'application, le fournisseur GCC enverra son nouvel enregistrement de nœud au nœud immédiatement supérieur dans une unité PDU RosterUpdateIndication (indication de mise à jour de répertoire). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur du nœud supérieur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RosterUpdateIndication est donné par le Tableau 8-35.

Comme dans le cas de l'inscription, le fournisseur GCC peut choisir d'émettre de nouveau la totalité de ses parties des répertoires d'application et de conférence ou envoyer seulement le nouvel enregistrement de nœud sous forme de mise à jour. La dernière méthode est préférable afin de minimiser la taille et donc la durée de transmission de l'unité PDU.

Si un fournisseur GCC reçoit une primitive d'indication MCS-Detach-User (détachement d'utilisateur) contenant un identificateur d'utilisateur qui est celui d'une entité de protocole d'application inscrite localement, il mettra à jour le répertoire de manière à supprimer cette entité de protocole d'application et il émettra vers le point d'accès GCCSAP une primitive d'indication GCC-Permission-To-Enroll (permission d'inscription) correspondant à cette entité de protocole d'application.

#### **8.3.2.5 Propagation de mises à jour du répertoire vers le fournisseur GCC sommital**

Lorsqu'il reçoit une unité PDU RosterUpdateIndication (indication de mise à jour de répertoire) d'un nœud directement subordonné, un fournisseur GCC mettra à jour en premier lieu son sous-ensemble des répertoires de conférence et d'application. Chaque nœud de la conférence gèrera le sous-ensemble des répertoires de conférence et d'application correspondant à ce nœud ainsi qu'à l'ensemble des nœuds subordonnés. Pour toute entité de protocole d'application qui a été inscrite sur l'un quelconque de ces nœuds, une liste d'enregistrements d'application contenant l'identificateur du nœud correspondant à cet enregistrement et une liste de capacités d'application partiellement condensée sont maintenues. Cette dernière constitue la liste de capacités d'application correspondant au résultat de l'application des règles de condensation aux listes de capacités d'application des nœuds directement subordonnés. Les listes subordonnées sont elles-mêmes partiellement condensées.

La procédure utilisée par le fournisseur GCC pour mettre à jour son sous-ensemble de répertoire dépend de la méthode mise à jour utilisée par l'unité PDU RosterUpdateIndication reçue. Si l'indicateur de rafraîchissement signale un rafraîchissement complet, la procédure suivante est utilisée. En premier lieu, tous les enregistrements de nœud et tous les enregistrements d'application sont retirés du sous-ensemble local de répertoires de conférence et d'application s'ils correspondent au nœud émetteur de l'unité PDU ou à des nœuds connus précédemment comme subordonnés de ce nœud. Tous les enregistrements de conférence et d'application listés dans l'unité PDU sont ensuite ajoutés au répertoire. La totalité de la liste de capacités d'application correspondant au nœud demandeur est supprimée et remplacée par la liste de capacités d'application fournie dans l'unité PDU. Une nouvelle liste de capacités d'application condensée sera recalculée à partir de chacune des listes de capacités d'application des nœuds directement subordonnés pour les entités de protocole d'application correspondant à l'une des situations suivantes: soit il est connu qu'elles sont

actuellement inscrites sur le nœud courant ou un nœud de niveau inférieur, soit il s'agit d'une nouvelle entité de protocole d'application indiquée dans la nouvelle PDU reçue, y compris celles qui ne sont pas données explicitement dans cette PDU. La condensation sera faite selon la procédure décrite au § 8.3.4.

Une procédure légèrement différente est utilisée dans le cas où la PDU RosterUpdateIndication n'indique pas un rafraîchissement total. En premier lieu, s'il est indiqué que l'information d'enregistrement de nœud a été modifiée et qu'elle se présente sous la forme d'un rafraîchissement de la liste d'enregistrements de nœud, tous les enregistrements de nœud sont retirés du sous-ensemble local du répertoire de conférence s'ils correspondent au nœud émetteur ou à des nœuds connus précédemment comme subordonnés de ce nœud. Tous les enregistrements de nœud listés dans la PDU sont ensuite ajoutés au répertoire. Il est possible que pour une session de protocole d'application donnée, la PDU indique des nœuds pour lesquels il n'existe plus d'entités de protocole d'application inscrites. Les mises à jour du répertoire de conférence – addition, remplacement ou suppression – sont faites conformément aux indications correspondantes de la PDU. Une tentative est ignorée si elle essaye de changer un enregistrement qui existe déjà, de remplacer ou de supprimer un enregistrement qui n'existe pas. Dans tous les cas, tous les enregistrements d'application correspondant à des nœuds supprimés qui se trouvaient précédemment dans le sous-ensemble de répertoire sont également supprimés, même si une information de mise à jour correspondante se trouve dans la partie répertoire d'application de la PDU. Une procédure similaire est employée pour toute session de protocole d'application correspondant à un enregistrement d'application pour lequel un changement a été indiqué. Si le changement a été fait sous la forme d'un rafraîchissement, toutes les entités de protocole d'application correspondant aux sessions de protocole d'application du nœud émetteur de la PDU ou de nœuds connus précédemment comme subordonnés sont supprimées du sous-ensemble local. Tous les enregistrements d'application présents dans la PDU sont ensuite ajoutés au répertoire. Il est possible que pour une session de protocole d'application donnée, la PDU indique des nœuds pour lesquels il n'existe plus d'entités de protocole d'application inscrites. Les mises à jour du répertoire de conférence – addition, remplacement ou suppression – sont faites conformément aux indications correspondantes de la PDU. Une tentative est ignorée si elle essaye de changer un enregistrement qui existe déjà, de remplacer ou de supprimer un enregistrement qui n'existe pas. Dans le cas du rafraîchissement complet, la procédure utilisée est comparable à celle employée pour les listes de capacités d'application. Dans ce cas toutefois, les nouvelles listes de capacités d'application ne sont recalculées que pour les ensembles d'entités de protocole d'application homologues ayant indiqué un changement.

Dans les deux cas ci-dessus, l'information mise à jour pour le répertoire d'application et pour chaque session de protocole d'application comporte un numéro d'instance. Si la mise à jour reçue a occasionné un changement dans l'information de conférence, le numéro d'instance de cette information sera incrémenté de 1 modulo  $2^{16}$ . Il en sera de même pour toute session de protocole d'application dont l'information a été modifiée en ce qui concerne les enregistrements d'application ou la liste de capacités d'application. Si des changements multiples ont lieu mais sont transmis comme une mise à jour unique, ceux-ci peuvent compter comme un incrément unique du numéro d'instance. Les numéros d'instance sont gérés localement pour le sous-ensemble concernant le répertoire du nœud et des nœuds subordonnés. Dans le cas du fournisseur GCC sommital, les numéros d'instance s'appliquent aux répertoires de conférence et d'application diffusés à tous les nœuds et présents dans la primitive GCC-Application-Roster-Report (compte rendu de répertoire d'application). L'information de conférence comprend également un indicateur qui signale si des nœuds ont été ajoutés et/ou supprimés depuis la dernière instance. Il existe de même pour chaque session de protocole d'application un indicateur qui signale si des nœuds ont été ajoutés et/ou supprimés depuis la dernière instance de l'information concernant cet ensemble d'entités de protocole d'application. Dans ce cas, l'addition ou la suppression de nœuds indique qu'une entité de

protocole d'application a fait l'objet d'une inscription ou d'une résiliation dans un nœud sans que le nœud lui-même ait été nécessairement ajouté ou supprimé de la conférence.

Une fois que l'information de répertoire de conférence et/ou d'application a été recalculée, un fournisseur GCC qui n'est pas le fournisseur GCC sommital générera une PDU RosterUpdateIndication (indication de mise à jour de répertoire) qui sera envoyée au fournisseur GCC immédiatement supérieur. Le format de l'information contenue dans cette PDU peut indiquer soit un rafraîchissement complet, soit un rafraîchissement de certains ensembles d'entités de protocole d'application homologues et/ou de l'information de conférence, soit encore des mises à jour individuelles. Le choix du format à utiliser pour envoyer l'information de mise à jour est laissé au fournisseur GCC. Il est préférable de faire un choix minimisant la taille et la durée de transmission de la PDU.

### **8.3.2.6 Distribution des répertoires de conférence et d'application**

Lorsque le fournisseur GCC sommital reçoit une unité RosterUpdateIndication (indication de mise à jour de répertoire), il modifiera l'information des répertoires de conférence et d'application comme décrit ci-dessus dans le cas d'une MCU intermédiaire. Il s'agira toutefois dans ce cas des répertoires de conférence et d'application complets plutôt que de sous-ensembles. Une fois le répertoire complet mis à jour, le fournisseur GCC sommital diffusera cette information à l'ensemble des nœuds par l'envoi d'une PDU RosterUpdateIndication. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) avec en paramètre l'identificateur de canal GCC-Broadcast-Channel (canal de diffusion), une priorité de données élevée et la PDU dans le champ de données.

La PDU RosterUpdateIndication sera diffusée sous la forme d'un rafraîchissement complet si le répertoire de conférence a été modifié par l'addition de nouveaux nœuds. Ceci signifie que les répertoires de conférence et d'application seront transmis intégralement à tous les nœuds. Si la modification du répertoire n'implique pas de nouveaux nœuds, le fournisseur GCC sommital peut choisir d'envoyer la mise à jour soit comme un rafraîchissement complet, soit comme une série de rafraîchissements contenant une ou plusieurs des informations suivantes: la liste d'enregistrements de nœud, la liste des enregistrements d'application de certains ou de la totalité des ensembles d'entités de protocole d'application homologues, la liste des capacités d'application de certaines ou de la totalité des sessions de protocole d'application.

Tout fournisseur GCC qui reçoit sur le canal GCC-Broadcast-Channel une PDU RosterUpdateIndication signalant un changement quelconque dans le répertoire de conférence générera une primitive d'indication GCC-Conference-Roster-Report (compte rendu de répertoire de conférence) qu'il enverra au point de contrôle GCCSAP. Si la PDU reçue indique un changement de la totalité ou d'une partie du répertoire d'application, le fournisseur GCC générera une série d'indications GCC-Application-Roster-Report et les enverra aux points GCCSAP associés à chaque entité de protocole d'application inscrite dans une session de protocole d'application pour laquelle une mise à jour a été reçue. Il peut également émettre des indications GCC-Application-Roster-Report à destination d'autres points GCCSAP, mais l'opportunité d'une telle mesure est un problème local qui est considéré comme ne faisant pas partie du domaine d'application de la présente Recommandation. Il n'est nécessaire de transmettre dans la primitive correspondante que la partie du répertoire d'application de conférence associée à la clé de session de cette entité de protocole d'application. Le fournisseur GCC peut choisir de transmettre des parties du répertoire concernant d'autres clés de session, mais l'opportunité d'une telle mesure est un problème local qui est considéré comme ne faisant pas partie du domaine d'application de la présente Recommandation. Dans le cas d'une entité de protocole d'application ne possédant pas d'identificateur de session et enrôlée comme inactive, le fournisseur GCC enverra au point d'accès GCCSAP concerné une primitive d'indication GCC-Application-Roster-Report (compte rendu de répertoire d'application) pour les parties du répertoire qui concernent toute session de protocole d'application utilisant le même protocole d'application que l'entité de protocole d'application. Le

fournisseur GCC enverra également au point de contrôle GCCSAP une primitive d'indication GCC-Application-Roster-Report comprenant les mises à jour du répertoire d'application pour tous les ensembles d'entités de protocole d'application homologues pour lesquels la PDU reçue a indiqué un changement.

### **8.3.2.7 Départ des nœuds d'une conférence**

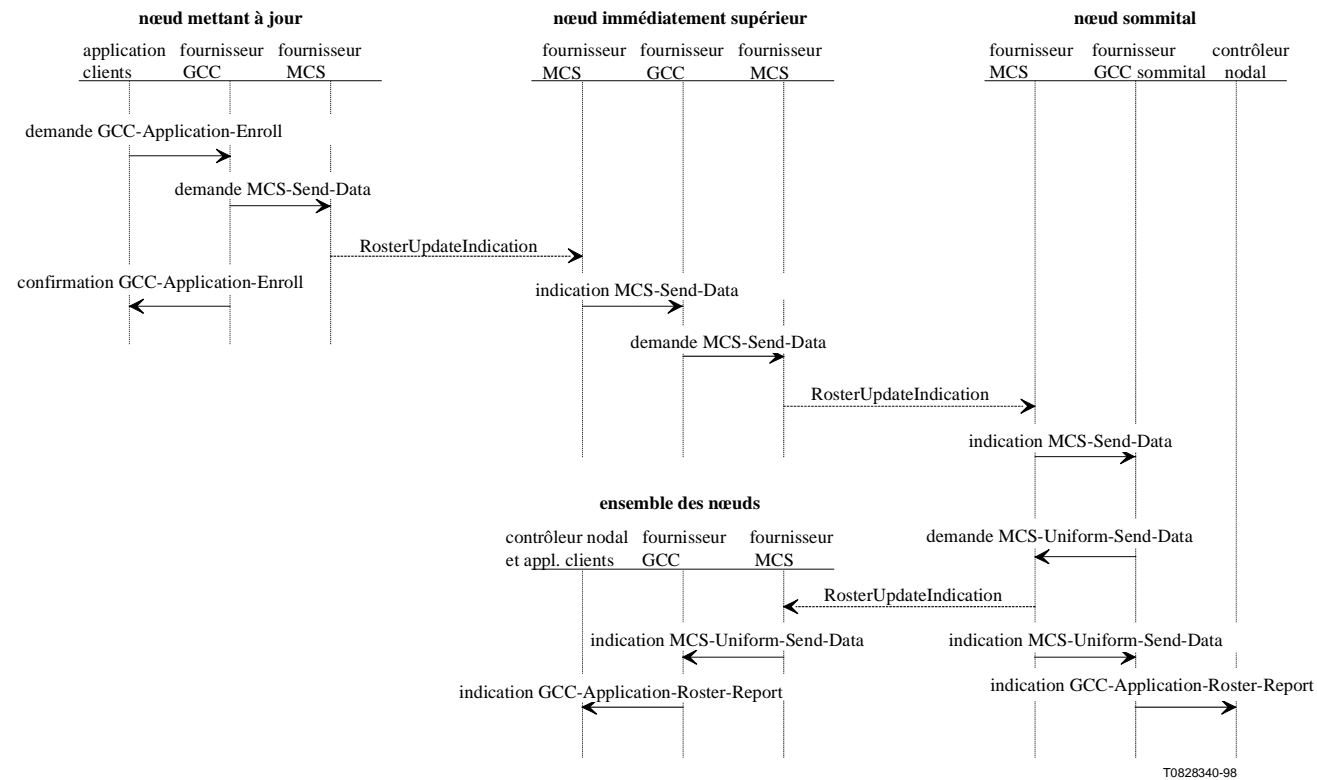
Lorsqu'ils reçoivent une primitive d'indication MCS-Detach-User (détachement d'utilisateur), les nœuds des MCU vérifieront son identificateur d'utilisateur et détermineront s'il correspond à l'identificateur d'un nœud qui leur est directement subordonné. Dans ce cas, ils supprimeront dans leur sous-ensemble de répertoires de conférence et d'application toutes les entrées correspondant à ce nœud ainsi qu'à tout nœud dont il est connu qu'il est connecté sous ce nœud. Ils recalculeront ensuite les listes de capacités d'application pour tous les ensembles d'entités de protocole d'application homologues de la façon décrite au § 8.3.4. Une fois que le sous-ensemble de répertoires a été complètement mis à jour, le fournisseur GCC mettra en œuvre la procédure décrite au § 8.3.2.5 pour propager cette information vers les autres nœuds de la conférence.

Le fournisseur GCC d'un nœud qui quitte une conférence (par déconnexion, fin de la conférence ou éjection de la conférence) produira une primitive d'indication GCC-Permission-To-Enroll (permission d'immatriculation) qui révoque l'autorisation de s'inscrire dans la conférence correspondante. Il émettra cette indication vers tous les points GCCSAP qui ne sont pas des points de contrôle.

### **8.3.2.8 Exemple de mise à jour de répertoire**

La Figure 8-5 donne un exemple de primitive de demande GCC-Application-Enroll émise pendant une conférence qui a pour effet une mise à jour du répertoire d'application de conférence. Dans cet exemple, le nœud se situe au troisième niveau de la hiérarchie de connexion. Il envoie l'indication de mise à jour au nœud supérieur suivant qui formate et envoie à son tour une indication de mise à jour vers le nœud supérieur suivant, qui est dans cet exemple le nœud sommital de la hiérarchie. Le nœud sommital assemble le répertoire d'application complet et diffuse les parties correspondant à la session de protocole d'application mises à jour à tous les nœuds de la conférence, ce qui a pour résultat l'envoi de primitives GCC-Application-Report vers les contrôleurs nodaux de l'ensemble des nœuds et vers les entités de protocole d'application homologues existantes.





**Figure 8-5 – Exemple de mise à jour du répertoire d'application**

### **8.3.3 Protocole de répertoire échelonné**

Le protocole de répertoire échelonné a été introduit afin de prendre en charge divers niveaux d'échelonnabilité à l'intérieur d'une même conférence. Ce protocole se fonde sur l'utilisation de mises à jour delta au lieu de mises à jour complètes du répertoire (sur lesquelles le protocole de répertoire original se fondait) en vue de transférer les informations de répertoire qui subissent le plus de changements dynamiques. Ce répertoire se fonde également sur l'utilisation des catégories nodales. Il spécifie un comportement nettement différent selon la catégorie nodale assignée à chaque nœud. Les sous-paragraphes qui suivent décrivent en détail le protocole de répertoire échelonné.

#### **8.3.3.1 Aperçu général**

Les deux répertoires, de conférence et d'application, sont communiqués aux nœuds qui utilisent le même ensemble d'unités PDU. L'unité PDU RosterUpdateIndication (indication de mise à jour de répertoire) sert à envoyer vers d'autres nœuds de la conférence des informations de répertoire complètes ou partielles. Par ailleurs, l'unité PDU RosterRefreshRequest (demande de mise à jour du répertoire) sert à envoyer, à partir d'un nœud parent, des demandes de mise à jour complète d'un répertoire de conférence et/ou d'application.

Lorsque chaque nœud (autre que le nœud sommital) annonce pour la première fois ses informations de répertoire pour entrer dans une conférence, ce nœud envoie une indication de mise à jour de répertoire au nœud qui lui est immédiatement supérieur dans la hiérarchie. Tel est le cas pour les trois catégories nodales. Des mises à jour ultérieures d'une partie quelconque des informations de répertoire de conférence et d'application sont annoncées par la réémission d'une unité d'indication de mise à jour de répertoire contenant les nouvelles informations.

Lorsqu'un nœud met à jour sa partie du répertoire (ou l'annonce pour la première fois), cette information est propagée vers le nœud qui lui est hiérarchiquement supérieur, jusqu'à ce qu'elle atteigne le fournisseur GCC sommital. Celui-ci est chargé de former les répertoires de conférence et d'application complets puis de distribuer à tous les nœuds de la conférence des mises à jour delta de ces répertoires. La décision d'ajouter ou non un nœud au répertoire dépend de la catégorie du nœud qui entre dans la conférence. Par exemple, les nœuds conventionnels seront toujours ajoutés au répertoire, tandis que les nœuds anonymes ne sont jamais ajoutés. Les nœuds comptés ne sont ajoutés qu'aux répertoires tenus à jour par des nœuds conventionnels.

Si les informations de répertoire qui se propagent en remontant dans la hiérarchie de connexion sont associées à un nœud conventionnel, chaque nœud intermédiaire (ponts MCU) est chargé de former un sous-ensemble des répertoires de conférence et d'application complets en mémorisant ces informations de répertoire nodal. Le répertoire de conférence est un sous-ensemble qui comprend des enregistrements nodaux pour le nœud récepteur et pour tous les autres nœuds conventionnels qui lui sont hiérarchiquement inférieurs dans la connexion. Le répertoire d'application comporte une liste séparée d'enregistrements d'application pour chaque session du protocole d'application existante (dans un nœud conventionnel) située dans le nœud qui reçoit la mise à jour de répertoire (ou au-dessous de ce nœud). Il comporte également les capacités d'application pour ces sessions. Dès qu'il reçoit une indication de mise à jour de répertoire d'un nœud conventionnel inférieur, un nœud intermédiaire apporte les modifications appropriées à son sous-ensemble de répertoires, puis transmet la mise à jour de répertoire au nœud qui lui est immédiatement supérieur, en émettant une indication de mise à jour de répertoire. On notera que, quelle que soit la catégorie nodale, l'indication de mise à jour de répertoire est toujours envoyée. La seule différence ici est que les nœuds comptés et anonymes ne sont pas tenus à jour dans les sous-ensembles de répertoire se trouvant dans des nœuds intermédiaires.

La raison principale de la tenue à jour de sous-ensembles des informations de répertoire dans des nœuds intermédiaires est de conserver l'ensemble de capacités qui a été condensé. Etant donné que les nœuds comptés et anonymes n'ont pas d'incidence sur les capacités, il n'est pas nécessaire de

tenir à jour leurs enregistrements dans les sous-ensembles du répertoire. Si une mise à jour issue d'un nœud conventionnel inférieur comportait une modification ayant entraîné un nouveau calcul des listes de capacités d'application pour au moins une session du protocole d'application, les nouvelles informations de répertoire pour le sous-ensemble de répertoire ne peuvent être produites que si l'on connaît les listes de capacités d'application individuelles pour chaque nœud conventionnel immédiatement inférieur à ce nœud dans la hiérarchie de connexion (afin que les règles de condensation puissent être réappliquées avec les nouvelles informations mises à jour). Le fournisseur GCC de chaque nœud doit tenir à jour ces informations localement.

Lorsqu'une mise à jour de répertoire associée à un nœud conventionnel atteint le fournisseur GCC sommital, les nouvelles informations de répertoire doivent être diffusées à tous les nœuds de la conférence (y compris les nœuds comptés et anonymes). A cette fin, on ne diffuse à tous les nœuds que les informations delta contenues dans une unité PDU d'indication de mise à jour de répertoire. Si l'information de mise à jour de répertoire qui atteint le fournisseur GCC sommital est associée à un nœud compté, une mise à jour delta est diffusée aussi bien aux nœuds conventionnels qui participent à la conférence qu'au nœud compté entrant qui a lancé la mise à jour de répertoire initiale. Aucun autre nœud, compté ou anonyme, ne recevra de mise à jour de répertoire lorsque cet événement se produira. Les mises à jour de nœud anonyme, reçues par le fournisseur GCC sommital, ne provoquent pas la diffusion d'une mise à jour. Par contre, le fournisseur GCC sommital local peut tenir à jour une liste distincte des nœuds anonymes qui sont actuellement (ou qui ont été) à l'écoute de la conférence à des fins informatives ou statistiques (ce qui relève de la mise en œuvre locale).

Les informations de répertoire peuvent être envoyées au moyen de l'unité PDU RosterUpdateIndication de trois façons différentes. Le répertoire peut être envoyé sous forme de mise à jour des répertoires de conférence et d'application complets, en remplacement de toutes les entrées de répertoire existantes. En variante, des portions du répertoire peuvent être envoyées de manière que le répertoire de conférence et/ou des portions du répertoire d'application puissent être mis à jour en association avec une ou plusieurs clés de session. Dans ce cas, toutes les entrées contenues dans la partie émise du répertoire sont remplacées, mais les portions non expédiées restent inchangées. Finalement, des modifications de portions du répertoire peuvent être envoyées sous forme de mises à jour de manière que seuls soient envoyés les éléments particuliers du répertoire (comme les enregistrements d'application) qui ont été ajoutés, modifiés ou supprimés et que toutes les autres entrées soient laissées sans changement.

Etant donné que seules les unités PDU d'indication de mise à jour de répertoire sont finalement diffusées à partir du fournisseur sommital, la charge d'assimilation d'un répertoire de conférence ou d'application complet est confiée au nœud entrant et à son nœud parent. A cette fin, le nœud entrant envoie à son nœud parent une unité PDU RosterRefreshRequest pour demander une mise à jour complète de son répertoire. Le point auquel cette demande est émise dépend de la catégorie nodale du nœud entrant. Pour les nœuds conventionnels et comptés, cette demande est émise après réception d'une mise à jour de répertoire issue du fournisseur sommital pour lui-même, ce qui garantit que sa présence dans la conférence est connue une fois qu'il a reçu une mise à jour de répertoire pour lui-même (ce qui garantit la connaissance de sa présence dans la conférence). Pour les nœuds anonymes, la demande peut être formulée immédiatement après la réception de l'identificateur d'utilisateur de nœud parent.

Une fois que le nœud parent a reçu l'unité PDU de demande de mise à jour de répertoire, il doit soit traiter cette demande (s'il a accès aux répertoires requis) soit réexpédier la demande au nœud suivant dans la hiérarchie de connexion. Lorsqu'un nœud qui contient l'information requise est atteint, ce nœud envoie au nœud demandeur une indication de mise à jour de répertoire contenant une mise à jour complète de répertoire. On notera que la réponse à une demande de mise à jour de répertoire paraîtra différente selon la catégorie nodale à laquelle le nœud demandeur appartient. Par exemple, si un nœud conventionnel a émis la demande, une mise à jour complète, contenant les nœuds aussi bien conventionnels que comptés, doit être acheminée. Un nœud conventionnel parent

doit donc être atteint avant que la demande puisse être traitée (car seuls les nœuds conventionnels tiennent à jour une liste des nœuds comptés). Si le nœud demandeur est anonyme ou compté, la demande peut être prise en charge par le premier nœud qui contient le répertoire de nœud conventionnel.

Il est nécessaire que les nœuds entrants mémorisent temporairement les indications de mise à jour de répertoire reçues du fournisseur GCC sommital, avant qu'ils reçoivent la réponse à leur demande initiale de mise à jour de répertoire. Il faut en effet éviter la perte de mises à jour garantissant la synchronisation de tous les répertoires dans la conférence. Toutes les indications de mise à jour de répertoire contiennent un numéro d'instance qui est incrémenté séquentiellement chaque fois que le fournisseur sommital détecte qu'un répertoire a changé. Ce numéro d'instance de répertoire est utilisé pour déterminer l'ordre dans lequel les indications de mise à jour de répertoire doivent être traitées dans un nœud entrant, ce qui peut être particulièrement compliqué pour les nœuds conventionnels qui participent à une conférence pouvant inclure les deux types de nœuds: conventionnels et comptés: les indications de mise à jour de répertoire seront en effet reçues sur les deux canaux de diffusion GCC, pour nœuds conventionnels et pour nœuds comptés. Il appartient au nœud conventionnel de traiter les indications de mise à jour de répertoire dans l'ordre correct, sur la base de sa connaissance préalable du fait que les numéros d'instance de répertoire augmentent en séquence.

