



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

Q.832.3

(01/2001)

SÉRIE Q: COMMUTATION ET SIGNALISATION
Interface Q3

Coordination de l'accès à large bande

Recommandation UIT-T Q.832.3

(Antérieurement Recommandation du CCITT)

RECOMMANDATIONS UIT-T DE LA SÉRIE Q
COMMUTATION ET SIGNALISATION

SIGNALISATION DANS LE SERVICE MANUEL INTERNATIONAL	Q.1–Q.3
EXPLOITATION INTERNATIONALE AUTOMATIQUE ET SEMI-AUTOMATIQUE	Q.4–Q.59
FONCTIONS ET FLUX D'INFORMATION DES SERVICES DU RNIS	Q.60–Q.99
CLAUSES APPLICABLES AUX SYSTÈMES NORMALISÉS DE L'UIT-T	Q.100–Q.119
SPÉCIFICATIONS DES SYSTÈMES DE SIGNALISATION N° 4 ET N° 5	Q.120–Q.249
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION N° 6	Q.250–Q.309
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION R1	Q.310–Q.399
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION R2	Q.400–Q.499
COMMULATEURS NUMÉRIQUES	Q.500–Q.599
SPÉCIFICATIONS DU SYSTÈME DE SIGNALISATION N° 7	Q.700–Q.799
INTERFACE Q3	Q.800–Q.849
SYSTÈME DE SIGNALISATION D'ABONNÉ NUMÉRIQUE N° 1	Q.850–Q.999
RÉSEAUX MOBILES TERRESTRES PUBLICS	Q.1000–Q.1099
INTERFONCTIONNEMENT AVEC LES SYSTÈMES MOBILES À SATELLITES	Q.1100–Q.1199
RÉSEAU INTELLIGENT	Q.1200–Q.1699
PRESCRIPTIONS ET PROTOCOLES DE SIGNALISATION POUR LES IMT-2000	Q.1700–Q.1799
RNIS À LARGE BANDE	Q.2000–Q.2999

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

Recommandation UIT-T Q.832.3

Coordination de l'accès à large bande

Résumé

La présente Recommandation définit l'interface X entre le système d'exploitation d'un nœud de service et le système d'exploitation d'un réseau d'accès pour la coordination de la gestion associée aux interfaces de trafic VB5.1 et VB5.2 et aux interfaces VB5 Q3.

Source

La Recommandation Q.832.3 de l'UIT-T, élaborée par la Commission d'études 4 (2001-2004) de l'UIT-T, a été approuvée le 19 janvier 2001 selon la procédure définie dans la Résolution 1 de l'AMNT.

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.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

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

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

© UIT 2001

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

TABLE DES MATIÈRES

		Page
1	Introduction.....	1
1.1	Domaine d'application et objet.....	1
2	Références normatives.....	1
3	Définitions, abréviations et conventions.....	1
3.1	Définitions.....	1
	3.1.1 Ressources VB5.....	1
3.2	Abréviations	2
3.3	Conventions	2
4	Aperçu général	3
4.1	Modèles de relations entre entités	3
	4.1.1 Diagramme des relations entre entités	4
4.2	Hiérarchie d'héritage.....	4
5	Définitions formelles	4
5.1	Classes d'objets.....	4
	5.1.1 Notes sur le profil des classes importées	5
	5.1.2 Définition des classes.....	5
5.2	Corrélations de noms	6
	5.2.1 xvb5-managedElementR1	6
5.3	Définition des paquetages	7
5.4	Définition des attributs	7
	5.4.1 xvb5Id (identificateur X-VB5).....	7
5.5	Définition des actions	7
	5.5.1 addAnLoopRequest (demande d'adjonction de boucle AN).....	7
	5.5.2 addLupsRequest (demande d'adjonction de ports LUP).....	7
	5.5.3 addVb5ConnectionRequest (demande d'adjonction de connexion VB5).....	7
	5.5.4 addVb5InterfaceRequest (demande d'adjonction d'interface VB5)	8
	5.5.5 addVb5ProtocolRequest (demande d'adjonction de protocole VB5)	8
	5.5.6 addVb5ProtocolVpRequest (demande d'adjonction de VP de protocole VB5).....	8
	5.5.7 addVb5VcsRequest (demande d'adjonction de voies virtuelles VB5)	8
	5.5.8 addVb5VpsRequest (demande d'adjonction de conduits virtuels VB5).....	9
	5.5.9 anServiceLabelInquiry (recherche d'étiquette de service AN).....	9
	5.5.10 auditVb5ConnectionRequest (demande d'audit de connexion VB5)	9
	5.5.11 auditVb5VpciRequest (demande d'audit d'identificateur VPCI d'interface VB5).....	9
	5.5.12 listLupsRequest (demande d'énumération des ports LUP)	9

5.5.13	listVb5ProtocolDetailsRequest (demande d'énumération des détails de protocole)	10
5.5.14	listVb5InterfacesRequest (demande d'énumération des interfaces VB5).....	10
5.5.15	listVb5VcsRequest (demande d'énumération des voies virtuelles d'interface VB5).....	10
5.5.16	listVb5VpsRequest (demande d'énumération des conduits virtuels VB5) ...	10
5.5.17	removeAnLoopRequest (demande de suppression d'un circuit AN)	11
5.5.18	removeLupsRequest (demande de suppression de ports LUP).....	11
5.5.19	removeVb5ConnectionRequest (demande de suppression d'une connexion VB5).....	11
5.5.20	removeVb5InterfaceRequest (demande de suppression d'interface VB5)....	11
5.5.21	removeVb5ProtocolRequest (demande de suppression de protocole VB5) .	12
5.5.22	removeVb5ProtocolVpRequest (demande de suppression de conduit virtuel de protocole VB5)	12
5.5.23	removeVb5VcsRequest (demande de suppression de voies virtuelles VB5).....	12
5.5.24	removeVb5VpsRequest (demande de suppression de conduits virtuels VB5).....	12
5.5.25	snAccessLabelsInquiry (recherche d'étiquettes d'accès SN).....	13
5.6	Définition des notifications	13
5.6.1	addLupsIndication (indication d'adjonction de ports LUP)	13
5.6.2	addVb5ConnectionIndication (indication d'adjonction de connexion VB5)	13
5.6.3	addVb5InterfaceIndication (indication d'adjonction d'interface VB5).....	13
5.6.4	addVb5ProtocolIndication (indication d'adjonction de protocole VB5)	14
5.6.5	addVb5ProtocolVpIndication (indication d'adjonction de conduit virtuel de protocole VB5)	14
5.6.6	addVb5VcsIndication (indication d'adjonction de voies virtuelles VB5)	14
5.6.7	addVb5VpsIndication (indication d'adjonction de conduits virtuels VB5) ..	14
5.6.8	removeLupsIndication (indication de suppression de ports LUP)	14
5.6.9	removeVb5ConnectionIndication (indication de suppression de connexion VB5).....	15
5.6.10	removeVb5InterfaceIndication (indication de suppression d'interface VB5).....	15
5.6.11	removeVb5ProtocolIndication (indication de suppression de protocole VB5).....	15
5.6.12	removeVb5ProtocolVpIndication (indication de suppression de conduit virtuel de protocole VB5)	15
5.6.13	removeVb5VcsIndication (indication de suppression de voies virtuelles VB5).....	15
5.6.14	removeVb5VpsIndication (indication de suppression de conduits virtuels VB5).....	16
5.6.15	resourceStatusIndication (indication de statut de ressource)	16