### **8.3.3.2 Nœuds entrant dans une conférence**

Lorsqu'un nœud est adjoint à une conférence, par création, addition ou invitation dans la conférence, un fournisseur GCC doit envoyer au point GCCSAP une primitive d'indication GCC-Application-Permission-To-Enroll pour toutes les entités de protocole d'application (APE, *application protocol entity*) qui ont indiqué localement leur présence au fournisseur GCC. Si un fournisseur GCC est avisé d'une entité APE additionnelle qui a déjà rejoint une conférence, il doit envoyer au point GCCSAP correspondant une indication GCC-Application-Permission-To-Enroll afin de signaler l'existence de cette conférence et d'informer l'entité APE de sa catégorie nodale.

Avant de passer à une autre mesure, le fournisseur GCC attend de recevoir, du point GCCSAP de contrôle, une demande GCC-Conference-Announce-Presence ainsi qu'une demande GCC-Application-Enroll en provenance de toutes les entités APE qui ont reçu des indications GCC-Application-Permission-To-Enroll. Certaines de ces demandes d'inscription peuvent indiquer que l'entité APE n'a pas l'intention de s'inscrire (l'indicateur d'inscription/non-inscription est mis à "non-inscription"). Pour chaque demande GCC-Application-Enroll reçue avec l'indicateur mis à "inscription", l'information fournie dans la primitive pour cette entité de protocole d'application est ajoutée au répertoire d'application local.

Le fournisseur GCC doit également attribuer localement, à chaque entité APE inscrite, un identificateur d'entité qui doit être utilisé par le fournisseur GCC pour identifier chaque entité de protocole d'application correspondante dans l'unité PDU d'indication de mise à jour de répertoire. Cet identificateur d'entité APE assigné est également inclus dans la primitive de confirmation GCC-Application-Enroll qui est renvoyée à chaque entité APE s'inscrivant, en même temps que l'identificateur du nœud local. L'identificateur d'entité est une valeur d'entier codée sur 16 bits qui doit être unique pour toutes les entités APE inscrites dans un nœud. Si une entité APE annule son inscription, la valeur de son identificateur d'entité ne doit pas être réutilisée tant que toutes les autres valeurs non attribuées à une entité de protocole d'application n'ont pas été attribuées au moment de l'annulation de l'inscription.

Le fournisseur GCC doit examiner, pour chaque entité APE inscrite, la valeur de l'indicateur de capacité de présider une opération: actif/inactif (les entités APE à indicateur inactif sont censées ne pas avoir la capacité de présider une opération). Si plusieurs entités APE inscrites ont cet indicateur actif, le fournisseur GCC doit en sélectionner une dont l'indicateur sera activé dans l'information de répertoire d'application qui sera envoyée dans l'unité PDU RosterUpdateIndication. La règle de

sélection de l'entité APE relève d'une décision locale non spécifiée dans la présente Recommandation. Une règle simple peut consister à sélectionner la première de ces entités APE qui s'inscrit.

En fonction de la catégorie nodale d'un nœud, le fournisseur GCC local doit déterminer la décision appropriée à prendre après réception de la demande GCC-Conference-Announce-Presence émise par le point GCCSAP de contrôle et après réception d'une demande GCC-Application-Enroll émise par toutes les entités APE. Etant donné que seuls les nœuds conventionnels sont autorisés à créer des sessions d'application, les fournisseurs GCC situés aux nœuds comptés et aux nœuds anonymes doivent conserver toute information d'indication de mise à jour de répertoire associée à des sessions d'entité APE inexistantes jusqu'à réception d'une unité PDU RosterUpdateIndication issue d'un nœud supérieur. Celui-ci signale que la session à laquelle l'on s'inscrit existe réellement. Les fournisseurs GCC locaux des nœuds comptés et anonymes peuvent également utiliser les clés de protocole d'application reçues lors des inscriptions initiales afin de déterminer l'intérêt local pour les indications de mise à jour de répertoire entrantes. Par exemple, chaque fois qu'un fournisseur GCC local est informé d'une nouvelle session par une indication de mise à jour de répertoire reçue soit du fournisseur sommital soit de son nœud parent, le fournisseur local doit déterminer quelles sont les entités APE qui sont intéressées en l'occurrence. Le fournisseur local utilise la clé de protocole d'application associée à la nouvelle session pour déterminer les entités APE auxquelles il enverra une indication de mise à jour. Les procédures détaillées sont décrites ci-dessous pour chaque catégorie nodale.

Un fournisseur GCC situé à un nœud conventionnel, qui n'est pas le fournisseur GCC sommital, doit envoyer une indication de mise à jour de répertoire au fournisseur GCC qui lui est immédiatement supérieur dans la hiérarchie de connexion, dès qu'il reçoit la demande GCC-Conference-Announce-Presence issue du point CGGAP de contrôle ainsi qu'une demande GCC-Application-Enroll issue de toutes les entités APE. A cette fin, il émet demande MCS-Send-Data spécifiant l'identificateur du nœud de destination en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. L'identificateur du nœud immédiatement supérieur dans la hiérarchie de connexion est déterminé au moment de la connexion à la conférence, soit à partir de l'unité PDU ConferenceCreateResponse, soit à partir de l'unité PDU ConferenceJoinResponse, soit à partir de l'unité PDU ConferenceInviteRequest, selon la manière dont la conférence a été rejointe. La valeur de l'identificateur du nœud immédiatement supérieur doit également être incluse, en tant que paramètre de nœud supérieur, dans la partie de l'unité PDU consacrée au répertoire de conférence. Le contenu de l'unité PDU RosterUpdateIndication est représenté dans le Tableau 8-35. Dans ce cas, le répertoire est envoyé sous la forme d'une mise à jour complète, y compris l'enregistrement nodal local, de concert avec les enregistrements d'application pour toutes les entités APE inscrites et avec les listes de capacités d'application pour toutes les sessions de protocole d'application correspondant à ces entités APE inscrites. Si plusieurs entités APE se sont inscrites localement pour une session de protocole d'application donnée, le fournisseur GCC doit effectuer une condensation de la liste de capacités d'application entre entités APE homologues. Il en résultera la liste de capacités d'application qui devra être incluse dans l'unité PDU RosterUpdateIndication pour cette session de protocole d'application. Cette condensation doit être effectuée selon la procédure décrite au § 8.3.4.

Un fournisseur GCC situé dans un nœud compté doit envoyer une unité d'indication de mise à jour de répertoire au fournisseur GCC qui lui est immédiatement supérieur dans la hiérarchie de connexion dès réception, du point GCCSAP de contrôle, de la demande GCC-Conference-Announce-Presence ainsi que, de toutes les entités APE, d'une demande GCC-Application-Enroll. A cette fin, il émet une demande MCS-Send-Data spécifiant l'identificateur du nœud de destination en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. La valeur de l'identificateur du nœud immédiatement supérieur dans la hiérarchie de connexion est de nouveau

incluse, sous forme de paramètre de nœud supérieur, dans la partie de l'unité PDU consacrée au répertoire de conférence. Dans ce cas, seul l'enregistrement nodal local est envoyé. Les enregistrements d'application sont conservés jusqu'à ce que le fournisseur GCC local reçoive, d'un nœud supérieur, une indication de mise à jour de répertoire l'informant de l'existence réelle des sessions inscrites localement.

Un fournisseur GCC situé dans un nœud anonyme doit envoyer une unité d'indication de mise à jour de répertoire au fournisseur GCC qui lui est immédiatement supérieur dans la hiérarchie de connexion dès réception, du point GCCSAP de contrôle, de la demande GCC-Conference-Announce-Presence ainsi que, de toutes les entités APE, d'une demande GCC-Application-Enroll. A cette fin, il émet une demande MCS-Send-Data spécifiant l'identificateur du nœud de destination en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. La valeur de l'identificateur du nœud immédiatement supérieur dans la hiérarchie de connexion est de nouveau incluse, sous forme de paramètre de nœud supérieur, dans la partie de l'unité PDU consacrée au répertoire de conférence. De nouveau, seul l'enregistrement nodal local est envoyé. On notera que cette indication de mise à jour de répertoire n'est utilisée qu'à titre d'information et n'a pas d'incidence sur le répertoire de conférence. Un nœud anonyme n'enverra jamais d'indication de mise à jour de répertoire comportant des informations de répertoire associées à une entité APE. Un nœud anonyme ne doit par ailleurs jamais envoyer plus d'une seule unité d'indication de mise à jour de répertoire.

Un nœud qui est le fournisseur GCC sommital, qui a reçu une demande GCC-Conference-Announce-Presence et qui vient d'entrer dans une conférence doit introduire l'information contenue dans sa base de données de répertoire de conférence. Mais ce nœud peut, avant d'envoyer des unités PDU, attendre d'avoir reçu des unités PDU d'indication de mise à jour de répertoire en provenance d'au moins un autre nœud participant à la conférence. Si une indication est reçue d'un nœud conventionnel, le fournisseur doit mettre à jour sa base de données de répertoires de conférence et d'application puis, en envoyant à tous les nœuds de la conférence une indication de mise à jour de conférence, renvoyer en diffusion les informations de répertoire associées à l'indication de mise à jour. A cette fin, il émet une demande MCS-Send-Data spécifiant le canal GCC-Conventional-Broadcast-Channel en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. Les mises à jour envoyées par le fournisseur GCC sommital à tous les nœuds de la conférence en raison de l'entrée d'un nouveau nœud ne sont jamais envoyées sous forme de mise à jour complète car celle-ci réduit le trafic réseau dû à l'adjonction de nouveaux nœuds à une conférence.

### **8.3.3.3 Inscription d'entités de protocole d'application**

Dès qu'il reçoit une primitive de demande d'inscription d'application GCC (GCC-Application-Enroll), un fournisseur GCC doit d'abord déterminer s'il existe déjà, associée au point GCCSAP du demandeur dans le répertoire d'application local, une entrée pour la conférence spécifiée. Si ce n'est pas le cas et que l'indicateur d'inscription/non-inscription contenu dans la demande soit mis à "inscription", une nouvelle entrée est créée dans le répertoire d'application local, contenant les informations spécifiées dans la primitive de demande. Dans ce cas, le fournisseur GCC doit également attribuer un identificateur d'entité à l'entité APE qui vient d'être inscrite, comme décrit au § 8.3.3.2. Si l'indicateur d'inscription/non-inscription a été réglé sur "non-inscription", la demande est immédiatement confirmée par la production d'une confirmation GCC-Application-Enroll et par son envoi au point GCCSAP du demandeur. Dans ce cas, aucune autre mesure n'est prise.

Si l'entrée contenue dans le répertoire d'application local existe déjà et que l'indicateur d'inscription/non-inscription soit réglé sur "inscription", le contenu de l'entrée existante est modifié de façon à refléter les nouvelles valeurs spécifiées. Si l'indicateur d'inscription/non-inscription a été

mis sur "non-inscription", le fournisseur GCC doit supprimer l'entrée correspondante du répertoire d'application local.

Si le fournisseur GCC n'est pas le fournisseur GCC sommital pour la conférence spécifiée et qu'il soit un nœud conventionnel, ce fournisseur doit envoyer son nouveau répertoire d'application local au nœud qui lui est immédiatement supérieur dans la hiérarchie de connexion, au moyen d'une unité RosterUpdateIndication. A cette fin, le fournisseur GCC doit émettre une demande MCS-Send-Data spécifiant l'identificateur du nœud supérieur en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. L'identificateur nodal du fournisseur GCC immédiatement supérieur dans la hiérarchie de connexion est déterminé au moment du raccordement à la conférence, au moyen de l'unité PDU ConferenceCreateResponse, ConferenceJoinResponse ou ConferenceInviteRequest, selon la façon dont la conférence a été approchée. Le contenu de l'unité PDU RosterUpdateIndication est décrit dans le Tableau 8-35.

Si le fournisseur GCC est un nœud compté, la procédure est un peu plus compliquée. Avant d'envoyer son nouveau répertoire d'application local au nœud qui lui est immédiatement supérieur dans la hiérarchie de connexion, le fournisseur GCC local doit déterminer si les sessions associées à chaque demande d'inscription existent réellement. Seules les demandes d'inscription qui correspondent à des sessions existantes seront retransmises, en raison de la règle qui interdit aux nœuds comptés de créer des sessions. Le fournisseur GCC local est informé des nouvelles sessions par deux moyens: soit par une unité PDU RosterUpdateIndication reçue du fournisseur sommital, soit par une réponse à une demande de mise à jour de répertoire reçue d'un nœud parent. Une fois que l'on a déterminé les demandes d'inscription qui sont associées à des sessions existantes, le fournisseur GCC envoie, au nœud qui lui est immédiatement supérieur dans la hiérarchie de connexion, une unité PDU RosterUpdateIndication contenant les enregistrements d'application appropriés. A cette fin, le fournisseur GCC doit émettre une demande MCS-Send-Data spécifiant l'identificateur du nœud supérieur en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. Les indications de mise à jour de répertoire émises par un nœud compté ne doivent jamais comporter d'information de capacité puisqu'un nœud compté ne peut pas modifier la liste de capacités condensée. Chaque entité APE doit déterminer si elle peut ou non prendre en charge les capacités établies dans une conférence.

Si le fournisseur GCC est un nœud anonyme, le fournisseur GCC local se borne à mémoriser l'information dans son répertoire d'application local et l'utilise pour déterminer les sessions de protocole d'application auxquelles s'intéresse un point d'accès au service (SAP, *service access point*) de commande GCC. Aucune information d'entité APE ne sera jamais envoyée à partir d'un nœud anonyme dans une unité d'indication de mise à jour de répertoire. Il importe que le fournisseur GCC local tienne à jour cette base d'information de façon qu'il puisse faire suivre vers les points SAP appropriés toutes les informations relatives aux répertoires d'application nouveaux ou modifiés. Cela garantit que toutes les entités APE exploitant un nœud anonyme auront accès à toutes les informations nécessaires pour rejoindre et écouter une session particulière.

En général, si l'inscription a été faite avant la transmission initiale d'une unité PDU RosterUpdateIndication, l'enregistrement nouveau ou modifié est inclus dans l'information de répertoire initiale et envoyé en tant que mise à jour complète comme décrit au § 8.3.3.2. Si une entité APE s'est inscrite avant la transmission initiale, son entrée enregistrée est supprimée du répertoire et n'est jamais insérée dans une unité PDU transmise.

Si l'inscription s'est produite après la transmission initiale d'une unité PDU d'indication de mise à jour de répertoire, l'enregistrement nouveau, modifié ou supprimé est traité comme une mise à jour du répertoire existant. Dans ce cas, le fournisseur GCC peut soit choisir de réexpédier entièrement sa portion des répertoires de conférence et d'application, soit envoyer les informations sous forme de mise à jour (en n'envoyant que les renseignements qui concernent l'enregistrement qui a été modifié). L'on préfère que cette dernière méthode soit utilisée afin de minimiser la longueur des

unités PDU et donc leur temps de transmission. Si plus d'une seule modification a été apportée (plusieurs demandes GCC-Application-Enroll ont été reçues) depuis la dernière mise à jour, ces modifications peuvent être concaténées en une seule unité PDU. Pour une session de protocole d'application donnée, si seule l'information d'enregistrement d'application a changé depuis la dernière instance de cette session du protocole d'application, la liste de capacités d'application n'a pas besoin d'être incluse dans l'unité PDU d'indication de mise à jour de répertoire, qui ne contiendra donc que l'enregistrement d'application modifié. Pour une session donnée du protocole d'application dans un nœud conventionnel, la liste des capacités d'application nécessite une mise à jour si l'une des conditions suivantes est vérifiée:

- son contenu a été modifié à partir d'une entité APE déjà inscrite;
- une entité APE nouvellement inscrite fait partie d'une session de protocole d'application pour laquelle des entités APE homologues ont déjà été inscrites au nœud local ou à des nœuds inférieurs au nœud local dans la hiérarchie de connexion;
- un nœud est désinscrit s'il y a d'autres entités APE homologues qui restent inscrites au nœud local ou à des nœuds inférieurs au nœud local dans la hiérarchie de connexion.

S'il y a plusieurs entités APE homologues, soit après une inscription soit avant une désinscription, le fournisseur GCC d'un nœud conventionnel doit reconstituer la liste condensée des capacités d'application pour cette session du protocole d'application avant de l'inclure dans l'unité PDU d'indication de mise à jour de répertoire. Cette condensation doit être effectuée par la procédure décrite au § 8.3.4.

Si l'indicateur d'activité/inactivité pour l'entité APE qui s'inscrit est réglé à la valeur "entité active" avant l'envoi de l'unité PDU RosterUpdateIndication, le fournisseur GCC doit examiner l'indicateur de capacité de présider une opération. Si cet indicateur est activé, le fournisseur GCC doit faire en sorte qu'une seule entité APE homologue de ce nœud soit incluse dans le répertoire d'application avec cet indicateur activé, pour chaque session du protocole d'application. Si aucune entité APE homologue de ce type n'a été sélectionnée jusqu'alors dans le répertoire d'application, l'entité APE nouvellement inscrite peut être incluse dans l'unité PDU RosterUpdateIndication transmise avec cet indicateur activé. Si une entité APE homologue se trouve déjà dans le répertoire d'application de ce nœud avec cet indicateur activé, le fournisseur GCC peut:

- soit inclure l'entité APE nouvellement inscrite dans l'unité PDU RosterUpdateIndication mais sans activer cet indicateur;
- soit inclure l'entité APE nouvellement inscrite avec cet indicateur activé et la précédente entité APE indiquée comme étant capable de présider une opération, avec cet indicateur mis à la valeur FALSE.

La règle de sélection de l'entité APE homologue à inclure dans le répertoire d'application avec cet indicateur activé est une question locale qui est hors du domaine d'application de la présente Recommandation. Si l'indicateur d'activité/inactivité pour l'entité APE qui s'inscrit est mis à "entité inactive", l'indicateur de capacité de présider une opération contenue dans la primitive de demande doit être ignoré et le champ correspondant dans le répertoire mis à jour doit être mis à FALSE. L'indicateur de capacité de présider une opération doit toujours être mis à FALSE pour les entités APE qui résident dans des nœuds présidés et anonymes.

Dans le cas du fournisseur GCC sommital, la modification normale d'une entrée du répertoire d'application se traduit par une modification directe du répertoire d'application de conférence et de la procédure associée afin de signaler à tous les autres nœuds de la conférence l'existence du nouveau répertoire d'application de conférence comme décrit au § 8.3.3.6.

Dans tous les cas favorables, le fournisseur GCC doit émettre une primitive de confirmation GCC-Application-Enroll indiquant un résultat favorable. Le fournisseur GCC doit envoyer cette primitive au point GCCSAP du demandeur.



#### **8.3.3.4 Mise à jour d'une entrée du répertoire de conférence**

Si une demande GCC-Conference-Announce-Presence est reçue alors que l'on a déjà transmis un répertoire initial, le nouvel enregistrement nodal contenu dans cette demande doit être traité comme une mise à jour du répertoire de conférence. Comme dans le cas d'un enregistrement d'application mis à jour, le fournisseur GCC doit alors envoyer son nouvel enregistrement nodal au nœud qui lui est immédiatement supérieur dans la hiérarchie de connexion, au moyen d'une unité PDU RosterUpdateIndication. A cette fin, le fournisseur GCC doit émettre une demande MCS-Send-Data spécifiant l'identificateur du nœud supérieur en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU sélectionnée dans le champ de données. Le contenu de l'unité PDU RosterUpdateIndication est représenté dans le Tableau 8-35. On notera qu'une demande GCC-Conference-Announce-Presence ne doit jamais être reçue lorsqu'on a déjà transmis un répertoire initial à partir de nœuds anonymes.

Si un fournisseur GCC reçoit une indication MCS-Detach-User pour laquelle l'identificateur d'utilisateur est celui d'une entité APE inscrite localement, ce fournisseur GCC doit mettre à jour le répertoire afin de supprimer cette entité APE; il doit également émettre une primitive d'indication GCC-Permission-To-Enroll vers le point GCCSAP correspondant à cette entité APE.

#### **8.3.3.5 Propagation de mises à jour de répertoire vers le fournisseur GCC sommital**

Dès qu'il reçoit une unité PDU RosterUpdateIndication d'un nœud directement inférieur dans la hiérarchie de connexion, un fournisseur GCC doit d'abord mettre à jour son sous-ensemble de répertoires de conférence et d'application. Chaque nœud de la conférence doit tenir à jour le sous-ensemble des répertoires de conférence et d'application correspondant à ce nœud, ainsi qu'à tous les nœuds qui lui sont inférieurs dans la hiérarchie de connexion. Ce sous-ensemble ne doit jamais comporter d'enregistrements nodaux ou d'enregistrements applicatifs associés à des nœuds anonymes. Chaque enregistrement nodal contient un indicateur qui signale la catégorie nodale de chaque nœud.

Pour chaque entité APE qui a été inscrite dans l'un de ces nœuds, on tient à jour une liste des enregistrements d'application comprenant l'identificateur du nœud auquel chaque enregistrement correspond, ainsi qu'une liste partiellement condensée des capacités d'application. Il s'agit de la liste de capacités d'application qui fait suite à l'application de la série de règles applicable aux listes de capacités d'application des nœuds immédiatement inférieurs à ce nœud dans la hiérarchie de connexion (qui sont elles-mêmes partiellement condensées).

La procédure que le fournisseur GCC doit utiliser pour mettre à jour son sous-ensemble de répertoires dépend de la méthode de mise à jour utilisée dans l'unité PDU RosterUpdateIndication reçue. Si l'on a indiqué (au moyen de l'indicateur de mise à jour complète contenu dans l'unité PDU) que la mise à jour était complète, la procédure suivante est appliquée. Tout d'abord, tous les enregistrements nodaux et tous les enregistrements d'application correspondant au nœud à partir duquel l'unité PDU a été reçue, ou correspondant aux nœuds qui étaient jusque-là réputés inférieurs à ce nœud dans la hiérarchie de connexion, sont supprimés du sous-ensemble local des répertoires de conférence et d'application. Tous les enregistrements de conférence et d'application énumérés dans l'unité PDU sont ensuite ajoutés au répertoire. Toute la liste des capacités d'application correspondant au nœud demandeur inférieur, pouvant avoir été mémorisée dans ce nœud, est supprimée et est remplacée par la liste des capacités d'application spécifiée dans l'unité PDU. Pour toutes les entités APE, actuellement connues comme étant inscrites au nœud actuel ou à un nœud inférieur, ainsi que pour toute nouvelle entité APE indiquée dans l'unité PDU nouvellement reçue (y compris celles qui ne sont pas spécifiées dans cette unité PDU), une nouvelle liste condensée de capacités d'application doit être calculée à partir de chacune des listes de capacités d'application issues des nœuds immédiatement inférieurs. Cette condensation doit être effectuée selon la procédure décrite au § 8.3.4.

Si l'unité PDU RosterUpdateIndication n'indiquait pas de mise à jour complète, une procédure légèrement différente est suivie. Tout d'abord, si l'information d'enregistrement nodal a été signalée comme étant modifiée et que cette modification a été effectuée au titre d'une mise à jour de la liste des enregistrements nodaux, tous les enregistrements nodaux correspondant au nœud duquel l'unité PDU a été reçue (ou aux nœuds jusque là réputés inférieurs à ce nœud dans la hiérarchie de connexion) sont supprimés du sous-ensemble local du répertoire de conférence. Tous les enregistrements nodaux énumérés dans l'unité PDU sont ensuite ajoutés au répertoire. Si la modification a été apportée sous la forme d'une série de mises à jour des enregistrements nodaux, tous les enregistrements indiqués comme ajoutés sont ajoutés au sous-ensemble de répertoire; tous les enregistrements indiqués comme remplacés sont utilisés pour remplacer les enregistrements existants; et tous les enregistrements indiqués comme supprimés sont supprimés. On ne tient pas compte d'une tentative de modification d'un enregistrement qui existe déjà, ni d'une tentative de modification ou de suppression d'un enregistrement qui n'existe pas. De toute façon, pour tous les nœuds qui se trouvaient déjà dans le sous-ensemble de répertoire et qui ont été supprimés, tous les enregistrements d'application correspondant à ces nœuds sont également supprimés (que les informations de mise à jour correspondantes aient ou non été incluses dans la partie de l'unité PDU consacrée au répertoire d'application). Si une quelconque information d'enregistrement d'application a été signalée comme modifiée, une procédure analogue est suivie pour chaque session du protocole d'application. Pour chaque session du protocole d'application, si la modification a été effectuée au titre d'une mise à jour, tous les enregistrements d'application correspondant au nœud à partir duquel l'unité PDU a été reçue (ou aux nœuds jusque là réputés inférieurs à ce nœud dans la hiérarchie de connexion) sont supprimés du sous-ensemble local du répertoire d'application. Tous les enregistrements d'application énumérés dans l'unité PDU sont ensuite ajoutés au répertoire. Il est possible que, pour une session de protocole d'application donnée, il soit indiqué qu'il n'existe pas de nœuds possédant l'inscription de cette entité APE. Si la modification a été effectuée sous la forme d'une série de mises à jour d'enregistrements d'application, tous les enregistrements indiqués comme ajoutés sont ajoutés au sous-ensemble de répertoire; tous les enregistrements indiqués comme remplacés sont utilisés pour remplacer l'enregistrement existant; et tous les enregistrements indiqués comme supprimés sont supprimés. On ne tient pas compte d'une tentative de modification d'un enregistrement qui existe déjà, ni d'une tentative de modification ou de suppression d'un enregistrement qui n'existe pas. Pour la liste des capacités d'application, on suit la procédure décrite pour le cas d'une mise à jour complète. Dans ce cas cependant, les nouvelles listes de capacités d'application ne sont recalculées que pour les ensembles d'entités APE qui ont signalé un changement.