	Page
6	Définitions de type 16
7	Piles de protocoles..... 23
	Annexe A – Conditions de gestion 24
A.1	Conditions générales 24
	A.1.1 Configuration coordonnée des conduits virtuels et des voies virtuelles 24
	A.1.2 Contrôle des connexions VPC..... 24
	A.1.3 Coordination des données de configuration des ports 24
	A.1.4 Coordination des valeurs d'identificateur VPCI..... 24
	A.1.5 Cohérence de la configuration..... 24
	A.1.6 Disponibilité des informations 24
A.2	Coordination de l'interface VB5..... 24
	A.2.1 Création..... 24
	A.2.2 Vérification et audit 25
	A.2.3 Modification 25
	A.2.4 Suppression 25
	A.2.5 Configuration de VC et VP 25
	A.2.6 Vérification et audit 26
	A.2.7 Modification des VP et des VC..... 26
	A.2.8 Vérification et audit 26
	A.2.9 Suppression des VP et des VC 26
A.3	Coordination de l'interface UNI 26
	A.3.1 Création..... 26
	A.3.2 Vérification et audit 26
	A.3.3 Modification 26
	A.3.4 Suppression 27
A.4	Conditions de coordination de commande de connexion support à large bande 27
	A.4.1 Voies VC à l'interface VB5..... 27
	A.4.2 Voies virtuelles à l'interface UNI 27
A.5	Gestion des dérangements et de la performance 28
	A.5.1 Signalisation des dérangements..... 28
	A.5.2 Localisation des dérangements..... 28
	Annexe B – Conditions de transaction 28
	Appendice I – Bibliographie..... 49
	Appendice II – Résumé des transactions..... 51

Recommandation UIT-T Q.832.3

Coordination de l'accès à large bande

1 Introduction

1.1 Domaine d'application et objet

La présente Recommandation spécifie l'interface X entre le système d'exploitation (OS, *operations system*) d'un nœud de service (SN, *service node*) et le système d'exploitation (OS) d'un réseau d'accès (AN, *access network*) pour la coordination de la gestion associée aux interfaces de trafic VB5.1 et VB5.2 [2] et [3] et aux interfaces Q3 – VB5 [4] et [5].

Les protocoles existants et les travaux axés sur la définition du modèle objet ont été utilisés dans la mesure du possible. La définition de la fonctionnalité des systèmes d'exploitation du RGT ne relève pas de la présente Recommandation.

Il en est de même pour la gestion de la sécurité.

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.

- [1] UIT-T G.902 (1995), *Recommandation de base sur les réseaux d'accès fonctionnels – Architecture et fonctions, types d'accès, gestion et aspects relatifs aux nœuds de service.*
- [2] UIT-T G.967.1 (1998), *Interfaces V au nœud de service – Spécification du point de référence VB5.1.*
- [3] UIT-T G.967.2 (1999), *Interfaces V au nœud de service – Spécification du point de référence VB5.2.*
- [4] UIT-T Q.832.1 (1998), *Gestion associée aux interfaces VB5.1.*
- [5] UIT-T Q.832.2 (1998), *Gestion associée aux interfaces VB5.2.*
- [6] UIT-T X.721 (1992) | ISO/CEI 10165-2:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: définitions des informations de gestion.*
- [7] IETF RFC 1057(1988): *RPC: Remote Procedure Call Protocol Specification: Version 2* (RPC: spécification du protocole d'appel de procédure distante: version 2).

3 Définitions, abréviations et conventions

3.1 Définitions

La présente Recommandation définit les termes suivants:

3.1.1 Ressources VB5: la gestion des fonctions d'accès usager et des fonctions d'accès de service, offrant respectivement une fonctionnalité d'interface usager-réseau (UNI) et une fonctionnalité d'interface service-nœud (SNI), est considérée dans la présente Recommandation comme étant fondée sur le cadre défini dans l'UIT-T G.902 [1]. Les ressources propres à la transmission restent en

dehors de son domaine d'application. Les ressources VB5 sont désignées dans la présente Recommandation par le terme *ressources*.

Par ailleurs, la présente Recommandation utilise les termes suivants définis dans d'autres Recommandations UIT-T:

UIT-T G.902 [1]: réseau d'accès (AN), fonctions d'accès utilisateur, nœud de service (SN), interface de nœud de service (SNI), fonctions d'accès de service.

UIT-T G.967.1 [2]: port service logique (LSP), port utilisateur logique (LUP), port service physique (PSP, *physical service port*), port utilisateur physique (PUP, *physical user port*), coordination de gestion en temps réel (RTMC, *real time management coordination*), port utilisateur virtuel (VUP, *virtual user port*).

UIT-T G.967.2 [3]: commande de connexion support à large bande (B-BCC)

3.2 Abréviations

La présente Recommandation utilise les abréviations suivantes:

AN	réseau d'accès (<i>access network</i>)
ASN.1	notation de syntaxe abstraite numéro un (<i>abstract syntax notation one</i>)
ATM	mode de transfert asynchrone (<i>asynchronous transfer mode</i>)
B-BCC	commande de connexion support à large bande (<i>broadband bearer connection control</i>)
IETF	groupe de travail d'ingénierie Internet (<i>Internet engineering task force</i>)
LSP	port service logique (<i>logical service port</i>)
LUP	port utilisateur logique (<i>logical user port</i>)
OS	système d'exploitation (<i>operations system</i>)
RFC	demande de commentaire (<i>request for comment</i>)
RGT	réseau de gestion des télécommunications
RTMC	coordination de gestion en temps réel (<i>real time management coordination</i>)
SN	nœud de service (<i>service node</i>)
SNI	interface de nœud de service (<i>service node interface</i>)
TTP	(point de) terminaison de chemin (<i>trail termination point</i>)
UNI	interface utilisateur-réseau (<i>user-network interface</i>)
VC	voie virtuelle (<i>virtual channel</i>)
VP	conduit virtuel (<i>virtual path</i>)
VPC	connexion de conduit virtuel (<i>virtual path connection</i>)
VPCI	identificateur de connexion de conduit virtuel (<i>virtual path connection identifier</i>)

3.3 Conventions

Les objets, leurs caractéristiques et les notations ASN.1 associées, qui sont définis dans la présente Recommandation, sont des noms propres dans lesquels une majuscule indique le début du mot suivant et où les acronymes sont traités comme des mots.

Dans toute la présente Recommandation, tous les nouveaux attributs sont nommés conformément aux directives suivantes:

- le nom d'un attribut se termine par la chaîne "Ptr" si et seulement si la valeur de cet attribut doit désigner un seul objet;
- le nom d'un attribut se termine par la chaîne "PtrList" si et seulement si la valeur de cet attribut doit désigner un ou plusieurs objets;
- le nom d'un attribut se compose du nom d'une classe d'objets suivi de la chaîne "Ptr" si et seulement si la valeur de cet attribut doit désigner une classe d'objets spécifique;
- si un attribut doit désigner différentes classes d'objets, un nom descriptif est donné à cet attribut et une description est donnée dans le module de comportement correspondant;
- le nom d'un attribut se termine par la chaîne "Id" si et seulement si la valeur de cet attribut doit désigner le nom d'un objet, auquel cas il convient que cet attribut figure en tête d'énumération, qu'il utilise un type de nom (NameType en ASN.1) et qu'il ne soit pas utilisé pour acheminer d'autres informations;
- le nom d'un attribut se compose du nom d'une classe d'objets suivi de la chaîne "Id" si et seulement si la valeur de cet attribut doit désigner le nom de la classe d'objets contenant cet attribut.

4 Aperçu général

Les diagrammes de modèle d'informations suivants ont été établis afin de préciser les relations entre les diverses classes d'objets du modèle:

- 1) les modèles de relations entre entités montrent les relations entre les divers objets gérés;
- 2) la hiérarchie d'héritage montre la manière dont les objets gérés sont dérivés les uns des autres (c'est-à-dire les différents chemins d'héritage caractéristiques des divers objets gérés).

Ces diagrammes ne sont donnés qu'à titre explicatif. La spécification formelle en termes de gabarits GDMO et les définitions de type ASN.1 constituent les informations applicables aux implémentations.

4.1 Modèles de relations entre entités

Les conventions suivantes sont utilisées dans les diagrammes:

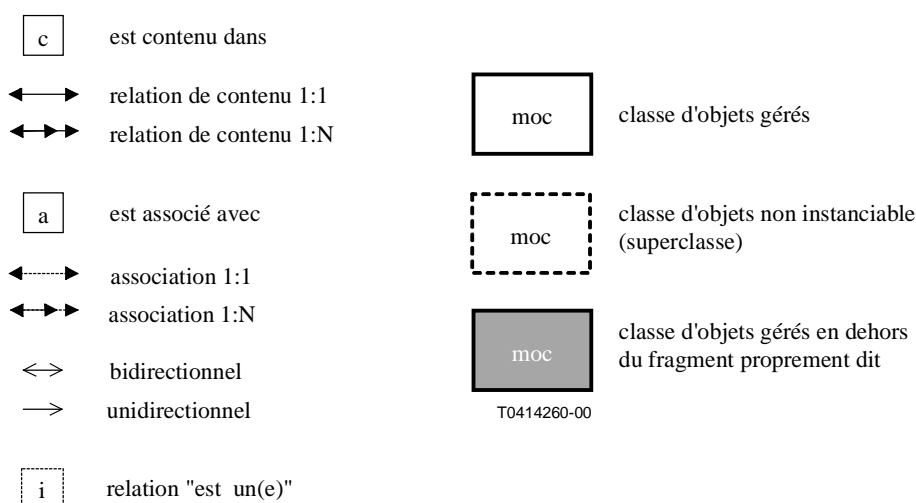


Figure 1/Q.832.3 – Conventions utilisées dans les diagrammes des modèles de relations entre entités

Lorsque le sens des relations de contenu n'est pas évident, il peut être identifié d'une manière implicite du fait que la classe racine est unique.

4.1.1 Diagramme des relations entre entités

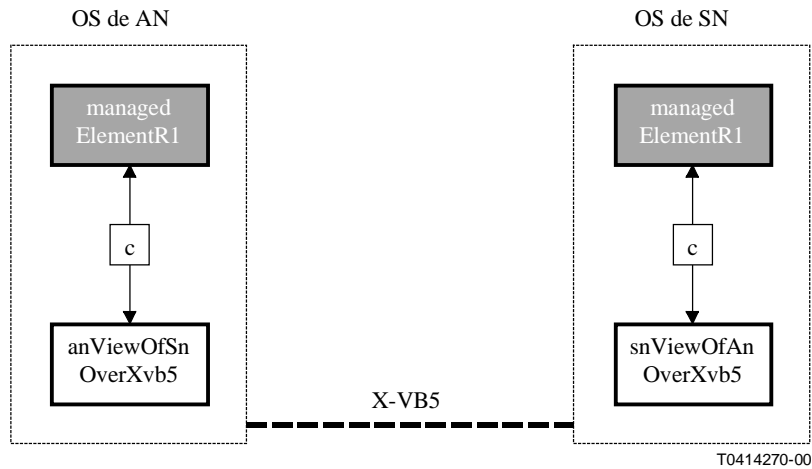


Figure 2/Q.832.3 – Diagramme des relations entre entités

4.2 Hiérarchie d'héritage

La Figure 3 montre les relations d'héritage de l'objet du niveau le plus élevé (le "sommet" dans l'UIT-T X.721) aux objets gérés qui sont définis dans la présente Recommandation.

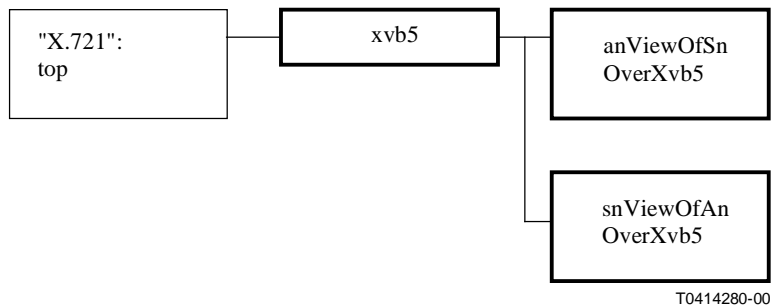


Figure 3/Q.832.3 – Hiérarchie d'héritage

5 Définitions formelles

Le présent paragraphe contient les définitions formelles des classes d'objets gérés, des corrélations de noms, des paquetages généraux, des comportements, des attributs, des actions et des notifications.

5.1 Classes d'objets

Le présent paragraphe spécifie les classes d'objets pour tous les objets gérés utilisés dans le modèle d'information de gestion. Ces classes d'objets sont définies ici même ou par référence à d'autres spécifications. Les classes d'objets gérés qui sont définies ailleurs et qui ne sont utilisées que pour

des raisons de relation de contenu n'ont pas été incluses; elles sont identifiées par corrélation de noms pour les classes spécifiées ici.

5.1.1 Notes sur le profil des classes importées

Il n'y a pas de classes importées.

5.1.2 Définition des classes

5.1.2.1 anViewOfSnOverXvb5 (vue AN du nœud SN par l'interface X-VB5)

```
anViewOfSnOverXvb5 MANAGED OBJECT CLASS
  DERIVED FROM xvb5;
  CHARACTERIZED BY
    anViewOfSnOverXvb5Pkg PACKAGE
      BEHAVIOUR anViewOfSnOverXvb5Beh;
    ACTIONS
      anServiceLabelInquiry;;;
REGISTERED AS {q832-3ManagedObjectClass 1};
```

```
anViewOfSnOverXvb5Beh BEHAVIOUR
  DEFINED AS
    "Cet objet géré représente le côté nœud de service d'une interface X-VB5, du point de vue du système
    d'exploitation du réseau d'accès.";
```

5.1.2.2 snViewOfAnOverXvb5 (vue SN du réseau AN par l'interface X-VB5)

```
snViewOfAnOverXvb5 MANAGED OBJECT CLASS
  DERIVED FROM xvb5;
  CHARACTERIZED BY
    snViewOfAnOverXvb5Pkg PACKAGE
      BEHAVIOUR snViewOfAnOverXvb5Beh;
    ACTIONS
      addAnLoopRequest,
      snAccessLabelsInquiry,
      removeAnLoopRequest;;;
REGISTERED AS {q832-3ManagedObjectClass 2};
```

```
snViewOfAnOverXvb5Beh BEHAVIOUR
  DEFINED AS
    "Cet objet géré représente le côté réseau d'accès d'une interface X-VB5, du point de vue du système
    d'exploitation du nœud de service.";
```