Dans les deux cas précédents, les informations mises à jour pour le répertoire de conférence et pour chaque session du protocole d'application contenue dans le répertoire d'application comportent un numéro d'instance. Si les informations de conférence ont été modifiées en raison de la mise à jour reçue, le numéro d'instance des informations de conférence doit être augmenté d'une unité modulo  $2^{16}$ . De même, pour chaque session du protocole d'application pour laquelle une information quelconque a été modifiée (soit des enregistrements d'application soit la liste des capacités d'application), ce numéro d'instance doit être augmenté d'une unité modulo  $2^{16}$ . Si de multiples modifications sont apportées, elles doivent être réexpédiées vers d'autres nœuds au titre d'une mise à jour unique; ces modifications peuvent être comptées comme un seul incrément du numéro d'instance. Les numéros d'instance sont tenus à jour localement et correspondent au sous-ensemble local du répertoire, c'est-à-dire à ce nœud plus les nœuds immédiatement inférieurs à celui-ci dans la hiérarchie de connexion. Dans le cas du fournisseur GCC sommital, les numéros d'instance s'appliquent aux répertoires de conférence et d'application qui sont diffusés à tous les nœuds et signalés dans la primitive GCC-Application-Roster-Report. Les informations de conférence comportent également un indicateur signalant si des nœuds ont été ajoutés et/ou supprimés depuis la dernière instance. De même, pour chaque session du protocole d'application, un indicateur signale si des nœuds ont été ajoutés et/ou supprimés depuis la dernière instance des informations pour cet ensemble d'entités APE. Dans ce cas, les nœuds ajoutés ou supprimés peuvent

indiquer qu'une entité APE a été, respectivement, inscrite ou désinscrite à un nœud donné et non nécessairement que celui-ci a été entièrement ajouté ou supprimé de la conférence.

Une fois que les informations de répertoire de conférence et/ou d'application ont été recalculées, un fournisseur GCC non sommital doit produire une unité PDU RosterUpdateIndication qui doit ensuite être envoyée au fournisseur GCC immédiatement supérieur dans la hiérarchie de connexion. Le format des informations contenus dans cette unité PDU peut être soit une mise à jour complète, une mise à jour de certains ensembles d'entités APE et/ou des informations de conférence, ou des mises à jour individuelles. La sélection du format d'expédition des informations de mise à jour est laissée aux soins du fournisseur GCC. Il est préféré que celui-ci choisisse un format qui minimise la longueur de l'unité PDU et donc le temps de transmission nécessaire.

### **8.3.3.6 Distribution des répertoires de conférence et d'application**

Lorsque le fournisseur GCC sommital a reçu une unité RosterUpdateIndication qui contient des informations de répertoire pour nœud soit conventionnel soit compté, il doit modifier les informations de répertoire de conférence et d'application qu'il conserve (et qui, dans ce cas, sont les répertoires de conférence et d'application complets plutôt que des sous-ensembles), de manière identique à celle qui a été décrite pour une unité MCU intermédiaire mettant à jour son sous-ensemble du répertoire. Une fois que le répertoire complet a été mis à jour, le fournisseur GCC sommital doit renvoyer l'information de répertoire mise à jour par diffusion soit à chaque nœud de la conférence soit aux nœuds conventionnels participant à la conférence. Le choix dépend des informations de répertoire qui ont été affectées à la suite de la réception de l'unité RosterUpdateIndication. Toute modification à la liste des enregistrements nodaux conventionnels, au répertoire d'application dû à l'association d'une entité APE avec un nœud conventionnel, ou à une liste de capacités, doit être communiquée à tous les nœuds de la conférence. A cette fin, le fournisseur GCC doit émettre une demande MCS-Send-Data spécifiant le canal GCC-Conventional-Broadcast-Channel en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU dans le champ de données. Les modifications à la liste des enregistrements nodaux comptés ou à un répertoire d'application dû à l'association d'une entité APE avec un nœud compté ne sont communiquées qu'aux nœuds conventionnels. A cette fin, le fournisseur GCC doit émettre une demande MCS-Send-Data spécifiant le canal GCC-Counted-Broadcast-Channel en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU dans le champ de données.

L'unité RosterUpdateIndication diffusée par le fournisseur GCC sommital ne comporte que les informations de répertoire spécifiques qui ont été modifiées en raison de la réception de l'unité RosterUpdateIndication originale. Il peut s'agir de tout ou partie des éléments suivants: un ou plusieurs enregistrements nodaux, un ou plusieurs enregistrements d'application, et de toute liste de capacités d'application qui peut avoir été altérée à cause du traitement de l'unité RosterUpdateIndication reçue. Celle-ci ne doit pas contenir d'informations déjà diffusées au reste de la conférence ni de mise à jour complète.

Dès qu'il reçoit une indication de mise à jour de répertoire sur le canal de diffusion GCC-Conventional-Broadcast-Channel ou GCC-Counted-Broadcast-Channel, chaque fournisseur GCC doit produire une indication GCC-Conference-Roster-Report et doit l'envoyer au point GCCSAP de contrôle si l'unité PDU indiquait une quelconque modification du répertoire de conférence. Si l'unité PDU reçue a indiqué une modification à une partie ou à la totalité des informations du répertoire d'application, le fournisseur GCC doit produire une série d'indications GCC-Application-Roster-Report. Il doit ensuite les envoyer au point GCCSAP associé à chaque entité APE inscrite qui correspond à une session du protocole d'application pour laquelle une mise à jour du répertoire d'application a été reçue dans l'unité PDU. Le fournisseur GCC peut aussi envoyer des indications GCC-Application-Roster-Report à d'autres points GCCSAP, bien que la nécessité de procéder ainsi soit considérée comme une question locale, hors du domaine d'application de la présente Recommandation. Seule la portion du répertoire d'application de

conférence qui est associée à la clé de session pour cette entité APE doit être incluse dans la primitive correspondante. Le fournisseur GCC peut décider d'inclure des portions du répertoire correspondant à d'autres clés de session, bien que la nécessité de procéder ainsi soit considérée comme une question locale, hors du domaine d'application de la présente Recommandation. Dans le cas d'une entité APE inscrite sans indicateur d'activité ni identificateur de session, le fournisseur GCC doit envoyer une indication GCC-Application-Roster-Report au point GCCSAP correspondant, pour les portions du répertoire qui correspondent à une session du protocole d'application ayant le même protocole d'application de base que l'entité APE. Le fournisseur GCC doit également envoyer une indication GCC-Application-Roster-Report au point GCCSAP de contrôle, y compris des mises à jour du répertoire d'application pour tous les ensembles d'entités APE homologues qui ont été signalées comme ayant été modifiées par l'unité PDU reçue.

### **8.3.3.7 Nœuds quittant une conférence**

Dès qu'ils reçoivent une indication MCS-Detach-User, les nœuds d'unité MCU qui ont des nœuds inférieurs dans la hiérarchie de connexion doivent vérifier l'identificateur d'utilisateur indiqué et déterminer s'il correspond à l'identificateur nodal d'un nœud immédiatement inférieur dans la hiérarchie de connexion. Si c'est le cas, le nœud MCU doit supprimer toutes les entrées correspondant à ce nœud ainsi que tous les nœuds réputés inférieurs à ce nœud, d'après son sous-ensemble des répertoires de conférence et d'application. Il doit ensuite recalculer les listes de capacités d'application pour tous les ensembles d'entités APE homologues, de la façon décrite au § 8.3.4. Une fois que le sous-ensemble de répertoire a été complètement mis à jour en fonction du nœud sortant, ce fournisseur GCC doit suivre la procédure décrite au § 8.3.3.6 pour propager cette mise à jour vers les autres nœuds de la conférence.

Dans un nœud qui s'est déconnecté d'une conférence (soit par déconnexion, par terminaison de la conférence ou par rejet de la conférence), le fournisseur GCC doit émettre une indication GCC-Permission-To-Enroll qui annule l'autorisation d'inscription à la conférence correspondante. Le fournisseur GCC doit envoyer cette indication à tous les points GCCSAP qui ne sont pas des points de contrôle.

Les enregistrements de nœud ou d'application qui sont associés à des identificateurs de nœuds anonymes ne doivent pas apparaître dans un répertoire de conférence ou d'application nodal; on peut donc ne pas en tenir compte.

### **8.3.3.8 Acquisition d'une mise à jour complète de répertoire en provenance d'un nœud parent**

A tout moment au cours d'une conférence, un nœud peut demander à son nœud parent une mise à jour complète du répertoire, en lui envoyant une unité RosterRefreshRequest. A cette fin, le nœud doit émettre une demande MCS-Send-Data spécifiant l'identificateur du nœud supérieur en tant qu'identificateur de canal, spécifiant une priorité élevée des données et incluant l'unité PDU RosterRefreshRequest dans le champ de données.

Normalement, une demande RosterRefreshRequest sera faite aux conditions suivantes: soit le nœud vient d'entrer dans la conférence et a besoin d'une mise à jour complète du répertoire pour synchroniser son répertoire d'application de conférence avec d'autres nœuds de la conférence, soit un fournisseur GCC a reçu une interrogation de répertoire à un nœud anonyme qui ne conserve pas une copie locale du répertoire d'application de conférence. Dans un cas comme dans l'autre, une fois que l'unité PDU RosterRefreshRequest est reçue par le nœud parent, elle doit soit traiter la demande (si elle a accès aux répertoires requis), soit elle doit faire suivre la demande vers le nœud suivant dans la hiérarchie de connexion, au moyen d'une autre RosterRefreshRequest.

Lorsqu'on atteint un nœud qui contient l'information requise, ce nœud doit envoyer directement au nœud demandeur une indication RosterUpdateIndication contenant une mise à jour complète de répertoire. A cette fin, le nœud émet une demande MCS-Send-Data spécifiant l'identificateur du nœud d'origine (contenu dans la demande RosterRefreshRequest) en tant qu'identificateur de canal,

spécifiant une priorité élevée des données et incluant l'unité PDU RosterUpdateIndication dans le champ de données.

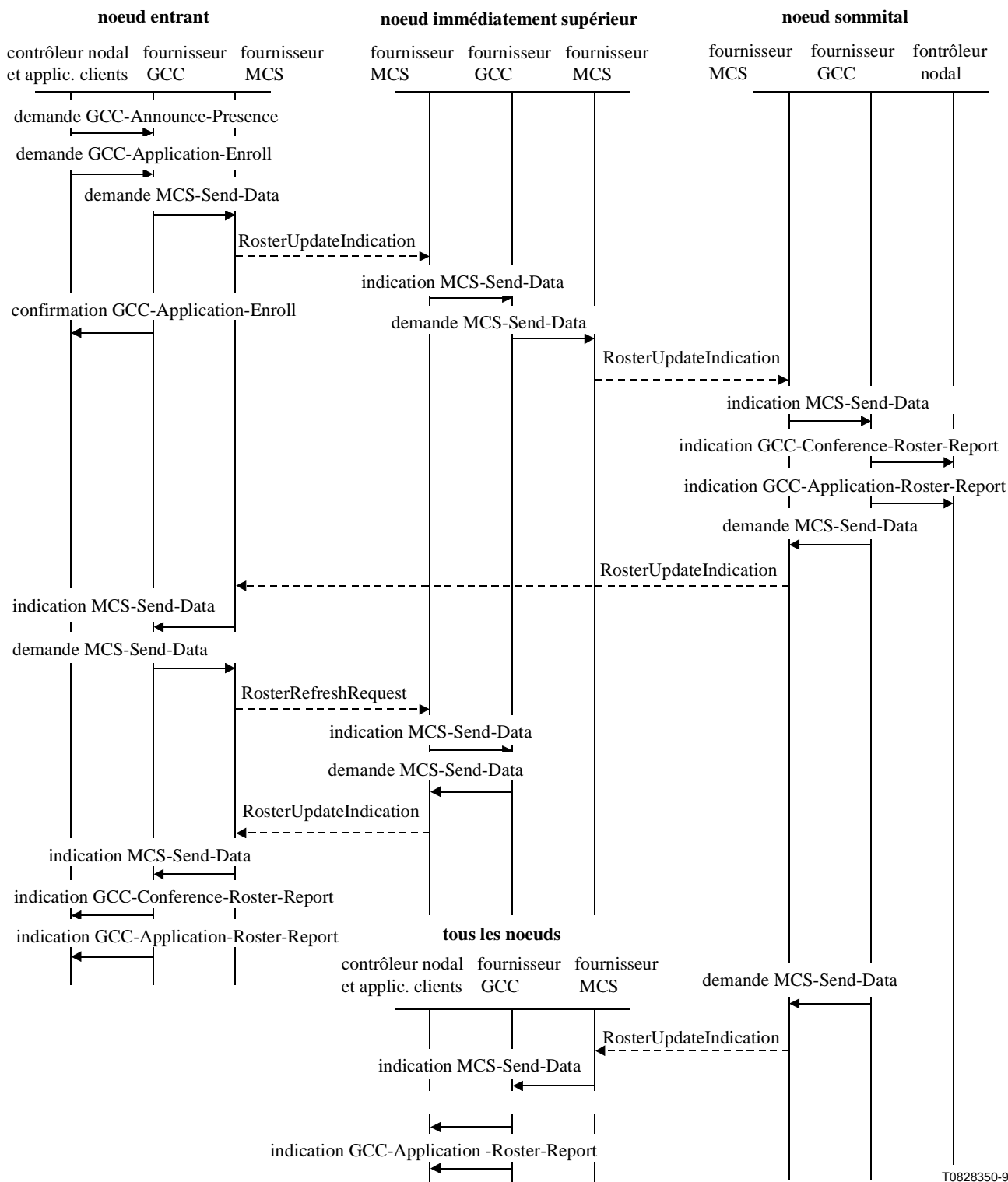
Comme mentionné ci-dessus, l'unité PDU RosterUpdateIndication renvoyée au nœud qui a formulé la demande RosterRefreshRequest originale aura une apparence différente selon la catégorie nodale à laquelle le nœud d'origine appartient. Par exemple, si c'est un nœud conventionnel qui a lancé la demande, la mise à jour de répertoire contiendra des enregistrements pour nœuds aussi bien conventionnels que comptés. Un nœud conventionnel parent doit donc être atteint avant que la demande puisse être traitée (car seuls les nœuds conventionnels conservent une liste de nœuds comptés). Si le nœud demandeur est soit anonyme soit compté, la demande peut être traitée par le premier nœud qui contient le répertoire de nœud conventionnel. Il appartient au nœud qui traite la demande RosterRefreshRequest de séparer les enregistrements qui ne doivent pas être reçus par certaines catégories nodales.

La demande RosterRefreshRequest se compose des éléments suivants: un champ d'identificateur nodal qui contient l'identificateur du nœud émetteur de la demande; un champ de catégorie nodale qui contient la catégorie du nœud émetteur de la demande; une valeur booléenne spécifiant qu'une mise à jour de répertoire complète doit être acheminée; une valeur booléenne spécifiant que le répertoire de conférence doit être envoyé; une liste de session énumérant les répertoires de session spécifiques qui sont demandés; et une liste d'application qui énumère les protocoles d'application au sujet desquels un nœud cherche à obtenir des informations. Si l'opérateur booléen de mise à jour complète est mis à la valeur TRUE, on ne tient pas compte du champ sendConferenceRoster, ni du champ sessionList, ni du champ applicationList. Si l'opérateur booléen de mise à jour complète est mis à la valeur FALSE, ces trois champs définissent les informations qui doivent être acheminées vers le nœud qui a émis la demande. Une unité RosterRefreshRequest peut donc demander un ou plusieurs répertoires d'application spécifiques ou seulement le répertoire de conférence, ainsi qu'une mise à jour complète de répertoire. Le mode de traitement d'une demande partielle est identique au processus décrit ci-dessus, sauf que seule l'information requise est renvoyée.

Il existe des contraintes de temporisation à respecter lorsqu'un nœud tente de joindre et de synchroniser son répertoire d'application de conférence avec d'autres nœuds de la conférence. Un nœud a la possibilité de recevoir des indications de mise à jour de répertoire avant d'avoir reçu, d'un nœud parent, sa première mise à jour de répertoire complète. Si ces mises à jour contiennent un numéro d'instance de répertoire qui précède le numéro d'instance de répertoire associé à la mise à jour de répertoire complète, celle-ci peut être rejetée. Si ces numéros d'instance sont supérieurs au numéro d'instance associé à la mise à jour complète, ils doivent être traités. Il est donc nécessaire qu'un nœud entrant mémorise toutes les indications de mise à jour de répertoire entrantes qu'il a reçues du fournisseur GCC sommital, avant de pouvoir recevoir la réponse à sa demande initiale de mise à jour de répertoire. Toutes les indications de mise à jour de répertoire contiennent un numéro d'instance qui est augmenté séquentiellement chaque fois que le fournisseur sommital détecte une modification de répertoire. Ce numéro d'instance de répertoire est utilisé pour déterminer l'ordre dans lequel une indication de mise à jour de répertoire doit être traitée par un nœud entrant. Comme examiné ci-dessus, cela peut être particulièrement complexe pour des nœuds conventionnels qui participent à une conférence pouvant comporter aussi bien des nœuds conventionnels que des nœuds comptés. Les indications de mise à jour de répertoire pourront en effet être reçues aussi bien par le canal GCC-Conventional-Broadcast-Channel que par le canal GCC-Counted-Broadcast-Channel. Il appartient au nœud conventionnel de traiter les indications de mise à jour de répertoire dans l'ordre correct, sur la base de la connaissance préalable du fait que les numéros d'instance de répertoire augmentent en séquence.

### 8.3.3.9 Exemple de mise à jour de répertoire

La Figure 8-6 montre le protocole relatif au répertoire qui se déroule lorsqu'un nouveau nœud conventionnel entre dans une conférence. Une seule application d'inscription de client est représentée activement au nœud entrant. Une fois que le fournisseur GCC a reçu une demande GCC-Announce-Presence-Request et une demande GCC-Application-Enroll, une mise à jour du répertoire d'application de conférence est envoyée au nœud immédiatement supérieur. Dans cet exemple, le nœud entrant est dans la troisième couche de la hiérarchie de connexion. Il envoie l'indication de mise à jour au nœud immédiatement supérieur, qui ensuite formate une indication de mise à jour et l'envoie au nœud immédiatement supérieur, qui dans cet exemple, est le nœud sommital de la hiérarchie. Le nœud sommital assemble le répertoire d'application complet et en diffuse les portions associées à la mise à jour vers tous les nœuds de la conférence. Cette opération donne lieu à l'envoi d'une primitive GCC-Conference-Report et GCC-Application-Report à tous les nœuds de la conférence, sauf le nœud conventionnel entrant. Celui-ci, par contre, doit recevoir le répertoire complet de son nœud parent, en émettant une demande RosterRefreshRequest. Une fois la réponse à cette demande reçue, une primitive GCC-Conference-Report et GCC-Application-Report peut être émise au nœud entrant.



T0828350-98

**Figure 8-6 – Autre exemple de mise à jour du répertoire d'application**

### 8.3.4 Condensation des listes de capacités d'application

Tout fournisseur GCC est responsable, pour toutes les sessions de protocole d'application, de constituer la liste de capacités d'application condensée correspondant à ses nœuds conventionnels subordonnés et à toutes les entités de protocole d'application homologues inscrites sur le nœud local (si celui-ci est conventionnel). Comme les nœuds directement subordonnés dans la hiérarchie de connexion gèrent la condensation de leurs propres capacités, le fournisseur GCC local n'a besoin de

connaître que les listes de capacités d'application condensées de ces nœuds plutôt que les listes de capacités d'application de tous les nœuds subordonnés. Le fournisseur GCC produira une liste de capacités d'application condensée à partir de ces listes de capacités d'application condensées et des listes de capacités d'application des entités de protocole d'application homologues du nœud local contenues dans le répertoire d'application. Il utilisera pour ce faire la procédure suivante:

- déterminer la classe de capacité pour chaque capacité de chaque liste;
- pour toute classe, la valeur du paramètre de comptage de chaque entrée de la liste de capacités d'application condensée recevra la somme des comptages indiqués dans l'entrée correspondante de toutes les listes de capacités d'application;
- pour les classes "minimum sans signe" et "maximum sans signe", la nouvelle entrée doit également recevoir la valeur minimale ou maximale indiquée dans l'entrée correspondante de toutes les listes de capacités d'application contenues dans l'ensemble communiqué.

### **8.3.5 Interrogation des répertoires de conférence et d'application**

Lorsqu'il reçoit une primitive de demande GCC-Application-Roster-Inquire (interrogation de répertoire d'application), un fournisseur GCC qui prend en charge cette primitive peut produire une primitive de confirmation GCC-Application-Roster-Inquire et l'envoyer en réponse au point d'accès GCCSAP du demandeur. Le contenu de la primitive de confirmation est produit à partir de la base de données du répertoire d'application de conférence gérée localement et contiendra exclusivement les entrées de répertoire pour lesquelles la clé de session donnée dans la demande correspond exactement, pour une longueur de la clé égale à celle de la demande, à la clé de session contenue dans l'entrée.

Lorsqu'il reçoit la primitive de demande GCC-Conference-Roster-Inquire, un fournisseur GCC qui prend en charge cette primitive répondra en produisant une primitive de confirmation GCC-Conference-Roster-Inquire et l'enverra au point d'accès GCCSAP du demandeur. Le contenu de la primitive de confirmation est produit à partir de la base de données locale de répertoire de conférence en prenant en compte tous les nœuds connus comme faisant partie de la conférence.

Un fournisseur GCC situé dans un nœud anonyme, qui reçoit une interrogation de répertoire ne possède peut-être pas une base de données locale de répertoire d'application de conférence car cela n'est obligatoire pour les nœuds anonymes. Les nœuds de ce type peuvent répondre à l'interrogation de répertoire soit en renvoyant une confirmation d'échec au point GCCSAP demandeur soit en envoyant à son nœud parent une demande de mise à jour de répertoire afin d'obtenir les informations requises. Cette demande produira finalement une indication de type RosterReportIndication qui comportera les informations requises. A ce stade, le fournisseur GCC répond en produisant une primitive de confirmation GCC-Application-Roster-Inquire et l'envoie au point GCCSAP du demandeur. On notera que les nœuds conventionnels et comptés sont tenus de posséder une base de données de répertoire d'application de conférence, de façon que les interrogations de répertoire reçues par ces nœuds puissent être traitées immédiatement.

### **8.3.6 Invocation à distance d'une entité de protocole d'application**

Lorsqu'il reçoit, du point de contrôle GCCSAP ou d'un point d'accès GCCSAP ordinaire, la primitive de demande GCC-Application-Invoke (invocation d'application), un fournisseur GCC qui prend en charge cette primitive diffusera une unité PDU ApplicationInvokeIndication (indication d'invocation d'application) à l'ensemble des nœuds de la conférence mentionnée. Pour ce faire, il émettra une primitive de demande MCS-Send-Data ou MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant l'identificateur des deux canaux GCC-Broadcast-Channel (canal de diffusion pour prendre en charge les nœuds appliquant l'ancien protocole) et GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. En variante, le fournisseur GCC peut émettre une ou plusieurs demandes MCS-Send-Data (envoi de données) spécifiant l'identificateur de l'un des nœuds destinataires listés, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Dans ce cas, il peut



laisser vide le paramètre "liste de nœuds destinataires" de l'unité PDU. Le Tableau 8-36 donne le contenu de l'unité PDU ApplicationInvokeIndication. Le fournisseur GCC produira ensuite une primitive de confirmation GCC-Application-Invoke pour indiquer que l'opération demandée a réussi et l'enverra au point d'accès GCCSAP du demandeur.

**Tableau 8-36 – GCCPDU ApplicationInvokeIndication**

Contenu	Origine	Destination
Liste des entités de protocole d'application	Primitive de demande	Primitive d'indication
Liste de nœuds destinataires ou NULL (néant)	Primitive de demande	Fournisseurs GCC destinataires

Lorsqu'il reçoit une unité PDU ApplicationInvokeIndication, un fournisseur GCC qui prend en charge la primitive GCC-Application-Invoke déterminera en premier lieu si le nœud local figure sur la liste de nœuds destinataires. Si la liste est NULL (vide) ou si l'identificateur du nœud local y est présent, le fournisseur GCC peut produire une primitive d'indication GCC-Application-Invoke et l'envoyer au point de contrôle GCCSAP. Le paramètre "nœud demandeur" est rempli avec le champ "identificateur d'utilisateur demandeur" de la primitive d'indication MCS-Send-Data ou MCS-Uniform-Send-Data reçue. Aucune autre action n'est exécutée si le nœud local ne figure pas explicitement ou implicitement sur la liste de nœuds destinataires.

#### **8.4 Le référentiel d'application**

Le référentiel d'application est une base de données active résidant au niveau du fournisseur GCC sommital d'une conférence. Tout nœud MCU prendra en charge l'ensemble complet de services de référentiel qui mettent en œuvre cette base de données, alors qu'un nœud terminal peut choisir de ne prendre en charge que ceux des services de référentiel exigés par les protocoles d'application pris en charge par ce terminal.

Lors de la création d'une conférence, la base de données du référentiel située au niveau du fournisseur GCC sommital doit être initialisée dans un état où les entrées de référentiel sont vides et où aucune entrée n'est supervisée. Cela sera fait avant le branchement du fournisseur GCC sommital sur le canal correspondant à son identificateur d'utilisateur.

Une entrée non vide du référentiel contient en cours de conférence les informations suivantes:

- la clé de référentiel identifiant l'entrée;
- le type d'information stockée: identificateur de canal, identificateur de jeton ou paramètre;
- l'état de supervision activée ou non;
- le contenu de l'entrée, valeur effective de l'identificateur de canal, de l'identificateur de jeton ou du paramètre;
- le propriétaire de l'entrée, qui peut être inexistant ou indiqué par la combinaison de l'identificateur de nœud et de l'identificateur d'entité de l'entité de protocole d'application propriétaire;
- pour les entrées du type "paramètre" uniquement, le droit de modification de l'entrée: propriétaire, session ou public.

Le fournisseur GCC sommital traitera des demandes multiples de référentiel issues d'un nœud donné strictement dans leur ordre de réception. La succession des unités PDU de réponse renvoyées à un nœud demandeur correspondra à l'ordre d'arrivée des demandes en provenance de ce nœud.