5.1.2.3 xvb5 (X-VB5)

```
xvb5 MANAGED OBJECT CLASS
  DERIVED FROM "Rec. X.721|ISO/IEC 10165-2":top;
  CHARACTERIZED BY
    "ITU-T M.3100":operationalStatePackage,
    xvb5Pkg PACKAGE
      BEHAVIOUR xvb5Beh;
    ATTRIBUTES
      xvb5Id
        GET SET-BY-CREATE,
        " Rec. X.721 | ISO/IEC 10165-2": administrativeState
        GET-REPLACE;
    ACTIONS
      addLupsRequest,
      addVb5ConnectionRequest,
      addVb5InterfaceRequest,
      addVb5ProtocolRequest,
```

```

    addVb5ProtocolVpRequest,
    addVb5VcsRequest,
    addVb5VpsRequest,
    auditVb5ConnectionRequest,
    auditVb5VpciRequest,
    listLupsRequest,
    listVb5ProtocolDetailsRequest,
    listVb5InterfacesRequest,
    listVb5VcsRequest,
    listVb5VpsRequest,
    removeLupsRequest,
    removeVb5ConnectionRequest,
    removeVb5InterfaceRequest,
    removeVb5ProtocolRequest,
    removeVb5ProtocolVpRequest,
    removeVb5VcsRequest,
    removeVb5VpsRequest;
NOTIFICATIONS
    addLupsIndication,
    addVb5ConnectionIndication,
    addVb5InterfaceIndication,
    addVb5ProtocolIndication,
    addVb5ProtocolVpIndication,
    addVb5VcsIndication,
    addVb5VpsIndication,
    removeLupsIndication,
    removeVb5ConnectionIndication,
    removeVb5InterfaceIndication,
    removeVb5ProtocolIndication,
    removeVb5ProtocolVpIndication,
    removeVb5VcsIndication,
    removeVb5VpsIndication,
    resourceStatusIndication,
    " Rec. X.721 | ISO/IEC 10165-2": stateChange,
    " Rec. X.721 | ISO/IEC 10165-2": objectCreation,
    " Rec. X.721 | ISO/IEC 10165-2": objectDeletion;;;
REGISTERED AS {q832-3ManagedObjectClass 3};

```

xvb5Beh BEHAVIOUR

DEFINED AS

"La classe d'objets gérés xvb5 représente les aspects d'une interface X-VB5 qui sont communs aux deux côtés.
La classe xvb5 n'est pas instanciée.";

5.2 Corrélations de noms

5.2.1 xvb5-managedElementR1

xvb5-managedElementR1 NAME BINDING

SUBORDINATE OBJECT CLASS xvb5 AND SUBCLASSES;

NAMED BY SUPERIOR OBJECT CLASS "ITU-T Rec. M.3100":managedElementR1 AND SUBCLASSES;

WITH ATTRIBUTE xvb5Id;

CREATE

WITH-REFERENCE-OBJECT,

WITH-AUTOMATIC-INSTANCE-NAMING;

DELETE

DELETES-CONTAINED-OBJECTS;

REGISTERED AS {q832-3NameBinding 1};

5.3 Définition des paquetages

Aucun paquetage n'est défini.

5.4 Définition des attributs

5.4.1 xvb5Id (identificateur X-VB5)

xvb5Id ATTRIBUTE

WITH ATTRIBUTE SYNTAX Q832-3ASN1Module.NameType;
MATCHES FOR EQUALITY;
BEHAVIOUR xvb5IdBeh;
REGISTERED AS {q832-3Attribute 1};

xvb5IdBeh BEHAVIOUR

DEFINED AS

"Cet attribut sert à nommer des instances de la classe d'objets gérés xvb5 et de ses sous-classes.";

5.5 Définition des actions

5.5.1 addAnLoopRequest (demande d'adjonction de boucle AN)

addAnLoopRequest ACTION

BEHAVIOUR addAnLoopRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddAnLoopRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddAnLoopRequestResult;
REGISTERED AS {q832-3Action 1};

addAnLoopRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée par le système d'exploitation du SN afin de demander au système d'exploitation du réseau d'accès de boucler une connexion de façon que les cellules envoyées au réseau d'accès soient renvoyées.";

5.5.2 addLupsRequest (demande d'adjonction de ports LUP)

addLupsRequest ACTION

BEHAVIOUR addLupsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddLupsRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddLupsRequestResult;
REGISTERED AS {q832-3Action 2};

addLupsRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter un port utilisateur logique à une interface VB5.";

5.5.3 addVb5ConnectionRequest (demande d'adjonction de connexion VB5)

addVb5ConnectionRequest ACTION

BEHAVIOUR addVb5ConnectionRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ConnectionRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ConnectionRequestResult;
REGISTERED AS {q832-3Action 3};

addVb5ConnectionRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter une connexion associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.";

5.5.4 addVb5InterfaceRequest (demande d'adjonction d'interface VB5)

addVb5InterfaceRequest ACTION
BEHAVIOUR addVb5InterfaceRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5InterfaceRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5InterfaceRequestResult;
REGISTERED AS {q832-3Action 4};

addVb5InterfaceRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter une interface VB5.";

5.5.5 addVb5ProtocolRequest (demande d'adjonction de protocole VB5)

addVb5ProtocolRequest ACTION
BEHAVIOUR addVb5ProtocolRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolRequestResult;
REGISTERED AS {q832-3Action 5};

addVb5ProtocolRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter un protocole à une interface VB5 existante.";

5.5.6 addVb5ProtocolVpRequest (demande d'adjonction de VP de protocole VB5)

addVb5ProtocolVpRequest ACTION
BEHAVIOUR addVb5ProtocolVpRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolVpRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolVpRequestResult;
REGISTERED AS {q832-3Action 6};

addVb5ProtocolVpRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter un conduit virtuel de protocole à une interface VB5.";

5.5.7 addVb5VcsRequest (demande d'adjonction de voies virtuelles VB5)

addVb5VcsRequest ACTION
BEHAVIOUR addVb5VcsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VcsRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VcsRequestResult;
REGISTERED AS {q832-3Action 7};

addVb5VcsRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter des voies virtuelles à un conduit virtuel qui est associé à une interface VB5.";

5.5.8 addVb5VpsRequest (demande d'adjonction de conduits virtuels VB5)

addVb5VpsRequest ACTION
BEHAVIOUR addVb5VpsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VpsRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VpsRequestResult;
REGISTERED AS {q832-3Action 8};
addVb5VpsRequestBeh BEHAVIOUR
DEFINED AS
"Cette action est utilisée afin de demander au système d'exploitation homologue d'ajouter des conduits virtuels associés à une interface VB5.";

5.5.9 anServiceLabelInquiry (recherche d'étiquette de service AN)

anServiceLabelInquiry ACTION
BEHAVIOUR anServiceLabelInquiryBeh;
MODE CONFIRMED;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AnServiceLabelInquiryResult;
REGISTERED AS {q832-3Action 9};

anServiceLabelInquiryBeh BEHAVIOUR
DEFINED AS
"Cette action est utilisée par le système d'exploitation d'un réseau d'accès afin de rechercher l'étiquette qu'un SN utilise pour le réseau d'accès.";

5.5.10 auditVb5ConnectionRequest (demande d'audit de connexion VB5)

auditVb5ConnectionRequest ACTION
BEHAVIOUR auditVb5ConnectionRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5ConnectionRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5ConnectionRequestResult;
REGISTERED AS {q832-3Action 10};

auditVb5ConnectionRequestBeh BEHAVIOUR
DEFINED AS
"Cette action est utilisée afin de demander au système d'exploitation homologue de contrôler une connexion associée à une interface VB5.";

5.5.11 auditVb5VpciRequest (demande d'audit d'identificateur VPCI d'interface VB5)

auditVb5VpciRequest ACTION
BEHAVIOUR auditVb5VpciRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5VpciRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.AuditVb5VpciRequestResult;
REGISTERED AS {q832-3Action 11};

auditVb5VpciRequestBeh BEHAVIOUR
DEFINED AS
"Cette action est utilisée afin de demander au système d'exploitation homologue de contrôler un identificateur VPCI qui est associé à une interface VB5.";

5.5.12 listLupsRequest (demande d'énumération des ports LUP)

listLupsRequest ACTION
BEHAVIOUR listLupsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListLupsRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListLupsRequestResult;
REGISTERED AS {q832-3Action 12};

listLupsRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'énumérer les ports utilisateur logique associés à une interface VB5 entre un réseau d'accès et un nœud de service que les deux systèmes d'exploitation commandent ensemble.";

5.5.13 listVb5ProtocolDetailsRequest (demande d'énumération des détails de protocole)

listVb5ProtocolDetailsRequest ACTION
BEHAVIOUR listVb5ProtocolDetailsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5ProtocolDetailsRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5ProtocolDetailsRequestResult;
REGISTERED AS {q832-3Action 13};

listVb5ProtocolDetailsRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'énumérer les détails des protocoles d'une interface VB5 entre un réseau d'accès et un nœud de service que les deux systèmes d'exploitation commandent ensemble.";

5.5.14 listVb5InterfacesRequest (demande d'énumération des interfaces VB5)

listVb5InterfacesRequest ACTION
BEHAVIOUR listVb5InterfacesRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5InterfacesRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5InterfacesRequestResult;
REGISTERED AS {q832-3Action 14};

listVb5InterfacesRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'énumérer les identités des interfaces VB5 entre réseau(x) d'accès et le(s) nœud(s) de service que les deux systèmes d'exploitation commandent ensemble.";

5.5.15 listVb5VcsRequest (demande d'énumération des voies virtuelles d'interface VB5)

listVb5VcsRequest ACTION
BEHAVIOUR listVb5VcsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VcsRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VcsRequestResult;
REGISTERED AS {q832-3Action 15};

listVb5VcsRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'énumérer les voies virtuelles associées à une interface VB5.";

5.5.16 listVb5VpsRequest (demande d'énumération des conduits virtuels VB5)

listVb5VpsRequest ACTION
BEHAVIOUR listVb5VpsRequestBeh;
MODE CONFIRMED;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VpsRequestInfo;
WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.ListVb5VpsRequestResult;
REGISTERED AS {q832-3Action 16};

listVb5VpsRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue d'énumérer les conduits virtuels associés à une interface VB5.";

5.5.17 removeAnLoopRequest (demande de suppression d'un circuit AN)

removeAnLoopRequest ACTION

BEHAVIOUR removeAnLoopRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveAnLoopRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveAnLoopRequestResult;

REGISTERED AS {q832-3Action 17};

removeAnLoopRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée par le système d'exploitation du SN afin de demander au système d'exploitation du réseau d'accès de supprimer un circuit d'une connexion.";

5.5.18 removeLupsRequest (demande de suppression de ports LUP)

removeLupsRequest ACTION

BEHAVIOUR removeLupsRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveLupsRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveLupsRequestResult;

REGISTERED AS {q832-3Action 18};

removeLupsRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer un port utilisateur logique d'une interface VB5.";

5.5.19 removeVb5ConnectionRequest (demande de suppression d'une connexion VB5)

removeVb5ConnectionRequest ACTION

BEHAVIOUR removeVb5ConnectionRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ConnectionRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ConnectionRequestResult;

REGISTERED AS {q832-3Action 19};

removeVb5ConnectionRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer une connexion associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.";

5.5.20 removeVb5InterfaceRequest (demande de suppression d'interface VB5)

removeVb5InterfaceRequest ACTION

BEHAVIOUR removeVb5InterfaceRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5InterfaceRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5InterfaceRequestResult;

REGISTERED AS {q832-3Action 20};

removeVb5InterfaceRequestBeh BEHAVIOUR
DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer une interface VB5.";

5.5.21 removeVb5ProtocolRequest (demande de suppression de protocole VB5)

removeVb5ProtocolRequest ACTION

BEHAVIOUR removeVb5ProtocolRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolRequestResult;

REGISTERED AS {q832-3Action 21};

removeVb5ProtocolRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer un protocole d'une interface VB5.";

5.5.22 removeVb5ProtocolVpRequest (demande de suppression de conduit virtuel de protocole VB5)

removeVb5ProtocolVpRequest ACTION

BEHAVIOUR removeVb5ProtocolVpRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolVpRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolVpRequestResult;

REGISTERED AS {q832-3Action 22};

removeVb5ProtocolVpRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer le conduit virtuel de protocole d'une interface VB5.";

5.5.23 removeVb5VcsRequest (demande de suppression de voies virtuelles VB5)

removeVb5VcsRequest ACTION

BEHAVIOUR removeVb5VcsRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VcsRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VcsRequestResult;

REGISTERED AS {q832-3Action 23};

removeVb5VcsRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer des voies virtuelles d'un conduit virtuel qui est associé à une interface VB5.";

5.5.24 removeVb5VpsRequest (demande de suppression de conduits virtuels VB5)

removeVb5VpsRequest ACTION

BEHAVIOUR removeVb5VpsRequestBeh;

MODE CONFIRMED;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VpsRequestInfo;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VpsRequestResult;

REGISTERED AS {q832-3Action 24};

removeVb5VpsRequestBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée afin de demander au système d'exploitation homologue de supprimer des conduits virtuels qui sont associés à une interface VB5.";

5.5.25 snAccessLabelsInquiry (recherche d'étiquettes d'accès SN)

snAccessLabelsInquiry ACTION

BEHAVIOUR snAccessLabelsInquiryBeh;

MODE CONFIRMED;

WITH REPLY SYNTAX Q832-3ASN1DefinedTypesModule.SnAccessLabelsInquiryResult;

REGISTERED AS {q832-3Action 25};

snAccessLabelsInquiryBeh BEHAVIOUR

DEFINED AS

"Cette action est utilisée par le système d'exploitation of an SN afin de rechercher les étiquettes d'accès qu'utilise un réseau d'accès pour le SN et l'interface VB5.";

5.6 Définition des notifications

5.6.1 addLupsIndication (indication d'adjonction de ports LUP)

addLupsIndication NOTIFICATION

BEHAVIOUR addLupsIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddLupsIndicationInfo;

REGISTERED AS {q832-3Notification 1};

addLupsIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue l'adjonction d'un port utilisateur logique à une interface VB5.";

5.6.2 addVb5ConnectionIndication (indication d'adjonction de connexion VB5)

addVb5ConnectionIndication NOTIFICATION

BEHAVIOUR addVb5ConnectionIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ConnectionIndicationInfo;

REGISTERED AS {q832-3Notification 2};

addVb5ConnectionIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue l'adjonction d'une connexion associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.";

5.6.3 addVb5InterfaceIndication (indication d'adjonction d'interface VB5)

addVb5InterfaceIndication NOTIFICATION

BEHAVIOUR addVb5InterfaceIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5InterfaceIndicationInfo;