Au niveau du nœud demandeur, l'ordre sera également préservé entre l'envoi des primitives de demande et la transmission des unités PDU ainsi qu'entre la réception des unités PDU et les primitives de confirmation correspondantes.

Seuls les nœuds conventionnels sont autorisés à ajouter, supprimer ou modifier une entrée dans la base de données du référentiel. Celui-ci est en lecture seulement pour toutes les autres catégories nodales. Les nœuds anonymes et comptés qui ne semblent pas inscrits à la conférence peuvent donc avoir encore accès à la base de données du référentiel.

#### 8.4.1 Enregistrement d'un canal

Lorsqu'il reçoit une primitive de demande GCC-Registry-Register-Channel (enregistrement d'un canal dans le référentiel), un fournisseur GCC situé dans un nœud conventionnel enverra une unité PDU RegistryRegisterChannelRequest (demande d'enregistrement d'un canal dans le référentiel) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryRegisterChannelRequest est donné par le Tableau 8-37. Une demande GCC-Registry-Register-Channel, émise par un nœud compté ou anonyme, sera ignorée par le fournisseur GCC.

**Tableau 8-37 – GCCPDU RegistryRegisterChannelRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Clé	Primitive de demande	Fournisseur GCC sommital
Identificateur de canal	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit l'unité PDU RegistryRegisterChannelRequest, le fournisseur GCC sommital vérifiera en premier lieu que l'identificateur d'utilisateur de l'émetteur de la primitive d'indication MCS-Send-Data reçue et l'identificateur d'entité contenu dans l'unité PDU correspondent exactement à l'identificateur de nœud et à l'identificateur d'une entité APE d'un nœud conventionnel catalogué dans le répertoire d'application. Dans ce cas, il tentera d'enregistrer le canal en créant une entrée de référentiel appropriée. Il vérifiera d'abord si l'entrée correspondant à la clé indiquée existe déjà. Si l'entrée n'existe pas, le fournisseur GCC sommital crée une entrée pour cette clé et l'initialise en fournissant comme propriétaire l'identificateur de nœud de l'émetteur figurant dans la primitive d'indication MCS-Send-Data reçue, associé à l'identificateur d'entité figurant dans l'unité PDU. Il indique également comme type d'entrée un identificateur de canal et comme valeur celle de l'identificateur de canal fourni dans l'unité PDU.

Le fournisseur GCC sommital signale ensuite au nœud demandeur que le canal a été enregistré correctement. Pour ce faire, il enverra une unité PDU RegistryResponse (réponse de référentiel) spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryResponse est donné dans le Tableau 8-38. Un paramètre "résultat positif" est fourni en cas de réussite de l'action. Le champ "droits de modification" n'est pas rempli (ce champ n'étant utilisé que pour des entrées du type "paramètre").

Si l'entrée de référentiel existe déjà, si le demandeur n'est pas valide car il n'apparaît pas dans le répertoire d'application, si le nœud demandeur n'est pas un nœud conventionnel ou si l'entrée de référentiel ne peut pas être créée à cause d'une pénurie de ressources, le référentiel n'est pas modifié et une unité PDU RegistryResponse est renvoyée comme ci-dessus, mais avec un paramètre résultat négatif indiquant la cause de l'erreur. Dans ce cas, l'élément de référentiel renvoyé dans l'unité PDU RegistryResponse contient la valeur de l'entrée de référentiel avant la tentative de mise à jour.

**Tableau 8-38 – GCCPDU RegistryResponse**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC sommital	Fournisseur GCC destinataire
Type de primitive (register-channel, assign-token, set-parameter, retrieve-entry, delete-entry, monitor-entry)	Fournisseur GCC sommital	Fournisseur GCC destinataire
Clé	Fournisseur GCC sommital	Primitive de confirmation
Élément de référentiel	Fournisseur GCC sommital	Primitive de confirmation
Propriétaire	Fournisseur GCC sommital	Primitive de confirmation
Droit de modification (optionnel)	Fournisseur GCC sommital	Primitive de confirmation
Résultat	Fournisseur GCC sommital	Primitive de confirmation

En cas de réussite de la modification d'une entrée de référentiel existante, le fournisseur GCC sommital vérifie si l'entrée de référentiel est supervisée. Si ce n'est pas le cas, le fournisseur GCC sommital n'exécute aucune autre action. Si l'entrée de référentiel est supervisée, le fournisseur GCC sommital envoie une unité PDU RegistryMonitorEntryIndication (indication d'entrée de référentiel supervisée) à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant l'identificateur du canal GCC-Broadcast-Channel (canal de diffusion), spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryMonitorEntryIndication est donné par le Tableau 8-44.

Lorsqu'il reçoit une unité PDU RegistryResponse de ce type, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Register-Channel, indiquant si la demande a réussi ou non compte tenu du paramètre résultat de l'unité PDU RegistryResponse. Il enverra cette confirmation au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC ne connaît pas d'entité de protocole d'application inscrite avec un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

#### **8.4.2 Attribution d'un jeton**

Lorsqu'il reçoit une primitive de demande GCC-Registry-Assign-Token (attribution d'un jeton par le référentiel), un fournisseur GCC situé dans un nœud conventionnel enverra une unité PDU RegistryAssignTokenRequest (demande d'attribution d'un jeton par le référentiel) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryAssignTokenRequest est donné dans le Tableau 8-39. Une demande GCC-Registry-Assign-Token, émise par un nœud compté ou anonyme, sera ignorée par le fournisseur GCC.

**Tableau 8-39 – GCCPDU RegistryAssignTokenRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Clé	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit l'unité PDU RegistryAssignTokenRequest, le fournisseur GCC sommital vérifiera en premier lieu que l'identificateur d'utilisateur de l'émetteur contenu dans la primitive d'indication MCS-Send-Data reçue et l'identificateur d'entité contenu dans l'unité PDU correspondent

exactement à l'identificateur de nœud et à l'identificateur d'entité d'une application dans un nœud conventionnel actuellement catalogué dans le répertoire d'application. Dans ce cas, il tentera d'attribuer un jeton en créant une entrée de référentiel appropriée. Il vérifie d'abord si l'entrée correspondant à la clé donnée existe déjà. Si l'entrée de référentiel n'existe pas, le fournisseur GCC sommital attribue un nouvel identificateur de jeton à partir de l'espace d'identificateurs de jeton dynamique allant de 16 384 à 65 535. Le fournisseur GCC sommital crée ensuite une entrée pour cette clé et l'initialise en fournissant comme propriétaire l'identificateur de nœud, indiqué par l'identificateur d'utilisateur de l'émetteur figurant dans la primitive d'indication MCS-Send-Data reçue, associé à l'identificateur d'entité figurant dans l'unité PDU. Il indique également comme type d'entrée un identificateur de jeton et comme valeur l'identificateur de jeton qui vient d'être attribué.

Le fournisseur GCC sommital signale ensuite au nœud demandeur que le jeton a été attribué correctement. Pour ce faire, il enverra une unité PDU RegistryResponse (réponse de référentiel) spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryResponse est donné dans le Tableau 8-38. Un paramètre "résultat positif" est fourni en cas de réussite de l'action. Le champ "droits de modification" n'est pas rempli, ce champ n'étant utilisé que pour des entrées du type "paramètre".

Si l'entrée de référentiel existe déjà, si le demandeur n'est pas valide car il n'apparaît pas dans le répertoire d'application, si le nœud demandeur n'est pas un nœud conventionnel, si l'entrée de référentiel ne peut pas être créée à cause d'une pénurie de ressources ou s'il n'y a plus d'identificateur dynamique de jeton disponible, le référentiel n'est pas modifié et une unité PDU RegistryResponse est renvoyée comme ci-dessus, mais avec un paramètre résultat négatif indiquant la cause de l'erreur. Dans ce cas, l'élément de référentiel renvoyé dans l'unité PDU RegistryResponse contient la valeur de l'entrée de référentiel avant la tentative de mise à jour.

En cas de réussite de la modification d'une entrée de référentiel existante, le fournisseur GCC sommital vérifie si l'entrée de référentiel est supervisée. Si ce n'est pas le cas, le fournisseur GCC sommital n'exécute aucune action supplémentaire. Si l'entrée de référentiel est supervisée, le fournisseur GCC sommital envoie une unité PDU RegistryMonitorEntryIndication (indication d'entrée de référentiel supervisée) à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant l'identificateur du canal GCC-Broadcast-Channel (canal de diffusion), spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryMonitorEntryIndication est donné par le Tableau 8-44.

Lorsqu'il reçoit une unité PDU RegistryResponse de ce type, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Assign-Token, indiquant si la demande a réussi ou non compte tenu du paramètre résultat de l'unité PDU RegistryResponse. Il enverra cette confirmation au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC ne connaît pas d'entité de protocole d'application inscrite avec un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

### **8.4.3 Positionnement d'un paramètre**

Lorsqu'il reçoit une primitive de demande GCC-Registry-Set-Parameter (positionnement de paramètre par le référentiel), un fournisseur GCC situé dans un nœud conventionnel enverra une unité PDU RegistrySetParameterRequest (demande de positionnement de paramètre par le référentiel) au fournisseur GCC sommital. Pour ce faire il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistrySetParameterRequest est donné dans le Tableau 8-40. Une demande GCC-Registry-Set-Parameter, émise par un nœud compté ou anonyme, sera ignorée par le fournisseur GCC.

**Tableau 8-40 – GCCPDU RegistrySetParameterRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Clé	Primitive de demande	Fournisseur GCC sommital
Paramètre	Primitive de demande	Fournisseur GCC sommital
Droit de modification (optionnel)	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit l'unité PDU RegistrySetParameterRequest, le fournisseur GCC sommital vérifiera en premier lieu que l'identificateur d'utilisateur de l'émetteur contenu dans la primitive d'indication MCS-Send-Data reçue et l'identificateur d'entité contenu dans l'unité PDU correspondent exactement à l'identificateur de nœud et à l'identificateur d'entité APE dans un nœud conventionnel catalogué dans le répertoire d'application. Dans ce cas, il tentera de mettre à jour une entrée existante ou de créer une nouvelle entrée. Il vérifie d'abord si l'entrée correspondant à la clé fournie existe déjà et détermine, le cas échéant, le type de l'entrée, le propriétaire et le droit de modification en vigueur. Si l'entrée de référentiel n'existe pas, le fournisseur GCC sommital crée une entrée pour cette clé et l'initialise en fournissant comme propriétaire l'identificateur de nœud indiqué par l'identificateur d'utilisateur de l'émetteur figurant dans la primitive d'indication MCS-Send-Data reçue, associé à l'identificateur d'entité figurant dans l'unité PDU. Il indique également comme type d'entrée un paramètre et comme valeur celle du paramètre figurant dans l'unité PDU. Si l'entrée de référentiel existe déjà et si elle est du type "paramètre", le fournisseur GCC vérifiera si le demandeur a le droit de la modifier. Si le propriétaire a le droit de modification, l'entrée de référentiel reçoit la valeur de paramètre fournie dans l'unité PDU. Si l'entrée n'avait pas de propriétaire, elle est modifiée de manière à indiquer le nouveau propriétaire.

Les règles qui suivent servent à déterminer si le propriétaire possède le droit de modification. Si l'attribut "droits de modification" de l'entrée est positionné sur "propriétaire", le demandeur doit être le propriétaire de l'entrée ou celle-ci doit être sans propriétaire pour que la demande réussisse. Le demandeur est défini par l'identificateur d'entité et l'identificateur de nœud contenus dans le champ "identificateur d'utilisateur demandeur" de la primitive d'indication MCS-Send-Data reçue. Si l'attribut "droits de modification" de l'entrée est positionné sur "session", le demandeur doit faire partie de la même session de protocole d'application pour que la demande réussisse. Cela est vérifié en recherchant l'entrée correspondant au demandeur dans le répertoire d'application en vigueur et en examinant la clé de session qui doit être identique à celle du propriétaire. Si l'attribut "droits de modification" de l'entrée est positionné sur "public", il n'y a pas de contraintes concernant le demandeur pour que sa demande soit suivie d'effet, sauf que le nœud doit être de type conventionnel.

L'état du droit de modification est examiné lorsqu'une entrée du type paramètre est créée. Si le paramètre "droits de modification" est fourni dans l'unité PDU de demande qui est à l'origine de la création de l'entrée, le type de modification est positionné sur la valeur indiquée. Dans le cas contraire, la valeur "public" est prise par défaut. L'état du droit de modification ne change pas si le droit de modification n'est pas fourni lors d'une demande de positionnement de paramètre. Le droit de modification est remplacé par le droit indiqué par l'unité PDU si le demandeur est le propriétaire en vigueur ou si l'entrée n'a pas de propriétaire. Dans le cas contraire, le droit de modification n'est pas modifié mais l'opération de positionnement du paramètre sera exécutée normalement et le résultat ne sera pas affecté.

Dans chacun des cas précédents, le fournisseur GCC sommital indique ensuite que le paramètre a été modifié correctement en envoyant une unité PDU RegistryResponse (réponse de référentiel) au nœud demandeur. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU

RegistryResponse est donné par le Tableau 8-38. En cas de réussite de l'action, le paramètre résultat indique le succès. Le paramètre "droits de modification" sera présent dans l'unité PDU de réponse.

Si l'entrée de référentiel n'a pas pu être créée par suite de manque de ressources disponibles, si le nœud demandeur n'est pas un nœud conventionnel, si l'entrée existe déjà mais que le demandeur ne dispose pas des droits de modification d'une entrée existante, si le demandeur n'est pas valide parce qu'il ne figure pas dans le répertoire d'application ou si l'entrée est du type paramètre, le référentiel n'est pas modifié et une unité PDU RegistryResponse est renvoyée avec un paramètre résultat négatif indiquant la cause de l'échec. Dans ce cas le paramètre "élément de référentiel" de l'unité PDU renvoyée contient la valeur de l'entrée avant la tentative de modification. En cas de droits de modification insuffisants, le paramètre résultat sera "appartient à un autre".

En cas de réussite de la modification d'une entrée existante, le fournisseur GCC sommital examine si l'entrée de référentiel est dans l'état supervisé. Si ce n'est pas le cas, aucune autre action n'est exécutée par le fournisseur GCC sommital. Si l'entrée est supervisée, le fournisseur GCC sommital envoie une unité PDU RegistryMonitorEntryIndication (indication de supervision d'entrée de référentiel) à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data spécifiant l'identificateur de canal GCC-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryMonitorEntryIndication est donné par le Tableau 8-44.

Lorsqu'il reçoit une unité PDU RegistryResponse de ce type, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Set-Parameter indiquant si la demande a réussi ou non conformément à l'indication du paramètre résultat de l'unité PDU RegistryResponse qu'il enverra au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC sommital ne connaît pas d'entité de protocole d'application inscrite actuellement ayant un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

#### 8.4.4 Recherche d'une entrée

Lorsqu'il reçoit une primitive de demande GCC-Registry-Retrieve-Entry (recherche d'entrée de référentiel), un fournisseur GCC enverra une unité PDU RegistryRetrieveEntryRequest (demande d'entrée de référentiel) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryRetrieveEntryRequest est donné dans le Tableau 8-41. Une telle unité peut être émise par tout nœud, quelle que soit sa catégorie.

**Tableau 8-41 – GCCPDU RegistryRetrieveEntryRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Clé	Primitive de demande	Fournisseur GCC sommital

Lorsque le fournisseur GCC sommital reçoit l'unité PDU RegistryRetrieveEntryRequest, il examinera le contenu de l'entrée de référentiel indiquée par la clé. Celle-ci peut posséder l'un des états suivants: vide, contenant un identificateur de canal, un identificateur de jeton ou un paramètre. Dans chacun de ces cas, le fournisseur GCC sommital renvoie au nœud demandeur une unité PDU RegistryResponse (réponse de référentiel) qui contient l'état de l'entité. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryResponse est donné par le Tableau 8-38. Le paramètre "élément de référentiel" contient l'état de l'entrée et, dans le cas d'une entrée non vide, la valeur actuellement mémorisée dans cette entrée. Le paramètre résultat indique un succès si l'entrée n'est

pas vide et "entrée non trouvée" dans le cas contraire. Le paramètre "droits de modification" figurera dans l'unité PDU de réponse si l'entrée est du type "paramètre".

Lorsqu'il reçoit une unité PDU RegistryResponse de ce type, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Retrieve-Entry contenant l'élément de référentiel fourni par l'unité PDU de réponse qu'il enverra au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC sommital ne connaît pas d'entité de protocole d'application inscrite avec un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

#### 8.4.5 Suppression d'une entrée

Lorsqu'il reçoit une primitive de demande GCC-Registry-Delete-Entry (suppression d'entrée par le référentiel), un fournisseur GCC situé dans un nœud conventionnel enverra une unité PDU RegistryDeleteEntryRequest (demande de suppression d'entrée par le référentiel) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryDeleteEntryRequest est donné dans le Tableau 8-42. Une demande GCC-Registry-Delete-Entry émise par un nœud compté ou anonyme sera ignorée par le fournisseur GCC.

**Tableau 8-42 – GCCPDU RegistryDeleteEntryRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Clé	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit une unité PDU RegistryDeleteEntryRequest, le fournisseur GCC sommital tentera de supprimer l'entrée de référentiel désignée. Il vérifie en premier lieu si l'entrée correspondant à la clé donnée existe dans la base de données du référentiel et en détermine dans ce cas le propriétaire. Si l'entrée existe et appartient au demandeur (l'identificateur de nœud et l'identificateur d'entité du demandeur et du propriétaire sont tous deux identiques) ou si l'entrée est actuellement sans propriétaire et que le nœud demandeur soit un nœud conventionnel, le fournisseur GCC sommital efface le contenu de cette entrée en positionnant son état sur vide et non supervisé.

Si la suppression de l'entrée de référentiel a réussi ou si l'entrée n'existait pas, le fournisseur GCC sommital indique que cette entrée a été supprimée en envoyant une unité PDU RegistryResponse (réponse de référentiel) au nœud demandeur. Pour ce faire, il émettra une primitive de demande MCS-Send-Data spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryResponse est donné par le Tableau 8-38. En cas de réussite de l'action, le paramètre résultat indique un succès et le paramètre "élément de référentiel" est vide.

Si l'entrée de référentiel existe déjà mais est la propriété d'un autre nœud ou si le nœud demandeur n'est pas de type conventionnel, le référentiel n'est pas mis à jour et une unité PDU RegistryResponse est renvoyée avec un code résultat négatif indiquant la cause de l'échec. Dans ce cas le paramètre "élément de référentiel" contient la valeur de l'entrée avant la tentative de suppression.

En cas de réussite de la suppression d'une entrée de référentiel existante, le fournisseur GCC sommital détermine si l'entrée de référentiel se trouvait, avant la suppression, dans l'état supervisé. Si ce n'est pas le cas, aucune autre action n'est exécutée. Si l'entrée est supervisée, le fournisseur GCC sommital envoie une unité PDU RegistryMonitorEntryIndication (indication de supervision d'entrée de référentiel) à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de

demande MCS-Uniform-Send-Data spécifiant l'identificateur de canal GCC-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryMonitorEntryIndication est donné par le Tableau 8-44.

En cas de réussite de la suppression d'une entrée de référentiel du type "jeton", le fournisseur GCC sommital peut supprimer l'attribution du jeton correspondant, ce qui signifie qu'il pourra réutiliser ultérieurement, en réponse à une demande RegistryAssignTokenRequest (demande d'attribution d'un jeton par le référentiel), l'identificateur de jeton qui était contenu dans l'entrée supprimée.

Lorsqu'il reçoit une unité PDU RegistryResponse de ce type, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Delete-Entry, indiquant si la demande a réussi ou non conformément au paramètre résultat de l'unité PDU RegistryResponse. Il enverra cette confirmation au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC sommital ne connaît pas d'entité de protocole d'application inscrite avec un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

#### 8.4.6 Supervision d'une entrée

Lorsqu'il reçoit une primitive de demande GCC-Registry-Monitor (supervision de référentiel) contenant un indicateur d'activité positionné sur actif, un fournisseur GCC situé dans un nœud conventionnel enverra une unité PDU RegistryMonitorEntryRequest (demande de supervision de référentiel) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryMonitorEntryRequest est donné par le Tableau 8-43. Le fournisseur GCC mémorise localement un enregistrement indiquant que l'entité de protocole d'application demandeur a activé la supervision de cette entrée. Cette information permet de déterminer s'il y a lieu ou non de produire des indications GCC-Registry-Monitor correspondant à cette entrée lors de la réception de l'unité PDU RegistryMonitorEntryIndication (indication de supervision de référentiel). Une demande GCC-Registry-Monitor émise par un nœud compté ou anonyme sera ignorée par le fournisseur GCC.

Lorsqu'il reçoit une primitive de demande GCC-Registry-Monitor contenant un indicateur d'activité positionné sur inactif, un fournisseur GCC mémorise localement un enregistrement indiquant que l'entité de protocole d'application a désactivé la supervision de cette entrée. Cette information permet de déterminer s'il y a lieu ou non de produire des indications GCC-Registry-Monitor correspondant à cette entrée lors de la réception de l'unité PDU RegistryMonitorEntryIndication.

**Tableau 8-43 – GCCPDU RegistryMonitorEntryRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Clé	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit l'unité PDU RegistryMonitorEntryRequest, le fournisseur GCC sommital tentera de modifier l'état de supervision de l'entrée de référentiel, si le nœud demandeur est de type conventionnel. Il examine en premier lieu si l'entrée de référentiel correspondant à la clé indiquée existe déjà. Dans ce cas, l'état de supervision de l'entrée indiquée est positionné sur actif.

NOTE 1 – Une fois que la supervision est activée, les indications de supervision continueront à être diffusées chaque fois que l'entrée est modifiée, pour la durée de la conférence ou jusqu'à ce que l'entrée soit supprimée.

Le fournisseur GCC sommital envoie ensuite une unité PDU RegistryResponse (réponse de référentiel) au nœud demandeur par l'émission d'une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU



RegistryResponse est donné par le Tableau 8-38. En cas de réussite de l'action, le paramètre résultat indique le succès. Si l'entrée indiquée n'existe pas, le paramètre résultat indique "entrée non trouvée". Les droits de modification figureront dans l'unité PDU de réponse si l'entrée est du type "paramètre".

Si l'entrée de référentiel n'existe pas, le répertoire n'est pas modifié et une unité PDU de réponse RegistryResponse est renvoyée avec un résultat négatif indiquant la cause de l'échec.

Lorsqu'il reçoit une unité PDU RegistryResponse de ce type, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Monitor indiquant, compte tenu du paramètre "réponse" de l'unité PDU RegistryResponse, si la demande a réussi. Il enverra cette demande au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC sommital ne connaît pas d'entité de protocole d'application inscrite qui a un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

Si une entrée de référentiel est réglée de façon à être supervisée, toute modification du contenu de l'entrée, comme le positionnement d'un paramètre, la suppression de l'entrée, le changement de propriétaire ou un changement des droits de modification, conduit le fournisseur GCC sommital à diffuser une unité PDU RegistryMonitorEntryIndication à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data spécifiant à la fois le canal GCC-Broadcast-Channel (pour prendre en charge les nœuds appliquant l'ancien protocole) et le canal GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryMonitorEntryIndication est donné par le Tableau 8-44.

**Tableau 8-44 – GCCPDU RegistryMonitorEntryIndication**

Contenu	Origine	Destination
Clé	Fournisseur GCC sommital	Primitive d'indication
Élément de répertoire	Fournisseur GCC sommital	Primitive d'indication
Propriétaire	Fournisseur GCC sommital	Primitive de confirmation
Droit de modification (optionnel)	Fournisseur GCC sommital	Primitive de confirmation

Lorsqu'il reçoit l'unité PDU RegistryMonitorEntryIndication, un fournisseur GCC d'un nœud prenant en charge la fonction de supervision produira une primitive d'indication GCC-Registry-Monitor qu'il enverra à tout point d'accès GCCSAP ayant demandé la supervision de l'entrée. Cela est le cas si le point d'accès a émis précédemment une primitive de demande GCC-Registry-Monitor avec une clé identique à celle donnée par l'unité PDU d'indication et un indicateur d'activité positionné sur actif et qu'il n'a pas annulé entre-temps cette demande en émettant une unité PDU avec un indicateur d'activité sur inactif. L'unité PDU d'indication est ignorée par le fournisseur GCC s'il n'y a eu aucune demande pour cette clé particulière de ce nœud. La gestion de la base de données locale des points GCCSAP demandeurs, mise à jour chaque fois qu'une primitive de demande GCC-Registry-Monitor est reçue, est une fonction locale du fournisseur GCC.

NOTE 2 – Une implémentation particulière de fournisseur GCC peut choisir de décider localement de ne pas garder la trace des entités de protocole d'application qui ont activé ou désactivé la supervision de chaque entrée et de remplacer cette fonction locale par une émission, vers les points GCCSAP, des indications GCC-Registry-Monitor correspondant à toutes les entités de protocole d'application inscrites lorsqu'une unité PDU RegistryMonitorEntryIndication est reçue.

### 8.4.7 Attribution de poignées non ambiguës

Lorsqu'il reçoit une primitive de demande GCC-Registry-Allocate-Handle (attribution de poignée-pointeur par le référentiel), un fournisseur GCC enverra une unité PDU RegistryAllocateHandleRequest (demande d'attribution de poignée par le référentiel) au fournisseur GCC sommital. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant le canal de l'identificateur de nœud du fournisseur GCC sommital, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryAllocateHandleRequest est donné dans le Tableau 8-45.

**Tableau 8-45 – GCCPDU RegistryAllocateHandleRequest**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC origine	Fournisseur GCC sommital
Nombre de poignées	Primitive de demande	Fournisseur GCC sommital

Lorsqu'il reçoit l'unité PDU RegistryAllocateHandleRequest, le fournisseur GCC sommital produira le nombre requis de poignées non ambiguës. A cet effet, il incrémente modulo  $2^{32}$  une variable d'état de 32 bits après chaque attribution et en utilise la valeur pour attribuer la poignée suivante. La demande est rejetée avec l'indication qu'il n'y a plus de poignées disponibles lorsque toutes les poignées ont été attribuées. Si un nombre suffisant de poignées est disponible, celles-ci sont envoyées au demandeur en indiquant la valeur de la première poignée attribuée ainsi que le nombre de poignées attribuées. Le fournisseur GCC sommital renvoie le résultat au demandeur dans une unité PDU RegistryAllocateHandleResponse (réponse d'attribution de poignée par le référentiel). Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de nœud du demandeur, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU RegistryAllocateHandleResponse est donné par le Tableau 8-46. Le paramètre "élément de référentiel" contient la liste des poignées-pointeurs en cas de résultat positif.

**Tableau 8-46 – GCCPDU RegistryAllocateHandleResponse**

Contenu	Origine	Destination
Identificateur d'entité	Fournisseur GCC sommital	Fournisseur GCC destinataire
Nombre de poignées	Fournisseur GCC sommital	Primitive de confirmation
Première poignée	Fournisseur GCC sommital	Primitive de confirmation
Résultat	Fournisseur GCC sommital	Primitive de confirmation

Lorsqu'il reçoit une indication RegistryAllocateHandleResponse, un fournisseur GCC produira une primitive de confirmation GCC-Registry-Allocate-Handle qu'il enverra au point d'accès GCCSAP de l'entité de protocole d'application indiquée par l'identificateur d'entité. Si le fournisseur GCC sommital ne connaît pas d'entité de protocole d'application inscrite ayant un identificateur d'entité correspondant, l'indication est ignorée et aucune autre action n'est exécutée.

### 8.4.8 Changements de propriétaire et nettoyage du référentiel

Chaque fois que le répertoire d'application est mis à jour, le fournisseur GCC sommital recherchera si une entité de protocole d'application quelconque a été supprimée du répertoire. Dans ce cas, il examinera la totalité de la base de données du référentiel pour déterminer si l'entité de protocole d'application déconnectée était propriétaire d'une entrée de référentiel. De telles entrées sont modifiées pour indiquer qu'elles n'ont plus de propriétaire.

Si le résultat d'une modification du répertoire d'application indique que toutes les entités de protocole d'application ont été résiliées, le fournisseur GCC sommital examinera toutes les entrées de référentiel afin de déterminer si la partie clé de session de la clé de référentiel coïncide avec la clé de session de la session précédente. Le fournisseur supprimera les entrées du référentiel correspondant à la session supprimée. Si cette entrée de référentiel correspond à un type jeton, le fournisseur GCC sommital peut supprimer l'allocation du jeton afin de permettre sa réutilisation ultérieure.

## **8.5 Présidence de conférence**

La présidence de la conférence est gérée au moyen du jeton de présidence de conférence. La conférence se trouve en mode non présidé lorsque le jeton est libre. Elle se trouve en mode présidé lorsque le jeton est pris, le détenteur du jeton étant le président de conférence. Le jeton ne peut être pris que par des nœuds conventionnels. Les nœuds anonymes et comptés ne peuvent jamais assurer la présidence d'une conférence.

Lors de l'acquisition du jeton de présidence de conférence, le président de la conférence, s'il n'est pas le fournisseur GCC sommital, prouvera au fournisseur GCC sommital qu'il est en possession du jeton de présidence en offrant de le céder par l'invocation d'une primitive de demande MCS-Token-Give (cession de jeton) et en spécifiant le fournisseur GCC sommital comme récepteur. Le fournisseur GCC sommital répondra négativement à cette primitive, ce qui a pour effet de laisser le jeton en possession du président de la conférence. Si par la suite le jeton est à nouveau offert par le président de conférence en exercice, le fournisseur interprétera cela comme faisant partie d'une opération GCC-Conductor-Give et ne refusera pas automatiquement le jeton.

### **8.5.1 Prise de présidence**

Lorsqu'il reçoit une primitive de demande GCC-Conductor-Assign (assignation de président), un fournisseur GCC situé dans un nœud conventionnel examinera en premier lieu le profil de conférence pour déterminer si la conférence peut être présidée ou non. Si ce n'est pas le cas, il produira une primitive de confirmation GCC-Conductor-Assign avec un résultat négatif pour indiquer que la conférence ne peut pas être présidée. Si la conférence peut être présidée, le fournisseur GCC examinera ensuite sa base de données locale afin de déterminer s'il est le président de la conférence. Il produira dans ce cas une primitive de confirmation GCC-Conductor-Assign contenant un résultat positif qu'il enverra au point de contrôle GCCSAP. Dans le cas contraire, il tentera de s'emparer du jeton en envoyant la primitive de demande MCS-Token-Grab (saisie de jeton) indiquant le jeton Conference-Conductorship-Token (jeton de présidence de conférence) comme identificateur du jeton à saisir. Une demande GCC-Conductor-Assign émise par un nœud compté ou anonyme sera ignorée par le fournisseur GCC.

Lorsqu'il reçoit la primitive de confirmation MCS-Token-Grab, le fournisseur GCC examinera son paramètre résultat. Si le résultat est positif, le fournisseur GCC enverra la primitive de demande MCS-Token-Give, spécifiant l'identificateur de présidence de conférence comme jeton proposé et l'identificateur de nœud du fournisseur GCC sommital comme identificateur d'utilisateur devant recevoir le jeton. Lorsqu'il reçoit la primitive de confirmation MCS-Token-Give, le fournisseur GCC examinera le paramètre résultat. Si le résultat est négatif, le fournisseur GCC produira localement la primitive de confirmation GCC-Conductor-Assign indiquant un résultat positif qu'il enverra au point de contrôle GCCSAP.

Lorsqu'il reçoit une primitive d'indication spécifiant le jeton de présidence de conférence comme jeton proposé par un identificateur d'utilisateur que le fournisseur GCC sommital ne reconnaît pas comme conducteur de conférence en exercice, le fournisseur GCC sommital mettra à jour ses enregistrements afin de reconnaître le donneur comme président de la conférence. Il enverra également à tous les nœuds une unité PDU ConductorAssignIndication (indication d'assignation de président). Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant l'identificateur de canal GCC-Broadcast-Channel (canal de

diffusion), une priorité de données maximale et l'unité PDU dans le champ de données. Le contenu de l'unité PDU `ConductorAssignIndication` est donné par le Tableau 8-47.

Si la primitive de confirmation `MCS-Token-Grab` indique un résultat négatif ou si le paramètre de résultat de la primitive de confirmation `MCS-Token-Give` indique un résultat positif, le fournisseur GCC produira une primitive de confirmation `GCC-Conductor-Assign` indiquant un résultat négatif qu'il enverra au point de contrôle GCCSAP.

**Tableau 8-47 – GCCPDU ConductorAssignIndication**

Contenu	Origine	Destination
Nœud président	Fournisseur GCC sommital	Fournisseur GCC destinataire

### 8.5.2 Abandon de la présidence

Lorsqu'il reçoit une primitive de demande `GCC-Conductor-Release` (abandon de la présidence), un fournisseur GCC examinera en premier lieu sa base de données locale afin de déterminer s'il est le président de la conférence. Si ce n'est pas le cas, il produira une primitive de confirmation `GCC-Conductor-Release` indiquant un résultat négatif qu'il enverra au point de contrôle GCCSAP. S'il est le président en exercice, il enverra d'abord une unité PDU `ConductorReleaseIndication` (indication d'abandon de la présidence) à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande `MCS-Uniform-Send-Data` (envoi uniforme de données) spécifiant les deux identificateurs de canal: `GCC-Broadcast-Channel` (canal de diffusion prenant en charge les nœuds appliquant l'ancien protocole) et `GCC-Conventional-Broadcast-Channel`, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU `ConductorReleaseIndication` est donné par le Tableau 8-48. Cette PDU n'a pas de paramètres. Il abandonnera ensuite le jeton de présidence par l'émission d'une primitive de demande `MCS-Token-Release` (libération de jeton) indiquant comme identificateur de jeton à abandonner le jeton `Conference-Conductorship-Token` (jeton de direction de conférence).

Lorsqu'il reçoit la primitive de confirmation `MCS-Token-Release`, le fournisseur GCC produira localement la primitive de confirmation `GCC-Conductor-Release` indiquant un résultat positif qu'il enverra au point de contrôle GCCSAP.

**Tableau 8-48 – GCCPDU ConductorReleaseIndication**

Contenu	Origine	Destination
-- Pas de paramètres --		

### 8.5.3 Indication de prise et d'abandon de présidence

Un nœud qui prend en charge des protocoles d'application qui ont un comportement différent selon que le mode est présidé ou non répondra à la réception des unités PDU `ConductorAssignIndication` (indication d'assignation de président) et `ConductorReleaseIndication` (indication d'abandon de présidence). Un nœud qui ne prend pas en charge de tels protocoles d'application peut choisir de les ignorer.

Lorsqu'il reçoit une unité PDU `ConductorAssignIndication` contenue dans une primitive d'indication `MCS-Uniform-Send-Data`, un fournisseur GCC examinera le profil de conférence afin de déterminer si la conférence peut être présidée. Il ignorera l'unité PDU si ce n'est pas le cas. Si la conférence peut être présidée, il peut produire une primitive d'indication `GCC-Conductor-Assign` et l'envoyer au point de contrôle GCCSAP ainsi qu'au point d'accès GCCSAP de toutes les entités de protocole d'application inscrites.

Pour les nœuds conventionnels, le fournisseur GCC déterminera en premier lieu si le champ "identificateur d'utilisateur de l'émetteur" de la primitive d'indication MCS-Uniform-Send-Data signale que l'unité PDU ConductorAssignIndication a été transmise par le fournisseur GCC sommital. Si ce n'est pas le cas, l'unité PDU sera ignorée et aucune primitive d'indication ne sera produite. Si l'unité PDU a été transmise par le fournisseur GCC sommital, le paramètre "identificateur de nœud" de la primitive reçoit la valeur indiquée par le paramètre "nœud chargé de la présidence" de l'unité PDU reçue. Le fournisseur GCC d'un nœud qui prend en charge les primitives de présidence mémoriserà dans sa base de données locale l'indication que la conférence se trouve maintenant en mode présidé ainsi que l'identificateur de nœud du président.

Pour les nœuds comptés et anonymes, une unité ConductorAssignIndication peut être reçue soit du nœud parent soit du fournisseur GCC sommital. Si l'unité PDU provient d'un nœud autre que l'un de ceux-là, elle doit être ignorée et aucune indication de la primitive ne doit être produite. Si l'unité PDU a été émise soit par le nœud parent soit par le fournisseur GCC sommital, le paramètre "identificateur de nœud" de la primitive reçoit la valeur indiquée par le paramètre "nœud chargé de la présidence" dans l'unité PDU reçue. Une unité ConductorAssignIndication reçue du fournisseur GCC sommital a toujours priorité sur celle qui est reçue d'un nœud parent. Lorsqu'une unité ConductorAssignIndication a été reçue du fournisseur GCC sommital, toute indication de ce type reçue d'un nœud parent doit être ignorée. Le fournisseur GCC d'un nœud qui prend en charge les primitives de présidence et qui reçoit une unité ConductorAssignIndication valide mémoriserà dans sa base de données locale l'indication que la conférence se trouve maintenant en mode présidé ainsi que l'identificateur de nœud du président.

Lorsqu'il reçoit une unité PDU ConductorReleaseIndication contenue dans une primitive d'indication MCS-Uniform-Send-Data, un fournisseur GCC peut produire une primitive d'indication GCC-Conductor-Release (abandon de présidence) et l'envoyer au point de contrôle GCCSAP ainsi qu'au point d'accès GCCSAP de toutes les entités de protocole d'application inscrites.

Pour les nœuds conventionnels, le fournisseur GCC déterminera d'abord si le champ "identificateur d'utilisateur émetteur" de la primitive d'indication MCS-Uniform-Send-Data indique que l'unité PDU ConductorReleaseIndication a été transmise par le fournisseur GCC sommital ou par le nœud qu'il considère actuellement comme président. Si tel n'est pas le cas, l'unité PDU sera ignorée et aucune primitive d'indication ne sera produite. Le fournisseur GCC d'un nœud qui prend en charge les primitives de présidence mémoriserà dans sa base de données locale l'information indiquant que la conférence n'est plus en mode présidé. Le fournisseur GCC positionnera son indicateur local d'autorisation pour signaler qu'aucune autorisation de mode présidé n'est attribuée.

Pour les nœuds comptés et anonymes, une unité ConductorReleaseIndication peut être reçue soit du nœud parent soit du fournisseur GCC sommital ou du président actuel. Si l'unité PDU provient d'un nœud autre que l'un de ceux-là, elle doit être ignorée et aucune indication de la primitive ne doit être produite. Si un nœud compté ou anonyme reçoit une unité ConductorReleaseIndication sur le canal GCC-Conventional-Broadcast-Channel et que ce nœud doit encore déterminer l'état actuel de la présidence dans la conférence (parce qu'il n'a pas reçu d'indication de son nœud parent), ce nœud doit partir du principe que l'unité ConductorReleaseIndication provient d'un nœud valide: il doit alors mémoriser dans sa base de données le fait que la conférence n'est pas en mode présidé. Toute unité ConductorReleaseIndication reçue sur le canal GCC-Conventional-Broadcast-Channel a priorité sur une quelconque unité ConductorReleaseIndication reçue du nœud parent. Le fournisseur GCC d'un nœud qui prend en charge les primitives de présidence et qui reçoit une unité ConductorReleaseIndication valide mémoriserà dans sa base de données locale l'indication que la conférence n'est plus en mode présidé.

Un fournisseur GCC respectera l'ordre de réception des unités PDU ConductorAssignIndication et ConductorReleaseIndication reçues soit par le canal GCC-Broadcast-Channel soit par le canal GCC-Conventional-Broadcast-Channel, quand il produira les primitives d'indication correspondantes.

Le fournisseur GCC d'un nœud qui prend en charge les primitives de présidence enverra une primitive d'indication GCC-Conductor-Assign ou GCC-Conductor-Release au point d'accès GCCSAP d'une entité de protocole d'application qui s'inscrit pour la première fois après le début d'une conférence. L'état de la présidence de la conférence tel qu'il est connu par le fournisseur GCC détermine laquelle des deux primitives est émise.

#### **8.5.4 Demande d'attribution de présidence**

Lorsqu'il reçoit une primitive de demande GCC-Conductor-Please (demande de présidence), un fournisseur GCC situé dans un nœud conventionnel déterminera en premier lieu à partir de sa base de données locale si la conférence se trouve en mode présidé. Si ce n'est pas le cas, il produira une primitive de confirmation GCC-Conductor-Please avec un résultat négatif indiquant que la conférence n'est pas en mode présidé et l'enverra au point de contrôle GCCSAP. Si la conférence est en mode présidé, le fournisseur GCC enverra une primitive de demande MCS-Token-Please (demande de jeton) indiquant comme jeton demandé le jeton Conference-Conductorship-Token (jeton de présidence de conférence). Il produira également une primitive de confirmation GCC-Conductor-Please indiquant un résultat positif et l'enverra au point de contrôle GCCSAP.

Lorsqu'il reçoit une primitive d'indication MCS-Token-Please contenant l'identificateur du jeton Conference-Conductorship-Token, un fournisseur GCC produira et enverra au point de contrôle GCCSAP une primitive d'indication GCC-Conductor-Please contenant l'identificateur du nœud demandeur déduit du paramètre "identificateur d'utilisateur" reçu dans la primitive d'indication MCS-Token-Please. Une demande GCC-Conductor-Please émise par un nœud compté ou anonyme sera ignorée par le fournisseur GCC.

#### **8.5.5 Transfert de présidence**

Lorsqu'il reçoit une primitive de demande GCC-Conductor-Give (cession de présidence), un fournisseur GCC déterminera en premier lieu à partir de la base de données locale s'il est le président en exercice. Si ce n'est pas le cas, il produira une primitive de confirmation GCC-Conductor-Give avec un résultat négatif indiquant qu'il n'est pas le président de la conférence et enverra cette confirmation au point de contrôle GCCSAP. Sinon, le fournisseur GCC tentera de donner la présidence de la conférence au nœud en question en envoyant une primitive de demande MCS-Token-Give (cession de jeton) indiquant comme identificateur de jeton le jeton Conference-Conductorship-Token (jeton de présidence de conférence) ainsi que le nœud récepteur fourni dans la primitive de demande GCC-Conductor-Give reçue. Le nœud qui reçoit la présidence doit toujours être un nœud conventionnel.

Lorsqu'il reçoit la primitive de confirmation MCS-Token-Give, le fournisseur GCC examinera son paramètre résultat. Si le résultat est positif, le fournisseur GCC produira localement la primitive de confirmation GCC-Conductor-Give indiquant un résultat positif qu'il enverra au point de contrôle GCCSAP. Si le paramètre résultat de la confirmation MCS-Token-Give indique un résultat négatif, le fournisseur GCC produira ensuite et enverra au point de contrôle GCCSAP une primitive de confirmation GCC-Conductor-Give indiquant un résultat négatif qui reproduit le résultat fourni par la confirmation MCS-Token-Give.

Lorsqu'il reçoit une primitive d'indication MCS-Token-Give indiquant le jeton Conference-Conductorship-Token, un fournisseur GCC qui n'est pas le fournisseur GCC sommital produira une primitive d'indication GCC-Conductor-Give qu'il enverra au point de contrôle GCCSAP. Lorsqu'il reçoit la primitive de réponse GCC-Conductor-Give, le fournisseur GCC examinera son paramètre résultat. Si le résultat est positif, le fournisseur GCC enverra la primitive de réponse MCS-Token-Give indiquant que le jeton a été accepté.

Si la primitive de réponse GCC-Conductor-Give indique que le jeton n'a pas été accepté, le fournisseur GCC enverra la primitive de réponse MCS-Token-Give indiquant que le jeton a été rejeté par l'utilisateur.

Si un nœud qui ne prend pas en charge les primitives de présidence reçoit une primitive d'indication MCS-Token-Give, le fournisseur GCC renverra une primitive de réponse MCS-Token-Give indiquant le résultat "rejet par l'utilisateur". Cela concerne les nœuds anonymes et les nœuds comptés.

Lorsqu'il reçoit une primitive d'indication MCS-Token-Give spécifiant le jeton de présidence de conférence comme jeton proposé par un identificateur d'utilisateur que le fournisseur GCC sommital ne reconnaît pas comme conducteur de conférence en exercice, le fournisseur GCC sommital répondra comme spécifié au § 8.5.1 pour rejeter le jeton d'une manière automatique. Toutefois, s'il reconnaît déjà le donneur comme président, le fournisseur GCC sommital produira, comme tout autre fournisseur GCC, une primitive d'indication GCC-Conductor-Give.

### **8.5.6 Interrogation de l'état de la présidence**

Lorsqu'il reçoit une primitive de demande GCC-Conductor-Inquire (interrogation de président), un fournisseur GCC déterminera en premier lieu à partir de sa base de données locale si la conférence se trouve actuellement en mode présidé. Si ce n'est pas le cas, le fournisseur GCC produira et enverra au point de contrôle GCCSAP une primitive de confirmation GCC-Conductor-Inquire indiquant que la conférence n'est pas en mode présidé. Sinon, il produira une réponse indiquant que la conférence se trouve en mode présidé et donnera comme président en exercice l'identificateur du nœud président mémorisé dans sa base de données locale. Si, dans cette situation, un nœud vient juste d'entrer dans une conférence et n'a pas encore reçu du président une unité PDU ConductorAssignIndication (indication d'assignation de président), il ne connaît pas encore l'identificateur du nœud président dans sa base de données locale. Dans ce cas, il attendra l'arrivée de l'unité PDU ConductorAssignIndication avant de produire la primitive de confirmation GCC-Conductor-Inquire.

Si la conférence se trouve dans le mode présidé, le fournisseur GCC déterminera le positionnement de l'indicateur d'autorisation à transmettre dans la primitive de confirmation GCC-Conductor-Inquire en recherchant si le nœud local est le président en exercice, auquel cas la permission est considérée comme accordée. Sinon, il déterminera si l'identificateur du nœud local a été placé dans la liste d'autorisations de l'unité PDU ConductorPermissionGrantIndication (indication d'autorisation d'attribution de présidence) la plus récente reçue depuis la dernière transition de l'état non présidé vers l'état présidé. Si le nœud local apparaît dans cette liste, l'indicateur d'autorisation est positionné sur TRUE (vrai). Si ce n'est pas le cas ou si aucune unité PDU ConductorPermissionGrantIndication n'a été reçue depuis la plus récente transition de l'état non présidé vers l'état présidé, cet indicateur est positionné sur FALSE (faux).

### **8.5.7 Annonce de présidence lors de l'entrée en conférence de nouveaux nœuds**

S'il reçoit une mise à jour du répertoire de conférence avec un indicateur signalant que de nouveaux nœuds conventionnels se trouvent dans la conférence alors que celle-ci est dans le mode présidé, le fournisseur GCC sommital enverra à tous les nœuds de la conférence une unité PDU ConductorAssignIndication (indication d'assignation de président). Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant les deux identificateurs de canal GCC-Broadcast-Channel (canal de diffusion pour les nœuds appliquant l'ancien protocole) et GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConductorAssignIndication est donné par le Tableau 8-47. Le paramètre "nœud président" indique l'identificateur de nœud du président en exercice.

S'il reçoit une mise à jour du répertoire de conférence avec un indicateur signalant que de nouveaux nœuds conventionnels se trouvent dans la conférence alors que la conférence n'est pas en mode présidé, le fournisseur GCC sommital enverra à tous les nœuds de la conférence une unité PDU ConductorReleaseIndication (indication d'abandon de présidence). Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant les deux

identificateurs de canal GCC-Broadcast-Channel (canal de diffusion pour les nœuds appliquant l'ancien protocole) et GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConductorReleaseIndication est donné par le Tableau 8-48.

Si le fournisseur GCC sommital reçoit d'un autre nœud une unité ConductorReleaseIndication signalant une modification de l'état de la présidence avant qu'il n'ait reçu ses propres PDU ConductorAssignIndication transmises, le fournisseur GCC sommital émettra de nouveau une unité PDU ConductorReleaseIndication pour indiquer le nouvel état.

Si le fournisseur GCC sommital reçoit une unité PDU RosterUpdateIndication indiquant que de nouveaux nœuds comptés ou anonymes sont présents dans la conférence, ce fournisseur ne diffuse pas d'annonce initiale de présidence et n'envoie donc pas d'unité ConductorAssignIndication ou ConductorReleaseIndication. L'annonce initiale de présidence est au contraire confiée au nœud parent du nœud entrant. Cette annonce initiale est produite comme suit: lorsqu'un nœud parent reçoit l'identificateur nodal d'un nouveau nœud anonyme ou compté entrant, il doit envoyer à ce nouveau nœud une unité ConductorAssignIndication ou ConductorReleaseIndication pour l'informer de l'état actuel de la présidence dans la conférence. Si celle-ci se trouve en mode présidé, le nœud parent doit envoyer une unité PDU ConductorAssignIndication directement au nouveau nœud compté ou anonyme, contenue dans une primitive de demande MCS-Send-Data spécifiant l'identificateur nodal du nouveau nœud comme identificateur de canal, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données. Si la conférence ne se trouve pas en mode présidé, le nœud parent doit envoyer une unité PDU ConductorReleaseIndication directement au nouveau nœud compté ou anonyme, contenue dans une primitive de demande MCS-Send-Data spécifiant l'identificateur nodal du nouveau nœud comme identificateur de canal, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données.

Un nœud qui vient d'entrer dans une conférence suspend le traitement de toute primitive en relation avec la présidence jusqu'à la réception d'une unité PDU ConductorAssignIndication ou ConductorReleaseIndication. Cela est nécessaire parce que l'état de la présidence géré par ce nœud dans sa base de données locale n'est pas connu tant que l'une de ces deux primitives n'a pas été reçue.

### **8.5.8 Déconnexion imprévue du président**

Si le président d'une conférence se déconnecte d'une manière imprévue, il peut ne pas avoir la possibilité d'envoyer à tous les nœuds une unité PDU ConductorReleaseIndication (indication d'abandon de conférence) indiquant que la conférence se trouve maintenant dans le mode non présidé. Le fournisseur GCC de tout nœud qui prend en charge les primitives de présidence surveillera les indications MCS-Detach-User (détachement d'utilisateur) et comparera l'identificateur d'utilisateur qu'elles contiennent avec l'identificateur du nœud qu'il estime être le président en exercice. En cas d'identité, le fournisseur GCC mémorisera dans sa base de données locale le fait que la conférence n'est plus présidée. Le fournisseur GCC produira une primitive d'indication GCC-Conductor-Release (abandon de présidence) qu'il enverra au point de contrôle GCCSAP ainsi qu'au point d'accès GCCSAP de toutes les entités de protocole d'application inscrites. Le fournisseur GCC positionnera son indicateur local d'autorisation pour signaler qu'aucune autorisation de mode présidé n'est attribuée.

### **8.5.9 Demande d'attribution de l'autorisation de mode présidé**

Lorsqu'il reçoit via le point de contrôle GCCSAP une primitive de demande GCC-Conductor-Permission-Ask (demande d'autorisation du président), un fournisseur GCC qui prend en charge cette primitive déterminera en premier lieu si la conférence se trouve dans le mode présidé. Si ce n'est pas le cas, il produira et enverra au point de contrôle GCCSAP une primitive de confirmation GCC-Conductor-Permission-Ask avec un code résultat indiquant le mode non présidé. Si la conférence est en mode présidé, le fournisseur GCC enverra au président mentionné une unité



PDU ConductorPermissionAskIndication (indication de demande d'autorisation du président). Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant l'identificateur de canal GCC-Broadcast-Channel (canal de diffusion), spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConductorPermissionAskIndication est donné par le Tableau 8-49. En cas d'exécution correcte, le fournisseur GCC produira une primitive de confirmation GCC-Conductor-Permission-Ask indiquant un résultat positif qu'il enverra au point de contrôle GCCSAP.

**Tableau 8-49 – GCCPDU ConductorPermissionAskIndication**

Contenu	Origine	Destination
Indicateur d'attribution/abandon	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU ConductorPermissionAskIndication, le fournisseur GCC déterminera s'il est actuellement le président de la conférence. Si ce n'est pas le cas, il ignorera l'unité PDU et n'exécutera aucune autre action. S'il est le président et qu'il prend en charge la primitive correspondante, le fournisseur GCC produira une primitive GCC-Conductor-Permission-Ask qu'il enverra au point de contrôle GCCSAP. L'identificateur de nœud du paramètre "demandeur" sera rempli avec la valeur du paramètre "identificateur d'utilisateur de l'émetteur" de la primitive d'indication MCS-Uniform-Send-Data reçue. Un fournisseur GCC respectera l'ordre de réception des unités PDU ConductorPermissionAskIndication lorsqu'il produira les primitives d'indication correspondantes.