REGISTERED AS {q832-3Notification 3};

addVb5InterfaceIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à informer le système d'exploitation homologue qu'une nouvelle interface VB5 a été ajoutée.";

5.6.4 addVb5ProtocolIndication (indication d'adjonction de protocole VB5)

addVb5ProtocolIndication NOTIFICATION
BEHAVIOUR addVb5ProtocolIndicationBeh;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolIndicationInfo;
REGISTERED AS {q832-3Notification 4};

addVb5ProtocolIndicationBeh BEHAVIOUR
DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue l'adjonction d'un protocole à une interface VB5 existante.";

5.6.5 addVb5ProtocolVpIndication (indication d'adjonction de conduit virtuel de protocole VB5)

addVb5ProtocolVpIndication NOTIFICATION
BEHAVIOUR addVb5ProtocolVpIndicationBeh;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5ProtocolVpIndicationInfo;
REGISTERED AS {q832-3Notification 5};

addVb5ProtocolVpIndicationBeh BEHAVIOUR
DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue l'adjonction d'un protocole VP à une interface VB5.";

5.6.6 addVb5VcsIndication (indication d'adjonction de voies virtuelles VB5)

addVb5VcsIndication NOTIFICATION
BEHAVIOUR addVb5VcsIndicationBeh;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VcsIndicationInfo;
REGISTERED AS {q832-3Notification 6};

addVb5VcsIndicationBeh BEHAVIOUR
DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue l'adjonction de voies virtuelles à un conduit virtuel qui est associé à une interface VB5.";

5.6.7 addVb5VpsIndication (indication d'adjonction de conduits virtuels VB5)

addVb5VpsIndication NOTIFICATION
BEHAVIOUR addVb5VpsIndicationBeh;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.AddVb5VpsIndicationInfo;
REGISTERED AS {q832-3Notification 7};

addVb5VpsIndicationBeh BEHAVIOUR
DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue l'adjonction de conduits virtuels qui sont associés à une interface VB5.";

5.6.8 removeLupsIndication (indication de suppression de ports LUP)

removeLupsIndication NOTIFICATION
BEHAVIOUR removeLupsIndicationBeh;
WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveLupsIndicationInfo;
REGISTERED AS {q832-3Notification 8};

removeLupsIndicationBeh BEHAVIOUR
DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression d'un port utilisateur logique d'une interface VB5.";

5.6.9 removeVb5ConnectionIndication (indication de suppression de connexion VB5)

removeVb5ConnectionIndication NOTIFICATION

BEHAVIOUR removeVb5ConnectionIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ConnectionIndicationInfo;

REGISTERED AS {q832-3Notification 9};

removeVb5ConnectionIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression d'une connexion associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.";

5.6.10 removeVb5InterfaceIndication (indication de suppression d'interface VB5)

removeVb5InterfaceIndication NOTIFICATION

BEHAVIOUR removeVb5InterfaceIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5InterfaceIndicationInfo;

REGISTERED AS {q832-3Notification 10};

removeVb5InterfaceIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression d'une interface VB5.";

5.6.11 removeVb5ProtocolIndication (indication de suppression de protocole VB5)

removeVb5ProtocolIndication NOTIFICATION

BEHAVIOUR removeVb5ProtocolIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolIndicationInfo;

REGISTERED AS {q832-3Notification 11};

removeVb5ProtocolIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression d'un protocole d'une interface VB5.";

5.6.12 removeVb5ProtocolVpIndication (indication de suppression de conduit virtuel de protocole VB5)

removeVb5ProtocolVpIndication NOTIFICATION

BEHAVIOUR removeVb5ProtocolVpIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5ProtocolVpIndicationInfo;

REGISTERED AS {q832-3Notification 12};

removeVb5ProtocolVpIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression du conduit virtuel de protocole d'une interface VB5.";

5.6.13 removeVb5VcsIndication (indication de suppression de voies virtuelles VB5)

removeVb5VcsIndication NOTIFICATION

BEHAVIOUR removeVb5VcsIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VcsIndicationInfo;

REGISTERED AS {q832-3Notification 13};

removeVb5VcsIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression de voies virtuelles d'un conduit virtuel qui est associé à une interface VB5.";

5.6.14 removeVb5VpsIndication (indication de suppression de conduits virtuels VB5)

removeVb5VpsIndication NOTIFICATION

BEHAVIOUR removeVb5VpsIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.RemoveVb5VpsIndicationInfo;

REGISTERED AS {q832-3Notification 14};

removeVb5VpsIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue la suppression de conduits virtuels qui sont associés à une interface VB5.";

5.6.15 resourceStatusIndication (indication de statut de ressource)

resourceStatusIndication NOTIFICATION

BEHAVIOUR resourceStatusIndicationBeh;

WITH INFORMATION SYNTAX Q832-3ASN1DefinedTypesModule.ResourceStatusIndicationInfo;

REGISTERED AS {q832-3Notification 15};

resourceStatusIndicationBeh BEHAVIOUR

DEFINED AS

"Cette notification sert à signaler au système d'exploitation homologue le changement de statut d'une ressource.";

6 Définitions de type

Q832-3ASN1DefinedTypesModule {

itu-t (0) recommendation (0) q (17) q832 (832) dot (127) coord (3)

informationModel (0) asn1Modules (2) asn1DefinedTypesModule (0)}

DEFINITIONS IMPLICIT TAGS ::=

BEGIN

-- EXPORTER TOUT

IMPORTS

ObjectInstance

FROM CMIP-1 {joint-iso-itu-t ms (9) cmip (1) modules (0) protocol (3)}

NameType

FROM ASN1DefinedTypesModule {ccitt recommendation m 3100

informationModel(0) asn1Modules(2) asn1DefinedTypesModule(0)}

VciValue, VpiValue

FROM AtmMIBMod {itu-t (0) recommendation (0) i (9) atmm (751)

informationModel (0) asn1Module (2) atm(0)}

VpciValue

FROM AtmMIBMod {itu-t (0) recommendation (0) q (17) 824 (824) dot (127) bsm (6)

informationModel (0) asn1Modules (2) asn1DefinedTypesModule (0)}

; -- fin de imports

-- début des définitions d'identificateur d'objet

q832-3InformationModel

OBJECT IDENTIFIER ::=

{itu-t (0) recommendation (0) q (17) q832 (832) dot (127) coord(3) informationModel(0)}

q832-3StandardSpecificExtension

OBJECT IDENTIFIER ::= {informationModel q832-3StandardSpecificExtension(0)}

q832-3ManagedObjectClass

OBJECT IDENTIFIER ::= {informationModel q832-3ManagedObjectClass(3)}

q832-3Package

OBJECT IDENTIFIER ::= {informationModel q832-3Package(4)}

q832-3NameBinding
 OBJECT IDENTIFIER ::= {informationModel q832-3NameBinding(6)}
q832-3Attribute
 OBJECT IDENTIFIER ::= {informationModel q832-3Attribute (7)}
q832-3Action
 OBJECT IDENTIFIER ::= {informationModel q832-3Action(9)}
q832-3Notification
 OBJECT IDENTIFIER ::= {informationModel q832-3Notification(10)}

-- fin des définitions d'identificateur d'objet

-- autres définitions ASN.1 par ordre alphabétique – toutes sont nouvelles

AddAnLoopRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 logicalUserPortNumber [1] INTEGER OPTIONAL,
 vpciValue [2] VpciValue,
 vciValue [3] VciValue OPTIONAL}