#### 8.5.10 Octroi d'autorisation de mode présidé

Lorsqu'il reçoit via le point de contrôle GCCSAP une primitive de demande GCC-Conductor-Permission-Grant (octroi d'autorisation du président), un fournisseur GCC qui prend en charge cette primitive examinera en premier lieu si la conférence se trouve dans le mode présidé. Si ce n'est pas le cas, il générera et enverra au point de contrôle GCCSAP une primitive de confirmation GCC-Conductor-Permission-Grant indiquant un résultat "pas en mode présidé" comme motif de rejet. Il n'exécutera ensuite aucune autre action. Si la conférence se trouve en mode présidé, le fournisseur GCC diffusera une PDU ConductorPermissionGrantIndication (indication d'octroi d'autorisation du président) à tous les nœuds de la conférence. Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) avec en paramètre l'identificateur de canal GCC-Broadcast-Channel (canal de diffusion), une priorité de données maximale et la PDU dans le champ de données. Le contenu de la PDU ConductorPermissionGrantIndication est donné par le Tableau 8-50. Les paramètres "liste d'autorisations" et "liste d'attente" reçoivent les valeurs de la liste d'identificateurs de nœud fournie par les paramètres correspondants de la primitive de demande. L'ordre des deux listes est préservé. Le fournisseur GCC générera une primitive de confirmation GCC-Conductor-Permission-Grant après la réussite de l'opération et l'enverra au point de contrôle GCCSAP pour indiquer un résultat positif.

**Tableau 8-50 – GCCPDU ConductorPermissionGrantIndication**

Contenu	Origine	Destination
Liste d'autorisations	Primitive de demande	Primitive d'indication
Liste d'attente (optionnel)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une PDU ConductorPermissionGrantIndication, un fournisseur GCC qui prend en charge la primitive correspondante examinera en premier lieu si la conférence se trouve actuellement dans le mode présidé. Si ce n'est pas le cas, il ignorera la PDU reçue et n'exécutera aucune autre action. Si la conférence est en mode présidé, il déterminera si la PDU reçue provient

du nœud qui est actuellement le président de conférence. Si ce n'est pas le cas, il ignorera également la PDU reçue et n'exécutera aucune autre action. Si le nœud origine est le président en exercice, le fournisseur GCC générera une primitive d'indication GCC-Conductor-Permission-Grant qu'il enverra au point de contrôle GCCSAP ainsi qu'aux points d'accès GCCSAP de toutes les entités de protocole d'application inscrites. Les listes d'autorisations et d'attente de la primitive d'indication sont remplies à partir des paramètres correspondants de la PDU en préservant l'ordre de ces listes. Le fournisseur GCC examinera le paramètre "liste d'autorisations" et déterminera si le nœud local appartient à cette liste. Dans ce cas, il positionnera le paramètre "indicateur d'autorisation" de la primitive d'indication sur TRUE (vrai), sinon il positionne cet indicateur sur FALSE (faux). Un fournisseur GCC respectera l'ordre de réception des PDU ConductorPermissionGrantIndication lorsqu'il génère les primitives d'indication correspondantes.

## 8.6 Fonctions diverses

### 8.6.1 Conférences limitées dans le temps

Lorsqu'il reçoit une primitive de demande GCC-Conference-Time-Remaining (temps restant dans une conférence), un fournisseur GCC enverra à tous les nœuds de la conférence indiquée une unité PDU ConferenceTimeRemainingIndication (indication de temps restant dans une conférence). Pour ce faire, il émettra une primitive de demande MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant les deux identificateurs de canal GCC-Broadcast-Channel (canal de diffusion pour les nœuds appliquant l'ancien protocole) et GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConferenceTimeRemainingIndication est donné par le Tableau 8-51. La valeur du paramètre "temps restant" est obtenue à partir du contenu de la primitive de demande. Le fournisseur GCC produira également une primitive de confirmation GCC-Conference-Time-Remaining indiquant un résultat positif qu'il enverra au point de contrôle GCCSAP.

**Tableau 8-51 – GCCPDU ConferenceTimeRemainingIndication**

Contenu	Origine	Destination
Temps restant	Primitive de demande	Primitive d'indication
Identificateur de nœud (optionnel)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU ConferenceTimeRemainingIndication, un fournisseur GCC peut, d'une manière facultative, générer une primitive d'indication GCC-Conference-Time-Remaining et l'envoyer au point de contrôle GCCSAP. Le paramètre "temps restant" est obtenu à partir de l'unité PDU. Le paramètre "nœud source" est rempli par la valeur de l'identificateur d'utilisateur émetteur de la primitive d'indication MCS-Uniform-Send-Data reçue.

Lorsqu'il reçoit une primitive de demande GCC-Conference-Time-Inquire (interrogation de temps d'une conférence), un fournisseur GCC enverra une unité PDU ConferenceTimeInquireIndication (indication d'interrogation de temps d'une conférence) à l'organisateur de conférence. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de canal GCC-Convenor-Channel (canal d'organisateur), spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConferenceTimeInquireIndication est donné par le Tableau 8-52. Le fournisseur GCC produira ensuite une primitive de confirmation GCC-Conference-Time-Inquire qu'il enverra au point de contrôle GCCSAP.

**Tableau 8-52 – GCCPDU ConferenceTimeInquireIndication**

Contenu	Origine	Destination
Indicateur de temps spécifique du nœud	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU ConferenceTimeInquireIndication, le fournisseur GCC de l'organisateur de conférence peut, d'une manière facultative, générer une primitive d'indication GCC-Conference-Time-Inquire et l'envoyer au point de contrôle GCCSAP. L'identificateur de nœud du demandeur, figurant dans le paramètre "nœud origine" de la primitive d'indication MCS-Send-Data, est utilisé comme paramètre "identificateur d'utilisateur du nœud demandeur" de la primitive d'indication GCC-Conference-Time-Inquire.

Lorsqu'il reçoit une primitive de demande GCC-Conference-Extend, un fournisseur GCC enverra une unité PDU ConferenceTimeExtendIndication à l'organisateur de conférence. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur de canal GCC-Convener-Channel (canal d'organisateur), spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConferenceTimeExtendIndication est donné par le Tableau 8-53. Le fournisseur GCC produira ensuite une primitive de confirmation GCC-Conference-Extend qu'il enverra au point de contrôle GCCSAP.

**Tableau 8-53 – GCCPDU ConferenceTimeExtendIndication**

Contenu	Origine	Destination
Extension de temps	Primitive de demande	Primitive d'indication
Indicateur de temps spécifique au nœud	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU ConferenceTimeExtendIndication, le fournisseur GCC de l'organisateur de conférence peut, d'une manière facultative, générer une primitive d'indication GCC-Conference-Extend et l'envoyer au point de contrôle GCCSAP. L'identificateur de nœud du demandeur, indiqué dans le paramètre "nœud origine" de la primitive d'indication MCS-Send-Data, est utilisé comme identificateur de nœud du paramètre "nœud demandeur" de la primitive d'indication GCC-Conference-Extend.

### 8.6.2 Demande d'assistance de conférence

Lorsqu'il reçoit une primitive de demande GCC-Conference-Assistance (assistance de conférence), un fournisseur GCC enverra une unité PDU ConferenceAssistanceIndication (indication d'assistance de conférence) à tous les nœuds de la conférence indiquée. Pour ce faire, il émettra une primitive de demande MCS-Send-Data ou MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant les deux identificateurs de canal GCC-Broadcast-Channel (canal de diffusion pour les nœuds appliquant l'ancien protocole) et GCC-Conventional-Broadcast-Channel, spécifiant une priorité de données maximale et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU ConferenceAssistanceIndication est donné par le Tableau 8-54. Le fournisseur GCC produira ensuite une primitive de confirmation GCC-Conference-Assistance indiquant si l'opération a réussi ou non.

**Tableau 8-54 – GCCPDU ConferenceAssistanceIndication**

Contenu	Origine	Destination
Données utilisateur (optionnel)	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU ConferenceAssistanceIndication, un nœud qui prend en charge cette fonction peut générer une primitive d'indication GCC-Conference-Assistance qu'il enverra au point de contrôle GCCSAP. Le paramètre "nœud origine" est déduit de l'identificateur d'utilisateur de l'émetteur de la primitive d'indication MCS-Uniform-Send-Data reçue.

### 8.6.3 Diffusion d'un message de texte

Lorsqu'il reçoit une primitive de demande GCC-Text-Message (message de texte) qui ne contient pas de nœud destinataire, un fournisseur GCC enverra à l'ensemble des nœuds de la conférence indiquée une unité PDU TextMessageIndication (indication de message de texte). Pour ce faire, il émettra une primitive de demande MCS-Send-Data ou MCS-Uniform-Send-Data (envoi uniforme de données) spécifiant les deux identificateurs de canal GCC-Broadcast-Channel (canal de diffusion pour les nœuds appliquant l'ancien protocole) et GCC-Conventional-Broadcast-Channel, en spécifiant une priorité de données maximale et en incluant l'unité PDU dans le champ de données. Si la primitive de demande GCC-Text-Message contient un nœud destinataire, le fournisseur GCC enverra la même PDU uniquement à ce nœud. Pour ce faire, il émettra une primitive de demande MCS-Send-Data (envoi de données) spécifiant l'identificateur du nœud demandé, spécifiant une priorité de données élevée et incluant l'unité PDU dans le champ de données. Le contenu de l'unité PDU TextMessageIndication est donné par le Tableau 8-55. Le fournisseur GCC produira ensuite une primitive de confirmation GCC-Text-Message indiquant si l'opération a réussi ou non.

**Tableau 8-55 – GCCPDU TextMessageIndication**

Contenu	Origine	Destination
Message	Primitive de demande	Primitive d'indication

Lorsqu'il reçoit une unité PDU TextMessageIndication, un nœud qui prend en charge cette fonction peut générer une primitive d'indication GCC-Text-Message et l'envoyer au point de contrôle GCCSAP. Le paramètre "nœud origine" est déduit de l'identificateur d'utilisateur de l'émetteur contenu dans la primitive d'indication MCS-Uniform-Send-Data reçue.

## 8.7 Définitions des GCCPDU

La structure des GCCPDU est spécifiée ci-dessous en utilisant la notation ASN.1 de [UIT-T X.680]. Toutes les unités GCCPDU sont codées à des fins de transmission en utilisant la variante de base cadrée des règles de codage condensées de la Recommandation X.691.

NOTE – L'utilisation d'indicateurs automatiques dans les définitions de protocole GCC implique que l'ordre des structures SEQUENCE et CHOICE contenues dans cette définition affecte les valeurs effectivement codées.

```
GCC-PROTOCOL DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
```

```
-- Export all symbols
```

```
-- =====
-- Part 1: Elements of which messages are composed
-- =====
```

```

ChannelID ::= INTEGER (1..65535)
StaticChannelID ::= INTEGER (1..1000)
-- Those assigned by specifications
DynamicChannelID ::= INTEGER (1001..65535)
-- Those created and deleted by MCS
UserID ::= DynamicChannelID
TokenID ::= INTEGER (1..65535)
StaticTokenID ::= INTEGER (1..16383)
-- Those assigned by specifications
DynamicTokenID ::= INTEGER (16384..65535)
-- Those assigned by the registry
Time ::= INTEGER (-2147483648..2147483647)
-- Time in seconds
Handle ::= INTEGER (0..4294967295)
-- 32-bit value

H221NonStandardIdentifier ::= OCTET STRING (SIZE (4..255))
-- First four octets shall be country code and
-- Manufacturer code, assigned as specified in
-- Annex A/H.221 for NS-cap and NS-comm

Key ::= CHOICE -- Identifier of a standard or non-standard object
{
    object OBJECT IDENTIFIER,
    h221NonStandard H221NonStandardIdentifier
}

NonStandardParameter ::= SEQUENCE
{
    key Key,
    data OCTET STRING
}

TextString ::= BMPString (SIZE (0..255))
-- Basic Multilingual Plane of ISO/IEC 10646-1 (Unicode)

simpleTextFirstCharacter UniversalString ::= {0, 0, 0, 0}
simpleTextLastCharacter UniversalString ::= {0, 0, 0, 255}

SimpleTextString ::= BMPString (SIZE (0..255)) (FROM
(simpleTextFirstCharacter..simpleTextLastCharacter))

SimpleNumericString ::= NumericString (SIZE (1..255)) (FROM ("0123456789"))

DiallingString ::= NumericString (SIZE (1..16)) (FROM ("0123456789"))

SubAddressString ::= NumericString (SIZE (1..40)) (FROM ("0123456789"))

ExtraDiallingString ::= TextString (SIZE (1..255)) (FROM ("0123456789#*,"))

UserData ::= SET OF SEQUENCE
{
    key Key,
    value OCTET STRING OPTIONAL
}

Password ::= SEQUENCE
{
    numeric SimpleNumericString,
    text SimpleTextString OPTIONAL,
    ...,
    unicodeText TextString OPTIONAL
}

```

```

PasswordSelector ::= CHOICE
{
    numeric      SimpleNumericString,
    text         SimpleTextString,
    ...,
    unicodeText  TextString
}

ChallengeResponseItem ::= CHOICE
{
    passwordString  PasswordSelector,
    responseData   UserData,
    ...
}

ChallengeResponseAlgorithm ::= CHOICE
{
    passwordInTheClear  NULL,
    nonStandardAlgorithm  NonStandardParameter,
    ...
}

ChallengeItem ::= SEQUENCE
{
    responseAlgorithm  ChallengeResponseAlgorithm,
    challengeData     UserData,
    ...
}

ChallengeRequest ::= SEQUENCE
{
    challengeTag  INTEGER,
    challengeSet  SET OF ChallengeItem,
    -- Set of algorithms offered for response
    ...
}

ChallengeResponse ::= SEQUENCE
{
    challengeTag      INTEGER,
    responseAlgorithm  ChallengeResponseAlgorithm,
    -- Specific algorithm selected from the set of
    -- items presented in the ChallengeRequest
    responseItem     ChallengeResponseItem,
    ...
}

PasswordChallengeRequestResponse ::= CHOICE
{
    passwordInTheClear      PasswordSelector,
    challengeRequestResponse  SEQUENCE
    {
        challengeRequest      ChallengeRequest OPTIONAL,
        challengeResponse     ChallengeResponse OPTIONAL,
        ...
    },
    ...
}

ConferenceName ::= SEQUENCE
{
    numeric      SimpleNumericString,
    text         SimpleTextString OPTIONAL,
}

```

```

    ...,
    unicodeText    TextString OPTIONAL
}

ConferenceNameSelector ::= CHOICE
{
    numeric        SimpleNumericString,
    text           SimpleTextString,
    ...,
    unicodeText    TextString
}

ConferenceNameModifier ::= SimpleNumericString

Privilege ::= ENUMERATED
{
    terminate      (0),
    ejectUser      (1),
    add            (2),
    lockUnlock     (3),
    transfer       (4),
    ...
}

TerminationMethod ::= ENUMERATED
{
    automatic (0),
    manual    (1),
    ...
}

ConferencePriorityScheme ::= CHOICE
{
    nonStandardScheme NonStandardParameter,
    ...
}

ConferencePriority ::= SEQUENCE
{
    priority          INTEGER (0..65535),
    scheme            ConferencePriorityScheme,
    ...
}

NodeCategory ::= CHOICE
{
    conventional      NULL,
    counted           NULL,
    anonymous          NULL,
    nonStandardCategory NonStandardParameter,
    ...
}

ConferenceMode ::= CHOICE
{
    conventional-only    NULL,
    counted-only         NULL,
    anonymous-only       NULL,
    conventional-control NULL,
    unrestricted-mode   NULL,
    non-standard-mode   NonStandardParameter,
    ...
}

```

```

NetworkAddress ::= SEQUENCE (SIZE (1..64)) OF CHOICE -- Listed in order of use
{
    aggregatedChannel SEQUENCE
    {
        transferModes SEQUENCE -- One or more
        {
            speech          BOOLEAN,
            voice-band      BOOLEAN,
            digital-56k     BOOLEAN,
            digital-64k     BOOLEAN,
            digital-128k    BOOLEAN,
            digital-192k    BOOLEAN,
            digital-256k    BOOLEAN,
            digital-320k    BOOLEAN,
            digital-384k    BOOLEAN,
            digital-512k    BOOLEAN,
            digital-768k    BOOLEAN,
            digital-1152k   BOOLEAN,
            digital-1472k   BOOLEAN,
            digital-1536k   BOOLEAN,
            digital-1920k   BOOLEAN,
            packet-mode     BOOLEAN,
            frame-mode      BOOLEAN,
            atm             BOOLEAN,
            ...
        },
        internationalNumber DiallingString,
        subAddress           SubAddressString OPTIONAL,
        extraDialling       ExtraDiallingString OPTIONAL,
        highLayerCompatibility SEQUENCE
        {
            telephony3kHz    BOOLEAN,
            telephony7kHz    BOOLEAN,
            videotelephony   BOOLEAN,
            videoconference  BOOLEAN,
            audiographic     BOOLEAN,
            audiovisual      BOOLEAN,
            multimedia       BOOLEAN,
            ...
        } OPTIONAL,
        ...
    },
    transportConnection SEQUENCE
    {
        nsapAddress          OCTET STRING (SIZE (1..20)),
        transportSelector    OCTET STRING OPTIONAL
    },
    nonStandard             NonStandardParameter,
    ...
}

MediaList ::= SEQUENCE {
    audio    BOOLEAN,
    video    BOOLEAN,
    data     BOOLEAN,
    ...
}

ChannelAggregationMethod ::= CHOICE {
    h221          NULL,
    h244          NULL,
    iso-iec-13871 NULL,
    -- The actual mode of bonding is dynamically selected according
    -- to the procedures described in ISO/IEC 13871.
}

```



```

    nonStandard    NonStandardParameter,
    ...
}

Profile ::= CHOICE {
    simpleProfile      CHOICE {
        -- Basic transfer modes:
        speech          NULL,      -- Simple telephony
        telephony-3kHz NULL,      -- Rec. G.711
        telephony-7kHz NULL,      -- Rec. G.722
        voice-band     NULL,      -- Modems
        frameRelay     NULL,
        -- T.120-only data profiles (Rec. 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,
            asvd        NULL,
            dsvd        NULL
        },
        t120Data       BOOLEAN
    },
    dsmccDownloadProfile NULL,
    nonStandard       NonStandardParameter,
    ...
}

ExtendedE164NetworkAddress ::= SEQUENCE {
    internationalNumber DiallingString,
    subAddress          SubAddressString OPTIONAL,
    extraDialling      ExtraDiallingString OPTIONAL,
    ...
}

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

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

ISDNConnection ::= SEQUENCE {
    circuitTypes        SET OF CHOICE {
        digital-64k     NULL,
        digital-2x64k   NULL,
        digital-384k    NULL,
        digital-1536    NULL,
        digital-1920k   NULL,
        multirate-base-64k INTEGER (1..30)    -- See Note 1
    },
    networkAddress ExtendedE164NetworkAddress,
    highLayerCompatibility SEQUENCE {

```

```

        -- Those are supported code points for IE HLC of the D
        -- protocol (Rec. Q.931).
        telephony3kHz      BOOLEAN,
        telephony7kHz      BOOLEAN,
        videotelephony     BOOLEAN,
        videoconference    BOOLEAN,
        audiographic       BOOLEAN,
        audiovisual         BOOLEAN,
        multimedia         BOOLEAN,
        ...
    } OPTIONAL,
    ...
}

-- Note 1:    digital-2x64k differs from multirate-base-64k
--           with a multiplier value of 2;
--           in the first case
--           the network is requested an 8 kHz integrity with Restricted
--           Differential Time Delay (RDTD);
--           in the second case
--           the network is requested a Time Slot
--           Sequence integrity (see 4.5.5/Q.931)

CSDNConnection ::= SEQUENCE {
    circuitTypes      SET OF CHOICE {
        digital-56k   NULL,
        digital-64k   NULL
    },
    networkAddressExtendedE164NetworkAddress,
    ...
}

PSDNConnection ::= SEQUENCE {
    networkAddressCHOICE {
        extendedE164NetworkAddress      ExtendedE164NetworkAddress,
        transportAddress                 TransportAddress,
        nonStandard                       NonStandardParameter
    },
    ...
}

ATMConnection ::= SEQUENCE {
    networkAddress      CHOICE {
        extendedE164     ExtendedE164NetworkAddress,
        nsapAddress      TransportAddress,
        -- this case is reserved for NSAPs only: the
        -- optional transport selector shall never be used
        nonStandard      NonStandardParameter
    },
    maxTransferRate     INTEGER (0..MAX) OPTIONAL,
    -- in cells per seconds
    ...
}

NetworkConnection ::= CHOICE {
    gstnConnection      GSTNConnection,
    isdnConnection      ISDNConnection,
    csdnConnection      CSDNConnection,
    psdnConnection      PSDNConnection,
    atmConnection       ATMConnection,
    extendedE164NetworkAddress      ExtendedE164NetworkAddress,
    -- NOTE - LAN connections and leased
    transportAddress     TransportAddress,
    -- lines (Rec. G.703/G.704) may be

```

```

        -- covered by one of these
nonStandard          NonStandardParameter,
...
}

NetworkAddressV2 ::= SET OF SEQUENCE {
    networkConnection CHOICE {
        singleConnection      NetworkConnection,
        aggregatedConnections SEQUENCE {
            connectionList    SET (SIZE(1..30)) OF CHOICE {
                isdnConnection ISDNConnection,
                csdnConnection CSDNConnection,
                ...
            },
            aggregationMethods SET OF ChannelAggregationMethod OPTIONAL,
            ...
        }
    },
    profiles          SET OF Profile OPTIONAL,
    mediaConcerned   MediaList OPTIONAL,
    ...
}

NodeType ::= ENUMERATED
{
    terminal          (0),
    multiportTerminal (1),
    mcu              (2),
    ...
}

NodeProperties ::= SEQUENCE
{
    managementDevice    BOOLEAN,
        -- Is the node a device such as a reservation system
    peripheralDevice    BOOLEAN,
        -- Is the node a peripheral to a primary node
    ...
}

AsymmetryIndicator ::= CHOICE
{
    callingNode    NULL,
    calledNode    NULL,
    unknown        INTEGER (0..4294967295)
        -- Uniformly distributed 32-bit random number
}

AlternativeNodeID ::= CHOICE
{
    h243NodeID    OCTET STRING (SIZE (2)),
    ...
}

ConferenceDescriptor ::= SEQUENCE
{
    conferenceName          ConferenceName,
    conferenceNameModifier ConferenceNameModifier OPTIONAL,
    conferenceDescription   TextString OPTIONAL,
    lockedConference        BOOLEAN,
    passwordInTheClearRequired BOOLEAN,
    networkAddress          NetworkAddress OPTIONAL,
    ...,
}

```

```

    defaultConferenceFlag      BOOLEAN,
    conferenceMode             ConferenceMode
}

NodeRecord ::= SEQUENCE
{
    superiorNode               UserID OPTIONAL,
    -- Not present only for the Top GCC Provider
    nodeType                   NodeType,
    nodeProperties              NodeProperties,
    nodeName                   TextString OPTIONAL,
    participantsList           SEQUENCE OF TextString OPTIONAL,
    siteInformation            TextString OPTIONAL,
    networkAddress             NetworkAddress OPTIONAL,
    alternativeNodeID          AlternativeNodeID OPTIONAL,
    userData                   UserData OPTIONAL,
    ...,
    nodeCategory               NodeCategory OPTIONAL,
    networkAddressV2           NetworkAddressV2 OPTIONAL
}

SessionKey ::= SEQUENCE
{
    applicationProtocolKey     Key,
    sessionID                  ChannelID OPTIONAL
}

ChannelType ::= ENUMERATED
{
    static                      (0),
    dynamicMulticast            (1),
    dynamicPrivate              (2),
    dynamicUserId               (3)
}

ApplicationRecord ::= SEQUENCE
{
    applicationActive           BOOLEAN,
    -- Active/Inactive flag
    conductingOperationCapable BOOLEAN,
    -- Maximum one per node per session
    startupChannel              ChannelType OPTIONAL,
    applicationUserID           UserID OPTIONAL,
    -- User ID assigned to the Application Protocol Entity
    nonCollapsingCapabilities  SET OF SEQUENCE
    {
        capabilityID           CapabilityID,
        applicationData         OCTET STRING OPTIONAL
    } OPTIONAL,
    ...
}

CapabilityID ::= CHOICE
{
    standard                    INTEGER (0..65535),
    -- Assigned by Application Protocol specifications
    nonStandard                 Key
}

CapabilityClass ::= CHOICE
{
    logical                     NULL,
    unsignedMin                 INTEGER (0..MAX), -- Capability value
    unsignedMax                 INTEGER (0..MAX), -- Capability value
}

```

```

    ...
}

EntityID ::= INTEGER (0..65535)

ApplicationInvokeSpecifier ::= SEQUENCE
{
    sessionKey          SessionKey,
    expectedCapabilitySet SET OF SEQUENCE
                        {
                            capabilityID      CapabilityID,
                            capabilityClass    CapabilityClass,
                            ...
                        } OPTIONAL,
    startupChannel      ChannelType OPTIONAL,
    mandatoryFlag       BOOLEAN,
                        -- TRUE indicates required Application Protocol Entity
    ...
}

RegistryKey ::= SEQUENCE
{
    sessionKey          SessionKey,
    resourceID          OCTET STRING (SIZE (0..64))
}

RegistryItem ::= CHOICE
{
    channelID          DynamicChannelID,
    tokenID            DynamicTokenID,
    parameter          OCTET STRING (SIZE (0..64)),
    vacant             NULL,
    ...
}

RegistryEntryOwner ::= CHOICE
{
    owned              SEQUENCE
                        {
                            nodeID      UserID,          -- Node ID of the owning node
                            entityID    EntityID        -- Entity ID of the owning
                        },
                        -- Application Protocol Entity
    notOwned           NULL
                        -- There is no current owner
}

RegistryModificationRights ::= ENUMERATED
{
    owner      (0),
    session    (1),
    public     (2)
}

-- =====
-- Part 2: PDU Messages
-- =====

UserIDIndication ::= SEQUENCE
{
    tag          INTEGER,
    ...
}

ConferenceCreateRequest ::= SEQUENCE
{
    -- MCS-Connect-Provider request user data

```

```

conferenceName      ConferenceName,
convenerPassword    Password OPTIONAL,
password            Password OPTIONAL,
lockedConference    BOOLEAN,
listedConference    BOOLEAN,
conductibleConference  BOOLEAN,
terminationMethod   TerminationMethod,
conductorPrivileges SET OF Privilege OPTIONAL,
conductedPrivileges SET OF Privilege OPTIONAL,
nonConductedPrivileges SET OF Privilege OPTIONAL,
conferenceDescription TextString OPTIONAL,
callerIdentifier     TextString OPTIONAL,
userData            UserData OPTIONAL,
...,
conferencePriority   ConferencePriority OPTIONAL,
conferenceMode       ConferenceMode OPTIONAL
}

```

```

ConferenceCreateResponse ::= SEQUENCE
{
  -- MCS-Connect-Provider response user data
  nodeID      UserID, -- Node ID of the sending node
  tag         INTEGER,
  result      ENUMERATED
  {
    success                (0),
    userRejected           (1),
    resourcesNotAvailable  (2),
    rejectedForSymmetryBreaking (3),
    lockedConferenceNotSupported (4),
    ...
  },
  userData      UserData OPTIONAL,
  ...
}

```

```

ConferenceQueryRequest ::= SEQUENCE
{
  -- MCS-Connect-Provider request user data
  nodeType      NodeType,
  asymmetryIndicator AsymmetryIndicator OPTIONAL,
  userData      UserData OPTIONAL,
  ...
}

```

```

ConferenceQueryResponse ::= SEQUENCE
{
  -- MCS-Connect-Provider response user data
  nodeType      NodeType,
  asymmetryIndicator AsymmetryIndicator OPTIONAL,
  conferenceList SET OF ConferenceDescriptor,
  result        ENUMERATED
  {
    success                (0),
    userRejected           (1),
    ...
  },
  userData      UserData OPTIONAL,
  ...,
  waitForInvitationFlag BOOLEAN OPTIONAL,
  noUnlistedConferenceFlag BOOLEAN OPTIONAL
}

```

```

ConferenceJoinRequest ::= SEQUENCE
{
  -- MCS-Connect-Provider request user data as well as
  -- MCS-Send-Data on Node ID Channel of Top GCC sent
  -- by the receiver of the MCS-Connect-Provider

```

```

conferenceName      ConferenceNameSelector OPTIONAL,
                    -- Required when part of MCS-Connect-Provider
conferenceNameModifier ConferenceNameModifier OPTIONAL,
tag                 INTEGER OPTIONAL,
                    -- Filled in when sent on Node ID Channel of Top GCC
password            PasswordChallengeRequestResponse OPTIONAL,
convenerPassword    PasswordSelector OPTIONAL,
callerIdentifier     TextString OPTIONAL,
userData            UserData OPTIONAL,
...
nodeCategory        NodeCategory OPTIONAL
}

```

```

ConferenceJoinResponse ::= SEQUENCE
{
  -- MCS-Connect-Provider response user data as well as
  -- MCS-Send-Data on Node ID Channel of
  -- the receiver of the MCS-Connect-Provider
nodeID              UserID OPTIONAL,
                    -- Node ID of directly connected node only
topNodeID           UserID,
                    -- Node ID of Top GCC Provider
tag                 INTEGER,
conferenceNameAlias ConferenceNameSelector OPTIONAL,
passwordInTheClearRequired BOOLEAN,
lockedConference    BOOLEAN,
listedConference    BOOLEAN,
conductibleConference BOOLEAN,
terminationMethod  TerminationMethod,
conductorPrivileges SET OF Privilege OPTIONAL,
                    -- No privilege shall be listed more than once
conductedPrivileges SET OF Privilege OPTIONAL,
                    -- No privilege shall be listed more than once
nonConductedPrivileges SET OF Privilege OPTIONAL,
                    -- No privilege shall be listed more than once
conferenceDescription TextString OPTIONAL,
password             PasswordChallengeRequestResponse OPTIONAL,
result              ENUMERATED
                    {
success              (0),
userRejected        (1),
invalidConference   (2),
invalidPassword     (3),
invalidConvenerPassword (4),
challengeResponseRequired (5),
invalidChallengeResponse (6),
...
                    },
userData            UserData OPTIONAL,
...
nodeCategory        NodeCategory OPTIONAL,
conferenceMode       ConferenceMode OPTIONAL
}

```

```

ConferenceInviteRequest ::= SEQUENCE
{
  -- MCS-Connect-Provider request user data
conferenceName      ConferenceName,
nodeID              UserID, -- Node ID of the sending node
topNodeID           UserID, -- Node ID of Top GCC Provider
tag                 INTEGER,
passwordInTheClearRequired BOOLEAN,
lockedConference    BOOLEAN,
listedConference    BOOLEAN,
conductibleConference BOOLEAN,
terminationMethod  TerminationMethod,
}

```

```

conductorPrivileges      SET OF Privilege OPTIONAL,
  -- No privilege shall be listed more than once
conductedPrivileges      SET OF Privilege OPTIONAL,
  -- No privilege shall be listed more than once
nonConductedPrivileges   SET OF Privilege OPTIONAL,
  -- No privilege shall be listed more than once
conferenceDescription    TextString OPTIONAL,
callerIdentifier          TextString OPTIONAL,
userData                  UserData OPTIONAL,
...,
conferencePriority       ConferencePriority OPTIONAL,
nodeCategory             NodeCategory OPTIONAL,
conferenceMode           ConferenceMode OPTIONAL
}

```

```

ConferenceInviteResponse ::= SEQUENCE
{
  -- MCS-Connect-Provider response user data
  result      ENUMERATED
  {
    success      (0),
    userRejected (1),
    ...
  },
  userData    UserData OPTIONAL,
  ...
}

```

```

ConferenceAddRequest ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of Top GCC or
  -- Node ID Channel of Adding MCU if specified
  networkAddress      NetworkAddress,
  requestingNode      UserID,
  tag                 INTEGER,
  addingMCU           UserID OPTIONAL,
  userData            UserData OPTIONAL,
  ...,
  nodeCategory        NodeCategory OPTIONAL,
  networkAddressV2    NetworkAddressV2
}

```

```

ConferenceAddResponse ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of requester
  tag                 INTEGER,
  result              ENUMERATED
  {
    success              (0),
    invalidRequester    (1),
    invalidNetworkType  (2),
    invalidNetworkAddress (3),
    addedNodeBusy       (4),
    networkBusy         (5),
    noPortsAvailable    (6),
    connectionUnsuccessful (7),
    ...
  },
  userData            UserData OPTIONAL,
  ...
}

```

```

ConferenceLockRequest ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of Top GCC
  -- No parameters
  ...
}

```



```

ConferenceLockResponse ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of requester
  result    ENUMERATED
    {
      success            (0),
      invalidRequester  (1),
      alreadyLocked     (2),
      ...
    },
  ...
}

ConferenceLockIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  -- or MCS-Send-Data on Node ID Channel
  -- No parameters
  ...
}

ConferenceUnlockRequest ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of Top GCC
  -- No parameters
  ...
}

ConferenceUnlockResponse ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of requester
  result    ENUMERATED
    {
      success            (0),
      invalidRequester  (1),
      alreadyUnlocked   (2),
      ...
    },
  ...
}

ConferenceUnlockIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  -- or MCS-Send-Data on Node ID Channel
  -- No parameters
  ...
}

ConferenceTerminateRequest ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of Top GCC
  reason    ENUMERATED
    {
      userInitiated      (0),
      timedConferenceTermination (1),
      ...
    },
  ...
}

ConferenceTerminateResponse ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of requester
  result    ENUMERATED
    {
      success            (0),
      invalidRequester  (1),
      ...
    },
}

```

```

}
...
}

ConferenceTerminateIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  reason      ENUMERATED
    {
      userInitiated          (0),
      timedConferenceTermination (1),
      ...
    },
  ...
}

ConferenceEjectUserRequest ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of Top GCC
  nodeToEject  UserID, -- Node ID of the node to eject
  reason       ENUMERATED
    {
      userInitiated (0),
      ...
    },
  ...
}

ConferenceEjectUserResponse ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of requester
  nodeToEject  UserID, -- Node ID of the node to eject
  result       ENUMERATED
    {
      success          (0),
      invalidRequester (1),
      invalidNode      (2),
      ...
    },
  ...
}

ConferenceEjectUserIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  nodeToEject  UserID, -- Node ID of the node to eject
  reason       ENUMERATED
    {
      userInitiated          (0),
      higherNodeDisconnected (1),
      higherNodeEjected      (2),
      ...
    },
  ...
}

ConferenceTransferRequest ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of Top GCC
  conferenceName      ConferenceNameSelector,
  -- Name of conference to transfer to
  conferenceNameModifier ConferenceNameModifier OPTIONAL,
  networkAddress      NetworkAddress OPTIONAL,
  transferringNodes   SET (SIZE (1..65536)) OF UserID OPTIONAL,
  password            PasswordSelector OPTIONAL,
  ...,
  networkAddressV2    NetworkAddressV2 OPTIONAL
}

```

```

ConferenceTransferResponse ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel of requester
  conferenceName      ConferenceNameSelector,
  -- Name of conference to transfer to
  conferenceNameModifier ConferenceNameModifier OPTIONAL,
  transferringNodes   SET (SIZE (1..65536)) OF UserID OPTIONAL,
  result              ENUMERATED
  {
    success            (0),
    invalidRequester   (1),
    ...
  },
  ...
}

```

```

ConferenceTransferIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  conferenceName      ConferenceNameSelector,
  -- Name of conference to transfer to
  conferenceNameModifier ConferenceNameModifier OPTIONAL,
  networkAddress      NetworkAddress OPTIONAL,
  transferringNodes   SET (SIZE (1..65536)) OF UserID OPTIONAL,
  -- List of Node IDs,
  -- not present if destined for all nodes
  password            PasswordSelector OPTIONAL,
  ...,
  networkAddressV2    NetworkAddressV2 OPTIONAL
}

```

```

RosterUpdateIndication ::= SEQUENCE
{
  -- MCS-Send-Data on Node ID Channel or
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  fullRefresh         BOOLEAN,
  -- Conference Roster and all
  -- ApplicationProtocol Sessions refreshed
  nodeInformation     SEQUENCE
  {
    nodeRecordList CHOICE
    {
      noChange        NULL,
      refresh         SET (SIZE (1..65536)) OF SEQUENCE
        -- One for each node in the conference;
        -- no node shall be listed more than once
        {
          nodeID      UserID, -- Node ID of the node
          nodeRecord   NodeRecord
        },
      update         SET (SIZE (1..65536)) OF SEQUENCE
        -- One for each node changing its node record;
        -- no node shall be listed more than once
        {
          nodeID      UserID, -- Node ID of the node
          nodeUpdate   CHOICE
          {
            addRecord    NodeRecord,
            replaceRecord NodeRecord,
            removeRecord NULL,
            ...
          }
        }
    },
    ...
  },
  rosterInstanceNumber INTEGER (0..65535),
  nodesAdded           BOOLEAN,
}

```

```

        -- Nodes have been added since last instance
nodesRemoved          BOOLEAN,
        -- Nodes have been removed since last instance
    ...
},
applicationInformation SET (SIZE (0..65535)) OF SEQUENCE
    -- One for each Application Protocol Session;
    -- all Application Protocol Sessions if full refresh;
    -- no Application Protocol Session shall be
    -- listed more than once
{
    sessionKey          SessionKey,
    applicationRecordList CHOICE
    {
        noChange        NULL,
        refresh          SET (SIZE (0..65535)) OF SEQUENCE
            -- One for each node with the
            -- Application Protocol Session enrolled;
            -- no node shall be listed more than once
        {
            nodeID        UserID,
                -- Node ID of node
            entityID      EntityID,
                -- ID for this Application Protocol Entity at this node
            applicationRecord ApplicationRecord
        },
        update            SET (SIZE (1..65536)) OF SEQUENCE
            -- One for each node modifying its Application Record;
            -- no node shall be listed more than once
        {
            nodeID        UserID,
                -- Node ID of node
            entityID      EntityID,
                -- ID for this Application Protocol Entity at this node
            applicationUpdate CHOICE
            {
                addRecord      ApplicationRecord,
                replaceRecord   ApplicationRecord,
                removeRecord    NULL,
                ...
            }
        },
        ...
    },
    applicationCapabilitiesList CHOICE
    {
        noChange        NULL,
        refresh          SET OF SEQUENCE
            {
                capabilityID    CapabilityID,
                capabilityClass  CapabilityClass,
                numberOfEntities INTEGER (1..65536),
                -- Number of Application Protocol Entities
                -- which issued the capability
                ...
            },
        ...
    },
    rosterInstanceNumber INTEGER (0..65535),
    peerEntitiesAdded     BOOLEAN,
        -- Peer Entities have been added since last instance
    peerEntitiesRemoved   BOOLEAN,
        -- Peer Entities have been removed since last instance
    ...
}

```

```

    },
    ...
}

ApplicationInvokeIndication ::= SEQUENCE
{
    -- MCS-Send-Data or MCS-Uniform-Send-Data
    -- on GCC-Broadcast-Channel or Node ID Channel
    applicationProtocolEntiyList  SET (SIZE (1..65536)) OF ApplicationInvokeSpecifier,
    destinationNodes              SET (SIZE (1..65536)) OF UserID OPTIONAL,
    -- List of Node IDs,
    -- not present if destined for all nodes
    ...
}

RegistryRegisterChannelRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID                      EntityID,
    key                           RegistryKey,
    channelID                     DynamicChannelID,
    ...
}

RegistryAssignTokenRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID                      EntityID,
    key                           RegistryKey,
    ...
}

RegistrySetParameterRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID                      EntityID,
    key                           RegistryKey,
    parameter                     OCTET STRING (SIZE (0..64)),
    modificationRights            RegistryModificationRights OPTIONAL,
    ...
}

RegistryRetrieveEntryRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID                      EntityID,
    key                           RegistryKey,
    ...
}

RegistryDeleteEntryRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID                      EntityID,
    key                           RegistryKey,
    ...
}

RegistryMonitorEntryRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID                      EntityID,
    key                           RegistryKey,
    ...
}

RegistryMonitorEntryIndication ::= SEQUENCE
{
    -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
    key                           RegistryKey,
    item                          RegistryItem,
    -- Contents: channel, token, parameter, or empty
}

```

```

    owner          RegistryEntryOwner,
    modificationRights  RegistryModificationRights OPTIONAL,
    ...
}

RegistryAllocateHandleRequest ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of Top GCC
    entityID      EntityID,
    numberOfHandles  INTEGER (1..1024),
    ...
}

RegistryAllocateHandleResponse ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of requester
    entityID      EntityID,
    numberOfHandles  INTEGER (1..1024),
    firstHandle   Handle,
    result        ENUMERATED
        {
            successful          (0),
            noHandlesAvailable (1),
            ...
        },
    ...
}

RegistryResponse ::= SEQUENCE
{
    -- MCS-Send-Data on Node ID Channel of requester
    entityID      EntityID,
    -- Entity ID of the requesting Application Protocol Entity
    primitiveType  ENUMERATED
        {
            registerChannel    (0),
            assignToken        (1),
            setParameter        (2),
            retrieveEntry       (3),
            deleteEntry         (4),
            monitorEntry        (5),
            ...
        },
    key           RegistryKey,
    -- Database index
    item          RegistryItem,
    -- Contents: channel, token, parameter, or vacant
    owner        RegistryEntryOwner,
    modificationRights  RegistryModificationRights OPTIONAL,
    result       ENUMERATED
        {
            successful          (0),
            belongsToOther     (1),
            tooManyEntries     (2),
            inconsistentType   (3),
            entryNotFound       (4),
            entryAlreadyExists (5),
            invalidRequester   (6),
            ...
        },
    ...
}

ConductorAssignIndication ::= SEQUENCE
{
    -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
    conductingNode  UserID,
    ...
}

```

```

}

ConductorReleaseIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  -- No parameters
  ...
}

ConductorPermissionAskIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  grantFlag          BOOLEAN,
  -- TRUE to request permission grant, FALSE to release
  ...
}

ConductorPermissionGrantIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  permissionList     SEQUENCE (SIZE (0..65535)) OF UserID,
  -- Node ID of nodes granted permission
  waitingList        SEQUENCE (SIZE (1..65536)) OF UserID OPTIONAL,
  -- Node ID of nodes waiting for permission
  ...
}

ConferenceTimeRemainingIndication ::= SEQUENCE
{
  -- MCS-Send-Data on GCC-Broadcast-Channel
  timeRemaining      Time,
  nodeID             UserID OPTIONAL,
  ...
}

ConferenceTimeInquireIndication ::= SEQUENCE
{
  -- MCS-Send-Data on GCC-Convener-Channel
  nodeSpecificTimeFlag  BOOLEAN,
  -- FALSE for conference-wide, TRUE for node-specific
  ...
}

ConferenceTimeExtendIndication ::= SEQUENCE
{
  -- MCS-Send-Data on GCC-Convener-Channel
  timeToExtend        Time,
  nodeSpecificTimeFlag  BOOLEAN,
  -- FALSE for conference-wide, TRUE for node-specific
  ...
}

ConferenceAssistanceIndication ::= SEQUENCE
{
  -- MCS-Uniform-Send-Data on GCC-Broadcast-Channel
  userData            UserData OPTIONAL,
  ...
}

TextMessageIndication ::= SEQUENCE
{
  message             TextString,
  -- MCS-Send-Data or MCS-Uniform-Send-Data
  -- on GCC-Broadcast-Channel or Node ID Channel
  ...
}

RosterRefreshRequest ::= SEQUENCE
{
  nodeID              UserID,
  nodeCategory        NodeCategory,
  fullRefresh         BOOLEAN,
}

```

```

sendConferenceRoster    BOOLEAN    OPTIONAL,
applicationList        SEQUENCE
{
    applicationKeyList SET OF SEQUENCE
    {
        applicationProtocolKey Key,
        nonStandardParameter  NonStandardParameter OPTIONAL,
        ...
    },
    nonStandardParameter      NonStandardParameter OPTIONAL,
    ...
} OPTIONAL,
sessionList            SEQUENCE
{
    sessionKeyList       SET OF SEQUENCE
    {
        sessionKey        SessionKey,
        nonStandardParameter  NonStandardParameter OPTIONAL,
        ...
    },
    nonStandardParameter      NonStandardParameter OPTIONAL,
    ...
} OPTIONAL,
nonStandardParameter      NonStandardParameter OPTIONAL,
...
}

FunctionNotSupportedResponse ::= SEQUENCE
{
    request      RequestPDU
}

NonStandardPDU ::= SEQUENCE
{
    data          NonStandardParameter,
    ...
}

-- =====
-- Part 3: Messages sent as MCS-Connect-Provider user data
-- =====

ConnectData ::= SEQUENCE
{
    t124IdentifierKey,
    -- This shall be set to the value {itu-t recommendation t 124 version(0) 1}
    connectPDU      OCTET STRING
}

ConnectGCCPDU ::= CHOICE
{
    conferenceCreateRequest      ConferenceCreateRequest,
    conferenceCreateResponse     ConferenceCreateResponse,
    conferenceQueryRequest       ConferenceQueryRequest,
    conferenceQueryResponse      ConferenceQueryResponse,
    conferenceJoinRequest        ConferenceJoinRequest,
    conferenceJoinResponse       ConferenceJoinResponse,
    conferenceInviteRequest      ConferenceInviteRequest,
    conferenceInviteResponse     ConferenceInviteResponse,
    ...
}

```



```
-- =====
-- Part 4: Messages sent using MCS-Send-Data or MCS-Uniform-Send-Data
-- =====
```

```
GCCPDU ::= CHOICE
```

```
{
    request                RequestPDU,
    response                ResponsePDU,
    indication              IndicationPDU
}
```

```
RequestPDU ::= CHOICE
```

```
{
    conferenceJoinRequest    ConferenceJoinRequest,
    conferenceAddRequest     ConferenceAddRequest,
    conferenceLockRequest    ConferenceLockRequest,
    conferenceUnlockRequest  ConferenceUnlockRequest,
    conferenceTerminateRequest ConferenceTerminateRequest,
    conferenceEjectUserRequest ConferenceEjectUserRequest,
    conferenceTransferRequest ConferenceTransferRequest,
    registryRegisterChannelRequest RegistryRegisterChannelRequest,
    registryAssignTokenRequest RegistryAssignTokenRequest,
    registrySetParameterRequest RegistrySetParameterRequest,
    registryRetrieveEntryRequest RegistryRetrieveEntryRequest,
    registryDeleteEntryRequest RegistryDeleteEntryRequest,
    registryMonitorEntryRequest RegistryMonitorEntryRequest,
    registryAllocateHandleRequest RegistryAllocateHandleRequest,
    nonStandardRequest       NonStandardPDU,
    ...
}
```

```
ResponsePDU ::= CHOICE
```

```
{
    conferenceJoinResponse    ConferenceJoinResponse,
    conferenceAddResponse     ConferenceAddResponse,
    conferenceLockResponse    ConferenceLockResponse,
    conferenceUnlockResponse  ConferenceUnlockResponse,
    conferenceTerminateResponse ConferenceTerminateResponse,
    conferenceEjectUserResponse ConferenceEjectUserResponse,
    conferenceTransferResponse ConferenceTransferResponse,
    registryResponse          RegistryResponse,
    registryAllocateHandleResponse RegistryAllocateHandleResponse,
    functionNotSupportedResponse FunctionNotSupportedResponse,
    nonStandardResponse       NonStandardPDU,
    ...
}
```

```
IndicationPDU ::= CHOICE
```

```
{
    userIDIndication          UserIDIndication,
    conferenceLockIndication  ConferenceLockIndication,
    conferenceUnlockIndication ConferenceUnlockIndication,
    conferenceTerminateIndication ConferenceTerminateIndication,
    conferenceEjectUserIndication ConferenceEjectUserIndication,
    conferenceTransferIndication ConferenceTransferIndication,
    rosterUpdateIndication    RosterUpdateIndication,
    applicationInvokeIndication ApplicationInvokeIndication,
    registryMonitorEntryIndication RegistryMonitorEntryIndication,
    conductorAssignIndication ConductorAssignIndication,
    conductorReleaseIndication ConductorReleaseIndication,
    conductorPermissionAskIndication ConductorPermissionAskIndication,
    conductorPermissionGrantIndication ConductorPermissionGrantIndication,
    conferenceTimeRemainingIndication ConferenceTimeRemainingIndication,
}
```

```

conferenceTimeInquireIndication      ConferenceTimeInquireIndication,
conferenceTimeExtendIndication        ConferenceTimeExtendIndication,
conferenceAssistanceIndication        ConferenceAssistanceIndication,
textMessageIndication                 TextMessageIndication,
nonStandardIndication                 NonStandardPDU,
...
}
END

```

## 9 Utilisation du service de communication multipoint

Toutes les communications du service GCC se feront au moyen du service de communication multipoint (MCS, *multipoint communication service*) tel que ce service est spécifié dans [UIT-T T.122]. Le présent paragraphe indique en détail comment le service GCC utilise les services MCS, l'attribution de canal, l'attribution de jeton et les priorités de données.

### 9.1 Services MCS

Le service GCC présuppose les services MCS donnés par le Tableau 9-1. Toutes les primitives et tous les paramètres marqués par un "M" sont utilisés par des composants obligatoires du service GCC. Les items marqués par un "O" ne sont utilisés que par des parties facultatives du service GCC.

**Tableau 9-1 – Services MCS utilisés par le service GCC**

Primitives	Utilisation	Paramètres	Utilisation
demande MCS-Connect-Provider	M	Adresse appelante	O
indication MCS-Connect-Provider	M	Sélecteur de domaine appelant	M
réponse MCS-Connect-Provider	M	Adresse appelée	O
confirmation MCS-Connect-Provider	M	Sélecteur de domaine appelé	–
		Indicateur montée/descente	M
		Paramètres de domaine	M
		Qualité de service	M
		Résultat	M
		Données utilisateur	M
demande MCS-Disconnect-Provider	M	Cause	M
indication MCS-Disconnect-Provider	M		
demande MCS-Attach-User	M	Sélecteur de domaine	M
confirmation MCS-Attach-User	M	Résultat	M
		Identificateur d'utilisateur	M
demande MCS-Detach-User	M	Cause	M
indication MCS-Detach-User	M	Identificateur d'utilisateur	M
demande MCS-Channel-Join	M	Canal de branchement	M
confirmation MCS-Channel-Join	M	Résultat	M
demande MCS-Channel-Leave	–	Canal à quitter	–
indication MCS-Channel-Leave	–	Cause	–
demande MCS-Channel-Convenc	–	Résultat	–
confirmation MCS-Channel-Convenc	–	Canal	–
demande MCS-Channel-Disband	–	Canal	–
indication MCS-Channel-Disband	–	Cause	–

**Tableau 9-1 – Services MCS utilisés par le service GCC**

<b>Primitives</b>	<b>Utilisation</b>	<b>Paramètres</b>	<b>Utilisation</b>
demande MCS-Channel-Admit indication MCS-Channel-Admit	– –	Canal Identificateur d'utilisateur du gestionnaire Liste d'identificateurs d'utilisateur	– – –
demande MCS-Channel-Expel indication MCS-Channel-Expel	– –	Canal Liste d'identificateurs d'utilisateur Cause	– – –
demande MCS-Send-Data indication MCS-Send-Data	M M	Priorité Identificateur de canal Identificateur d'utilisateur émetteur Données	M M M M
demande MCS-Uniform-Send-Data indication MCS-Uniform-Send-Data	M M	Priorité Identificateur de canal Identificateur d'utilisateur émetteur Données	M M M M
demande MCS-Token-Grab confirmation MCS-Token-Grab	O O	Identificateur de jeton Résultat	O O
demande MCS-Token-Inhibit confirmation MCS-Token-Inhibit	O O	Identificateur de jeton Résultat	O O
demande MCS-Token-Give indication MCS-Token-Give réponse MCS-Token-Give confirmation MCS-Token-Give	O M M O	Identificateur d'utilisateur du donataire Identificateur d'utilisateur à recevoir Identificateur de jeton Résultat	M  O  M M
demande MCS-Token-Please indication MCS-Token-Please	O O	Identificateur d'utilisateur du demandeur Identificateur de jeton	O O
demande MCS-Token-Release confirmation MCS-Token-Release	O O	Identificateur de jeton Résultat	O O
demande MCS-Token-Test confirmation MCS-Token-Test	O O	Identificateur de jeton Etat du jeton	O O

## 9.2 Affectation de canaux

Le service GCC réserve deux canaux statiques pour son utilisation exclusive. Sur le premier canal, GCC-CHANNEL-0 sont branchés tous les fournisseurs GCC de la conférence. Le deuxième canal, GCC-CHANNEL-1 sert uniquement au branchement de l'organisateur de conférence. Tout fournisseur GCC se branche également sur le canal d'identificateur de nœud qui lui est attribué par le service MCS. Le Tableau 9-2 récapitule l'utilisation de canaux faite par le service GCC.