AddAnLoopRequestResult ::= CHOICE {
 loopAdded [0] NULL,
 loopNotAdded [1] LoopNotAddedInfo}

AddLupsIndicationInfo ::= AddLupsRequestInfo

AddLupsRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 logicalUserPortNumber [1] SEQUENCE OF INTEGER}

AddLupsRequestResult ::= INTEGER {
 lupAdded (0),
 lupNotAdded (1)}

AddVb5ConnectionIndicationInfo ::= AddVb5ConnectionRequestInfo

AddVb5ConnectionRequestInfo ::= SEQUENCE {
 egressPeakCellRateCLP0Plus1 [0] INTEGER,
 egressPeakCellRateCLP0 [1] INTEGER,
 ingressPeakCellRateCLP0Plus1 [2] INTEGER,
 ingressPeakCellRateCLP0 [3] INTEGER,
 egressSustainableCellRateCLP0Plus1 [4] INTEGER,
 egressSustainableCellRateCLP0 [5] INTEGER,
 ingressSustainableCellRateCLP0Plus1 [6] INTEGER,
 ingressSustainableCellRateCLP0 [7] INTEGER,
 egressCDVToleranceCLP0Plus1 [8] INTEGER,
 egressCDVToleranceCLP0 [9] INTEGER,
 ingressCDVToleranceCLP0Plus1 [10] INTEGER,
 ingressCDVToleranceCLP0 [11] INTEGER,
 egressMaxBurstSizeCLP0Plus1 [12] INTEGER,
 egressMaxBurstSizeCLP0 [13] INTEGER,
 ingressMaxBurstSizeCLP0Plus1 [14] INTEGER,
 ingressMaxBurstSizeCLP0 [15] INTEGER,
 egressQosClass [16] INTEGER,
 ingressQosClass [17] INTEGER,
 vciValueA [18] INTEGER OPTIONAL,
 vciValueB [19] INTEGER OPTIONAL,
 physicalPortA [20] INTEGER,
 vpiValueA [21] INTEGER,
 logicalServicePortA [22] INTEGER OPTIONAL,
 vpciValueA [23] INTEGER OPTIONAL,
 physicalPortB [24] INTEGER,
 vpiValueB [25] INTEGER,
 logicalServicePortB [26] INTEGER OPTIONAL,
 vpciValueB [27] INTEGER OPTIONAL}

AddVb5ConnectionRequestResult ::= INTEGER {
 vb5ConnectionAdded (0),
 vb5ConnectionNotAdded (1)}

AddVb5InterfaceIndicationInfo ::= AddVb5InterfaceRequestInfo

AddVb5InterfaceRequestInfo ::= INTEGER -- numéro d'accès de service logique

AddVb5InterfaceRequestResult ::= INTEGER {
 vb5InterfaceAdded (0),
 vb5InterfaceNotAdded (1)}

AddVb5ProtocolIndicationInfo ::= AddVb5ProtocolRequestInfo

AddVb5ProtocolRequestInfo ::= SEQUENCE {
 logicalServicePortNumber [0] INTEGER,
 vpciValue [1] VpciValue,
 maxNumVciBitsNearEnd [2] INTEGER OPTIONAL,
 maxNumVciBitsSupported [3] INTEGER OPTIONAL,
 maxNumActiveVccsAllowed [4] INTEGER OPTIONAL,
 maxNumActiveVccsNearEnd [5] INTEGER OPTIONAL}

AddVb5ProtocolRequestResult ::= INTEGER {
 vb5ProtocolAdded (0),
 vb5ProtocolNotAdded (1)}

AddVb5ProtocolVpIndicationInfo ::= AddVb5ProtocolVpRequestInfo

AddVb5ProtocolVpRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 vpciValue [1] VpciValue,
 vpProfile [2] VpProfile OPTIONAL}

AddVb5ProtocolVpRequestResult ::= INTEGER {
 vb5ProtocolVpAdded (0),
 vb5ProtocolVpNotAdded (1)}

AddVb5VcsIndicationInfo ::= AddVb5VcsRequestInfo

AddVb5VcsRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 logicalServiceSubport [1] INTEGER OPTIONAL,
 vpciValue [2] VpciValue,
 vciValue [3] VciValue}

AddVb5VcsRequestResult ::= INTEGER {
 vb5VcsAdded (0),
 vb5VcsNotAdded (1)}

AddVb5VpsIndicationInfo ::= AddVb5VpsRequestInfo

AddVb5VpsRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 logicalServiceSubport [1] INTEGER OPTIONAL,
 addVpInfo [2] AddVpInfo}

AddVb5VpsRequestResult ::= INTEGER {
 vb5VpsAdded (0),
 vb5VpsNotAdded (1)}

AddVpInfo ::= SEQUENCE OF SEQUENCE {
 physicalPort **[0] INTEGER,**
 vpiValue **[1] VpiValue,**
 vpciValue **[2] VpciValue,**
vpProfile **[3] VpProfile OPTIONAL}**

AnServiceLabelInquiryResult ::= INTEGER OPTIONAL

AuditVb5ConnectionRequestInfo ::= SEQUENCE {
 logicalServicePortNumber**[0] INTEGER,**
 specifiedVpOrVc **[1] SpecifiedVpOrVc}**

AuditVb5ConnectionRequestResult ::= CHOICE {
 connectionAudited **[0] SpecifiedVpOrVc,**
 connectionNotAudited **[1] ConnectionNotAuditedInfo}**

AuditVb5VpciRequestInfo ::= SEQUENCE {
 logicalServicePortNumber **[0] INTEGER,**
 specifiedNniVpciOrRemoteVp **[1] SpecifiedNniVpciOrRemoteVp}**

AuditVb5VpciRequestResult ::= CHOICE {
 vpciAudited **[0] SpecifiedNniVpciOrRemoteVp,**
 vpciNotAudited **[1] VpciNotAuditedInfo}**

ConnectionNotAuditedInfo ::= INTEGER {
 unspecified **(0),**
 unknownPhysicalPort **(1),**
 unknownVpiValue **(2),**
 unknownVciValue **(3)}**

ListLupsRequestInfo ::= INTEGER -- *numéro d'accès de service logique*

ListLupsFailureInfo ::= INTEGER {
 unspecified **(0),**
 unknownLspNumber **(1)}**

ListLupsRequestResult ::= CHOICE {
 success **[0] INTEGER,** -- *numéro d'accès de service logique*
 failure **[1] ListLupsFailureInfo}**

ListVb5ProtocolDetailsRequestInfo ::= INTEGER -- *numéro d'accès de service logique*

ListVb5ProtocolDetailsRequestResult ::= CHOICE {
 success **[0] ListVb5ProtocolDetailsSuccessInfo,**
 failure **[1] ListVb5ProtocolDetailsFailureInfo}**

ListVb5ProtocolDetailsFailureInfo ::= INTEGER {
 unspecified **(0),**
 unknownLspNumber **(1)}**