**Tableau 9-2 – Utilisation de canaux par le service GCC**

Identificateur de canal	Type	Mnémonique	Description
GCC-CHANNEL-0	Statique	GCC-Broadcast-Channel (canal de diffusion)	Communication diffusée par tout fournisseur GCC vers l'ensemble des fournisseurs GCC de la conférence.
GCC-CHANNEL-1	Statique	GCC-Convener-Channel (canal de l'organisateur)	Communication adressée par un fournisseur GCC à l'organisateur de la conférence.
GCC-CHANNEL-2	Statique	GCC-Conventional-Broadcast-Channel (canal de diffusion pour nœuds conventionnels)	Communication adressée par un fournisseur GCC à tous les fournisseurs GCC d'une conférence.
GCC-CHANNEL-3	Statique	GCC-Counted-Broadcast-Channel (canal de diffusion pour nœuds comptés)	Communication adressée par un fournisseur GCC de nœud conventionnel à tout autre fournisseur GCC de nœud conventionnel de la conférence. Utilisé pour communiquer l'activité des nœuds comptés aux nœuds conventionnels.
–	Dynamique	Canal d'identificateur de nœud	Communication adressée par un fournisseur GCC quelconque d'une conférence au fournisseur GCC d'un nœud particulier d'une conférence. Tout nœud est identifié par son identificateur de nœud.

### 9.3 Attribution de jetons

Le service GCC réserve un seul jeton pour son utilisation exclusive. Ce jeton GCC-TOKEN-0 est utilisé comme jeton de présidence de conférence. Le service GCC réserve également les jetons de 16 384 à 65 535 comme jetons dynamiques qu'il attribue aux entités de protocole d'application. Le Tableau 9-3 récapitule l'utilisation des jetons faite par le fournisseur GCC.

**Tableau 9-3 – Utilisation de jetons par le service GCC**

Identificateur de jeton	Mnémonique	Description
GCC-TOKEN-0	Conference-Conductorship-Token (jeton de présidence)	Saisi par un fournisseur GCC dans le but de devenir le directeur de la conférence.
16 384 à 65 535	Jetons dynamiques	Attribués par le service GCC à des entités de protocole d'application ayant émis la primitive GCC-Registry-Assign-Token.

### 9.4 Utilisation des services de transmission de données MCS

Le Tableau 9-4 donne un résumé de l'utilisation des services de transmission de données MCS-Send-Data (envoi de données MCS) et MCS-Uniform-Send-Data (envoi uniforme de données MCS) par chaque GCCPDU. Pour chaque GCCPDU, le type de service de données MCS utilisé, l'identificateur de canal et la priorité requise sont indiqués.

**Tableau 9-4 – Utilisation des transmissions de données MCS pour les GCCPDU**

GCCPDU	Canal	Type de primitive de transmission	Priorité
UserIDIndication	Canal d'identificateur du nœud connecté directement	MCS-Send-Data	Maximale
ConferenceCreateRequest	–	–	–
ConferenceCreateResponse	–	–	–
ConferenceQueryRequest	–	–	–
ConferenceQueryResponse	–	–	–
ConferenceJoinRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Maximale
ConferenceJoinResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Maximale
ConferenceInviteRequest	–	–	–
ConferenceInviteResponse	–	–	–
ConferenceAddRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
	Canal d'identificateur du nœud de la MCU chargée de l'addition	MCS-Send-Data	Haute
ConferenceAddResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Haute
ConferenceLockRequest	Canal d'identificateur du nœud de fournisseur GCC sommital	MCS-Send-Data	Haute
ConferenceLockResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Haute
ConferenceLockIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
	Canal d'identificateur du nœud	MCS-Send-Data	Haute
ConferenceUnlockRequest	Canal d'identificateur du nœud fournisseur GCC sommital	MCS-Send-Data	Haute
ConferenceUnlockResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Haute
ConferenceUnlockIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
	Canal d'identificateur du nœud	MCS-Send-Data	Haute
ConferenceTerminateRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute

**Tableau 9-4 – Utilisation des transmissions de données MCS pour les GCCPDU**

<b>GCCPDU</b>	<b>Canal</b>	<b>Type de primitive de transmission</b>	<b>Priorité</b>
ConferenceTerminateResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Haute
ConferenceTerminateIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
ConferenceEjectUserRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Maximale
ConferenceEjectUserResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Haute
ConferenceEjectUserIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Maximale
ConferenceTransferRequest	Canal d'identificateur du nœud fournisseur GCC sommital	MCS-Send-Data	Haute
ConferenceTransferResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Haute
ConferenceTransferIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
RosterUpdateIndication	Canal d'identificateur du nœud récepteur	MCS-Send-Data	Haute
	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
ApplicationInvokeIndication	Canal GCC-Broadcast-Channel	MCS-Send-Data ou MCS-Uniform-Send-Data	Haute
	Canal d'identificateur du nœud récepteur	MCS-Send-Data	Haute
RegistryRegisterChannelRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
RegistryAssignTokenRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
RegistrySetParameterRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
RegistryRetrieveEntryRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
RegistryDeleteEntryRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
RegistryMonitorEntryRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute

**Tableau 9-4 – Utilisation des transmissions de données MCS pour les GCCPDU**

GCCPDU	Canal	Type de primitive de transmission	Priorité
RegistryMonitorEntryIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
RegistryAllocateHandleRequest	Canal d'identificateur du nœud du fournisseur GCC sommital	MCS-Send-Data	Haute
RegistryAllocateHandleResponse	Canal d'identificateur du nœud du demandeur	MCS-Send-Data	Haute
RegistryResponse	Canal d'identificateur du nœud du demandeur	MCS-Send-Data	Haute
ConductorAssignIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Maximale
ConductorReleaseIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Maximale
ConductorPermissionAskIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
ConductorPermissionGrantIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Maximale
ConferenceTimeRemainingIndication	Canal GCC-Broadcast-Channel	MCS-Uniform-Send-Data	Haute
ConferenceTimeInquireIndication	Canal GCC-Convener-Channel	MCS-Send-Data	Haute
ConferenceTimeExtendIndication	Canal GCC-Convener-Channel	MCS-Send-Data	Haute
ConferenceAssistanceIndication	Canal GCC-Broadcast-Channel	MCS-Send-Data ou MCS-Uniform-Send-Data	Haute
TextMessageIndication	Canal GCC-Broadcast-Channel	MCS-Send-Data ou MCS-Uniform-Send-Data	Haute
	Canal d'identificateur du nœud récepteur	MCS-Send-Data	Haute
FunctionNotSupportedResponse	Canal d'identificateur du nœud demandeur	MCS-Send-Data	Comme dans la demande
PDU hors normes	Indéfini	Indéfini	Indéfinie

### 9.5 Codage des PDU dans les primitives MCS

Toutes les PDU définies dans la présente Recommandation sont codées et placées dans le champ de données de l'une des primitives MCS. Les primitives pouvant être utilisées à cet effet sont MCS-Connect-Provider (connexion de fournisseur MCS), MCS-Send-Data (envoi de données MCS) ou MCS-Uniform-Send-Data (envoi uniforme de données MCS). Dans tous les cas, la chaîne binaire résultant du codage ASN.1 est placée dans la chaîne d'octets utilisée par le service MCS de telle façon que l'ordre séquentiel des bits du premier au dernier corresponde à l'ordre décroissant des poids des bits des octets dans lesquels ils sont placés.

Dans le cas des primitives MCS-Connect-Provider, la PDU n'est pas placée directement dans le paramètre "données" de la primitive MCS mais elle est encapsulée dans une structure définie par le § 9.6. Dans ce cas, la description ci-dessus s'applique à la structure externe plutôt qu'à la PDU elle-même.

## **9.6 Format de paramètre "données utilisateur" pour la primitive MCS-Connect-Provider**

Le paramètre "données utilisateur" des primitives MCS-Connect-Provider et MCS-Disconnect-Provider utilisées dans la présente Recommandation prend en charge le format défini par [UIT-T T.122]. Ce format fournit une identification unique du contrôleur MCS. Le contrôleur MCS peut être d'un type normalisé (se conformant par exemple à la présente Recommandation) ou d'un type non normalisé. Afin de permettre de faire la distinction, le paramètre "données utilisateur" est d'un type abstrait ASN.1 contenant l'identificateur du type de contrôleur MCS suivi de la PDU elle-même. Le champ identificateur est du type Key, structure de choix entre un identificateur d'objet ou un identificateur non normalisé H.221. Le type abstrait global, ainsi que ses composants, est défini au § 8.7. La structure globale est codée en utilisant les règles de codage compact (variante alignée) définies par [UIT-T X.691].

Un contrôleur MCS se conformant à la présente Recommandation utilisera un en-tête spécifiant le choix de l'identificateur d'objet avec une valeur {itu-t recommendation t 124 version(0) 1}.

La PDU n'est pas codée directement dans cette structure mais placée dans un champ de données du type chaîne d'octets. La PDU est codée séparément selon les règles de codage compact (variante alignée) comme pour les autres PDU dans la présente Recommandation. Le codage est tel que l'ordre séquentiel des bits du premier au dernier corresponde à l'ordre décroissant des poids des bits des octets dans lesquels ils sont placés. La PDU et la structure dont elle fait partie sont codées séparément afin de permettre à de futures Recommandations ou à des contrôleurs MCS non normalisés d'utiliser d'autres codages des PDU tout en fournissant un identificateur unique du type de contrôleur MCS qui sera toujours codé d'une manière uniforme en utilisant les règles de codage compact.

## **9.7 Interprétation du sélecteur de domaine MCS**

La présente Recommandation fait les hypothèses suivantes concernant le sélecteur de domaine lorsqu'elle utilise la primitive MCS-Connect-Provider (connexion de fournisseur MCS).

Le sélecteur de domaine appelant est toujours conforme au sélecteur de domaine local, lui même égal à l'identificateur de conférence. L'identificateur de conférence est également utilisé par les protocoles d'application lorsqu'ils s'attachent au service MCS.

La présente Recommandation n'utilise pas le sélecteur de domaine appelé. Au lieu de cela, elle suppose que le fournisseur MCS transmet les indications MCS-Connect-Provider au point d'accès de contrôle MCSAP sans s'intéresser à leur contenu. Il est de la responsabilité du fournisseur GCC de déterminer si une connexion doit être établie et dans ce cas, de déterminer quel doit être le domaine associé à la connexion. Ceci est fait d'une manière légèrement différente pour chacune des quatre primitives T.124 qui génèrent des demandes MCS-Connect-Provider:

- GCC-Conference-Create: il n'existe pas dans ce cas, avant la réception de l'indication MCS-Connect-Provider, de domaine auquel peut se rattacher la connexion. Lorsqu'il reçoit une indication MCS-Connect-Provider valide, un fournisseur GCC doit créer un nouveau domaine MCS et indiquer au service MCS que c'est ce domaine qui doit être associé à la nouvelle connexion. Le sélecteur de domaine est alors choisi par le fournisseur GCC comme identificateur de conférence local. Le moyen utilisé pour réaliser ces deux actions est un sujet local non spécifié par [UIT-T T.122];



- GCC-Conference-Query: dans ce cas, aucune connexion n'est établie puisque la réponse MCS-Connect-Provider indique que la demande a été rejetée avec la cause "utilisateur rejeté". Il en résulte qu'il n'est pas nécessaire d'associer de MCS à la demande;
- GCC-Conference-Join: dans ce cas, il existe déjà un domaine sur le nœud récepteur. Si la demande d'entrée en conférence est acceptée par le fournisseur GCC, celui-ci doit indiquer au fournisseur MCS quel est le domaine auquel doit être associée la connexion. Le nom de conférence et le modificateur éventuel de nom de conférence sont utilisés par le fournisseur GCC pour déterminer le domaine. Le sélecteur de domaine est l'identificateur local de cette conférence. Le moyen utilisé pour indiquer le domaine au fournisseur MCS est un sujet local non spécifié par [UIT-T T.122];
- GCC-Conference-Invite: comme dans le cas de la création de conférence, le nœud récepteur ne connaît pas de domaine préexistant. Si l'invitation est acceptée, le fournisseur GCC choisit un identificateur de conférence qui sera utilisé comme sélecteur de domaine. Le fournisseur MCS doit être informé de la création de ce domaine sur le nœud et la connexion en question doit être associée au domaine. Comme dans le cas de la création de conférence, le moyen utilisé pour réaliser ces deux actions est un sujet local non spécifié par [UIT-T T.122].

## Annexe A

### Affectations statiques d'identificateur de canal et de jeton

(Cette annexe fait partie intégrante de la présente Recommandation)

#### A.1 Affectations statiques d'identificateur de canal

Le Tableau A.1 donne la liste des allocations numériques des identificateurs statiques de canal pour les canaux statiques dont l'utilisation est attribuée à la présente Recommandation. Il est prévu que l'allocation numérique d'identificateurs de canal statique soit centralisée dans [UIT-T T.120], mais elle est donnée ici en attendant l'achèvement de [UIT-T T.120].

**Tableau A.1 – Affectations d'identificateurs de canal statique**

Nom symbolique	Identificateur de canal
GCC-CHANNEL-0	1
GCC-CHANNEL-1	2
GCC-CHANNEL-2	3
GCC-CHANNEL-3	4

#### A.2 Affectations statiques d'identificateur de jeton

Le Tableau A.2 donne la liste des allocations numériques des identificateurs statiques de jeton pour les jetons statiques dont l'utilisation est attribuée à la présente Recommandation. Il est prévu que l'allocation numérique d'identificateurs de jeton statique doit être centralisée dans [UIT-T T.120], mais elle est donnée ici en attendant l'achèvement de [UIT-T T.120].

**Tableau A.2 – Affectations d'identificateurs de jeton statique**

Nom symbolique	Identificateur de jeton
GCC-TOKEN-0	1

## Annexe B

### Affectations d'identificateurs d'objet

(Cette annexe fait partie intégrante de la présente Recommandation)

**Tableau B.1**

<b>Valeur de l'identificateur d'objet</b>	<b>Description</b>
{itu-t recommendation t 124 version(0) 1}	Le présent identificateur d'objet est utilisé pour indiquer la version de la présente Recommandation en usage pour le contrôleur MCS. A l'heure actuelle, une unique version normalisée est définie.

## Annexe C

### Paramètre Adresse réseau – Description et utilisation

(Cette annexe fait partie intégrante de la présente Recommandation)

La structure ASN.1 *NetworkAddressV2*, spécifiée au § 8.7 (Définitions des unités GCCPDU) remet en forme et améliore la structure *NetworkAddress*. En dissociant les concepts de types de réseau, de débits de transfert et de profils (ou modes) de transfert, qui sont confondus dans l'organisation unidimensionnelle utilisée par la structure *NetworkAddress*, la nouvelle structure apporte plus de flexibilité pour décrire une connexion réseau allant vers un nœud de conférence; elle résout également un certain nombre de problèmes de présignalisation qui n'avaient pas encore été examinés.

La structure *NetworkAddressV2* a été ajoutée à la structure *NetworkAddress* afin d'assurer la rétrocompatibilité avec d'anciennes implémentations T.120 ne l'utilisant pas. L'UIT-T a l'intention de tenir à jour la structure *NetworkAddressV2* au fur et à mesure de l'évolution de la T.124, tout en laissant diminuer l'utilisation de la structure *NetworkAddress* jusqu'à ce qu'elle puisse être retirée de la norme. La présente annexe constitue une étape intermédiaire vers cet objectif.

Pour garantir l'interfonctionnement entre dispositifs T.120 utilisant et n'utilisant pas la structure *NetworkAddressV2*, les règles suivantes doivent être respectées par les implémentations:

- 1) tout équipement T.120 qui est conforme à la présente version de la T.124, à savoir la Recommandation T.124 révisée – 1998, DOIT faire usage de la structure *NetworkAddressV2* CONJOINTEMENT AVEC la structure *NetworkAddress*;
- 2) chaque fournisseur T.124 utilisant la structure *NetworkAddressV2* est responsable de la fourniture d'une structure *NetworkAddress* correspondant le plus étroitement possible aux caractéristiques inhérentes à la structure *NetworkAddressV2*;
- 3) chaque fournisseur T.124 utilisant la structure *NetworkAddressV2* et devant, à un moment donné, traiter et retransmettre une unité GCCPDU contenant des paramètres d'adressage réseau et recevant l'information d'adressage réseau d'un nœud ne faisant pas usage de la structure *NetworkAddressV2*, est responsable de l'insertion, dans l'unité PDU retransmise, d'une structure *NetworkAddressV2* correspondant le plus étroitement possible à la description acheminée par la structure *NetworkAddress* (voir la Note);

NOTE – Un exemple courant est celui du mécanisme de transfert internodal de conférence qui est décrit au § 8.2.11; lorsque le fournisseur GCC sommital utilise la structure *NetworkAddressV2* et reçoit l'unité PDU *ConferenceTransferRequest* d'un nœud qui n'utilise que la structure *NetworkAddress*, ce fournisseur doit produire et insérer une structure *NetworkAddressV2* dans l'unité PDU *ConferenceTransferIndication* correspondante, qu'il diffusera dans le canal GCC-Broadcast-Channel.

- 4) les fournisseurs T.124 recevant des unités GCCPDU contenant une structure *NetworkAddressV2* et ne la prenant pas en charge n'en tiendront pas compte sans le mentionner.

## Appendice I

### Relation entre systèmes T.120 et H.243 dans des conférences H.320

(Cet appendice ne fait pas partie intégrante de la présente Recommandation)

#### I.1 Introduction

La série T.120 vise à reprendre finalement les responsabilités de la Rec. UIT-T H.243 concernant l'établissement de conférences multipoints en liaison avec des systèmes audio/vidéo/données H.320. Entre-temps, les Recommandations UIT-T H.243 et T.120 continueront à coexister. Le présent appendice présente des recommandations sur la façon dont les systèmes H.243 et T.120 devront entrer en relation pendant la longue phase de transition pour passer des commandes de conférence H.243 aux commandes de conférence T.120.

NOTE – Le paragraphe 15/H.243 impose des prescriptions normatives aux terminaux et aux ponts MCU à cet égard.

#### I.2 Sélection des conférences et protection par mot de passe

Les systèmes T.124 aussi bien que H.243 offrent des moyens permettant d'établir une connexion logique entre un terminal et une conférence particulière, hébergée dans un pont MCU. Ces deux Recommandations prévoient également un mécanisme de protection par mot de passe pour vérifier le droit d'un nœud à entrer dans la conférence souhaitée. Il est envisagé que les deux Recommandations, T.124 et H.243, continuent à être utilisées de concert pendant un certain temps. Les sous-paragraphes ci-après décrivent comment il conviendra d'utiliser ces deux Recommandations UIT-T T.124 et H.243 lors du processus d'établissement d'une conférence.

L'on part du principe qu'au plus une seule conférence H.243 est associée à chaque conférence T.124 et inversement. Si donc un terminal entre dans une des deux conférences, le choix de l'autre est donc prédéterminé. Si cette hypothèse n'est pas vérifiée, des procédures différentes pourront s'appliquer.

Afin de sélectionner une conférence à laquelle participer (dans le cas où le pont MCU autorise un tel choix) et afin d'assurer la protection par mot de passe, un pont MCU doit choisir d'utiliser le mécanisme T.124 ou le mécanisme H.243 mais pas les deux en même temps. Le choix du mécanisme à utiliser peut dépendre du fait que la conférence est à accès entrant ou à accès sortant, du fait qu'il peut être fait séparément pour chaque pont MCU et du fait qu'il peut dépendre des capacités du terminal de connexion à chaque accès. Dans le cas de terminaux prenant en charge [UIT-T T.120], il est préféré que les mécanismes T.124 soient utilisés à ces fins, plutôt que le mécanisme H.243.

##### I.2.1 Etablissement d'une conférence T.124

Le pont MCU peut choisir d'utiliser les mécanismes T.124 pour l'établissement d'une conférence pour un terminal donné. Le pont MCU peut, par exemple, permettre au terminal de choisir parmi une liste de conférences à joindre en fournissant cette liste dans la réponse GCC-Conference-Query et en permettant la sélection d'une conférence au moyen de la primitive GCC-Conference-Join. Le pont MCU peut, au contraire, avoir une conférence prédéterminée qu'il souhaite voir joindre par le terminal qui se connecte. Dans ce cas, il doit régler le fanion indicateur de conférence par défaut, contenu dans le descripteur de conférence correspondant de la réponse GCC-Conference-Query, de manière à préciser que le terminal qui se connecte doit entrer dans la conférence indiquée. Dans un cas comme dans l'autre, si le pont MCU a besoin d'une protection par mot de passe, les mécanismes de protection par mot de passe de la T.124 doivent être utilisés (mot de passe dans l'état "réponse en clair requise" ou "réponse à l'épreuve requise") afin de vérifier que le terminal qui se connecte est autorisé à entrer dans la conférence.

En variante, comme cela peut être le cas pour un accès sortant prédéterminé, si le choix de la conférence est déjà déterminé et que la protection par mot de passe ne soit pas requise, le pont MCU peut décider de prendre l'initiative et d'inviter le terminal à entrer dans une conférence T.124 en émettant une demande GCC-Conference-Invite. Dans ce cas, il doit régler le fanion indicateur d'attente d'invitation, contenu dans la réponse GCC-Conference-Query, de façon à signaler que le terminal qui se connecte ne doit pas prendre l'initiative de joindre ou de créer une conférence T.124.

Une fois que le pont MCU a reçu une demande GCC-Conference-Join assortie d'un nom de conférence et du mot de passe correspondant, ou s'il a décidé de prendre l'initiative et d'inviter un terminal à entrer dans une conférence T.124, ce terminal peut alors commencer à recevoir les données audio et vidéo associées à la conférence H.243 correspondante.

Dans ce cas, le pont MCU ne doit pas utiliser le mécanisme H.243 pour demander un mot de passe au terminal qui se connecte. Le mécanisme H.243 peut continuer à être utilisé pour la commande vidéo et pour d'autres fonctions.

### **I.2.2 Etablissement d'une conférence H.243**

En variante, le pont MCU peut décider d'utiliser le mécanisme H.243 pour établir une conférence. Dans ce cas, si un terminal se présente à un pont MCU, celui-ci peut utiliser le mot de passe H.243 pour déterminer à quelle conférence ce terminal appartient. Une fois que le mot de passe H.243 correct a été fourni, le terminal peut commencer à recevoir les données audio et vidéo associées à la conférence.

Etant donné que la H.243 prévoit un mécanisme de mot de passe, la prescription d'un mot de passe additionnel pour conférence T.124 est dans ce cas superflue (en supposant qu'une seule conférence T.124 soit autorisée conjointement avec la conférence H.243). Le pont MCU peut décider de prendre l'initiative et d'inviter le terminal à entrer dans la conférence T.124 associée à la conférence H.243, en émettant une primitive de demande GCC-Conference-Invite. Si le terminal le lui demande, le pont MCU doit régler l'indicateur d'attente d'invitation contenu dans la réponse GCC-Conference-Query de façon à signaler que le terminal n'a pas besoin de prendre l'initiative de joindre ou de créer une conférence T.124. En variante, le pont MCU peut attendre que le terminal émette une demande GCC-Conference-Join. Dans ce cas, si le terminal le lui demande, le pont MCU doit renvoyer un seul descripteur de conférence dans sa réponse GCC-Conference-Query, avec l'indicateur de conférence par défaut activé et l'indicateur de mot de passe à l'état de libération requise désactivé.

Les terminaux qui prennent en charge les systèmes T.120 doivent permettre l'utilisation du mécanisme de mot de passe H.243 pour l'établissement d'une conférence. Cette méthode doit être considérée comme étant d'utilisation temporaire par les ponts MCU en attendant que ceux-ci prennent en charge les procédures d'établissement de conférence T.124 décrites au sous-paragraphe précédent.

### **I.3 Variante d'identificateur nodal**

L'utilisation de la variante d'identificateur nodal de la T.124 a été envisagée pour la numérotation des terminaux H.243 car elle permet aux nœuds d'une conférence audio/vidéo/données d'exploiter les deux mécanismes T.120 et H.243 afin de mettre en correspondance les informations de répertoire de conférence T.124 avec les informations des sites H.243.

Lorsqu'ils émettent une primitive de demande GCC-Conference-Announce-Presence, les nœuds doivent inclure leur numéro de terminal H.243 assigné dans le champ de variante d'identificateur nodal. Si le numéro de terminal H.243 d'un nœud est réassigné au cours de la conférence, ce nœud doit réémettre une demande GCC-Conference-Announce-Presence avec le nouveau numéro de terminal dans le champ de variante d'identificateur nodal.

NOTE – Cette procédure nécessite que chaque terminal mémorise le contenu du code H.230 d'indication et de commande (C&I) d'affectation de numéro de terminal (TIA) reçu le plus récemment.

La variante d'identificateur nodal est codée sur un champ de deux octets, dont le premier doit contenir l'identificateur H.243 de pont MCU (M) et dont le second doit contenir l'identificateur H.243 du terminal (T).







## SÉRIES DES RECOMMANDATIONS UIT-T

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