ListVb5ProtocolDetailsSuccessInfo ::= SEQUENCE OF Vb5ProtocolDetails

Vb5ProtocolDetails ::= SEQUENCE {
 protocolType **[0] ProtocolType,**
 vpciValue **[1] VpciValue,**
 vciValue **[2] VciValue,**
 egressPeakCellRateCLP0Plus1 **[3] INTEGER,**
 egressPeakCellRateCLP0 **[4] INTEGER,**
 ingressPeakCellRateCLP0Plus1 **[5] INTEGER,**
 ingressPeakCellRateCLP0 **[6] INTEGER,**
 egressSustainableCellRateCLP0Plus1 **[7] INTEGER,**
 egressSustainableCellRateCLP0 **[8] INTEGER,**
 ingressSustainableCellRateCLP0Plus1 **[9] INTEGER,**

ingressSustainableCellRateCLP0	[10]	INTEGER,
egressCDVToleranceCLP0Plus1	[11]	INTEGER,
egressCDVToleranceCLP0	[12]	INTEGER,
ingressCDVToleranceCLP0Plus1	[13]	INTEGER,
ingressCDVToleranceCLP0	[14]	INTEGER,
egressMaxBurstSizeCLP0Plus1	[15]	INTEGER,
egressMaxBurstSizeCLP0	[16]	INTEGER,
ingressMaxBurstSizeCLP0Plus1	[17]	INTEGER,
ingressMaxBurstSizeCLP0	[18]	INTEGER,
bufferRelease	[19]	BOOLEAN,
maxCc	[20]	INTEGER,
maxInformationFieldLength	[21]	INTEGER,
maxLengthSscopUuField	[22]	INTEGER,
maxPd	[23]	INTEGER,
maxSscopCreditToPeer	[24]	INTEGER,
maxStat	[25]	INTEGER,
sscopTimerCc	[26]	INTEGER,
sscopTimerIdle	[27]	INTEGER,
sscopTimerKeepAlive	[28]	INTEGER,
sscopTimerNoResponse	[29]	INTEGER,
sscopTimerPoll	[30]	INTEGER}

ProtocolType ::= INTEGER {
RTMC (1),
BBCC (2)}

ListVb5InterfacesRequestInfo ::= SEQUENCE OF ObjectInstance

ListVb5InterfacesRequestResult ::= SEQUENCE OF INTEGER -- numéro d'accès de service logique

ListVcsFailureInfo ::= INTEGER {
unspecified (0),
unknownLspNumber (1),
unknownLupNumber (2),
unknownVpciValue (3),
unknownLupVpciCombination (4)}

ListVb5VcsRequestInfo ::= SEQUENCE {
logicalServicePortNumber [0] INTEGER,
logicalUserPortNumber [1] INTEGER OPTIONAL,
vpciValue [2] VpciValue OPTIONAL}

ListVb5VcsRequestResult ::= CHOICE {
success [0] ListVcsSuccessInfo,
failure [1] ListVcsFailureInfo}

ListVcsSuccessInfo ::= SEQUENCE {
logicalUserPortNumber [0] INTEGER OPTIONAL,
vpciValue [1] VpciValue,
vciValue [2] VciValue}

ListVb5VpsRequestInfo ::= SEQUENCE {
logicalServicePortNumber [0] INTEGER,
logicalUserPortNumber [1] INTEGER OPTIONAL}

ListVb5VpsRequestResult ::= CHOICE {
success [0] ListVb5VpsSuccessInfo,
failure [1] ListVb5VpFailureInfo}

ListVb5VpFailureInfo ::= INTEGER {
unspecified (0),
unknownLspNumber (1),

unknownLupNumber (2)}

ListVb5VpsSuccessInfo ::= SEQUENCE OF Vb5VpDetails

Vb5VpDetails ::= SEQUENCE {
logicalUserPortNumber [0] INTEGER OPTIONAL,
physicalPort [1] INTEGER,
vpiValue [2] INTEGER,
vpciValue [3] INTEGER OPTIONAL,
maxNumVciBitsNearEnd [4] INTEGER OPTIONAL,
maxNumVciBitsSupported [5] INTEGER OPTIONAL,
maxNumActiveVccsAllowed [6] INTEGER OPTIONAL,
maxNumActiveVccsNearEnd [7] INTEGER OPTIONAL}

LoopNotAddedInfo ::= INTEGER {
unspecified (0),
unknownLogicalServicePort (1),
unknownLogicalUserPort (2),
unknownVpciValue (3),
unknownVciValue (4),
loopAlreadyPresent (5)}

LoopNotRemovedInfo ::= INTEGER {
unspecified (0),
unknownLogicalServicePort (1),
unknownLogicalUserPort (2),
unknownVpciValue (3),
unknownVciValue (4),
noLoopPresent (5)}

RemoveAnLoopRequestInfo ::= AddAnLoopRequestInfo

RemoveAnLoopRequestResult ::= CHOICE {
loopRemoved [0] NULL,
loopNotRemoved [1] LoopNotRemovedInfo}

RemoveLupsIndicationInfo ::= RemoveLupsRequestInfo

RemoveLupsRequestInfo ::= AddLupsRequestInfo

RemoveLupsRequestResult ::= INTEGER {
lupRemoved (0),
lupNotRemoved (1)}

RemoveVb5ConnectionIndicationInfo ::= RemoveVb5ConnectionRequestInfo

RemoveVb5ConnectionRequestInfo ::= SEQUENCE {
vciValueA [0] INTEGER OPTIONAL,
vciValueB [1] INTEGER OPTIONAL,
physicalPortA [2] INTEGER,
vpiValueA [3] INTEGER,
logicalServicePortA [4] INTEGER OPTIONAL,
vpciValueA [5] INTEGER OPTIONAL,
physicalPortB [6] INTEGER,
vpiValueB [7] INTEGER,
logicalServicePortB [8] INTEGER OPTIONAL,
vpciValueB [9] INTEGER OPTIONAL}

RemoveVb5ConnectionRequestResult ::= INTEGER {
vb5ConnectionRemoved (0),
vb5ConnectionNotRemoved (1)}

RemoveVb5InterfaceIndicationInfo ::= RemoveVb5InterfaceRequestInfo

RemoveVb5InterfaceRequestInfo ::= AddVb5InterfaceRequestInfo

RemoveVb5InterfaceRequestResult ::= INTEGER {
 vb5InterfaceRemoved (0),
 vb5InterfaceNotRemoved (1)}

RemoveVb5ProtocolIndicationInfo ::= RemoveVb5ProtocolRequestInfo

RemoveVb5ProtocolRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 vb5ProtocolType [1] Vb5ProtocolType}

RemoveVb5ProtocolRequestResult ::= INTEGER {
 vb5ProtocolRemoved (0),
 vb5ProtocolNotRemoved (1)}

RemoveVb5ProtocolVpIndicationInfo ::= RemoveVb5ProtocolVpRequestInfo

RemoveVb5ProtocolVpRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 vpciValue [1] VpciValue}

RemoveVb5ProtocolVpRequestResult ::= INTEGER {
 vb5ProtocolVpRemoved (0),
 vb5ProtocolVpNotRemoved (1)}

RemoveVb5VcsIndicationInfo ::= RemoveVb5VcsRequestInfo

RemoveVb5VcsRequestInfo ::= AddVb5VcsRequestInfo

RemoveVb5VcsRequestResult ::= INTEGER {
 vb5VcsRemoved (0),
 vb5VcsNotRemoved (1)}

RemoveVb5VpsIndicationInfo ::= RemoveVb5VpsRequestInfo

RemoveVb5VpsRequestInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 logicalServiceSubport [1] INTEGER OPTIONAL,
 removeVpInfo [2] RemoveVpInfo}

RemoveVb5VpsRequestResult ::= INTEGER {
 vb5VpsRemoved (0),
 vb5VpsNotRemoved (1)}

RemoveVpInfo ::= SEQUENCE OF SEQUENCE {
 physicalPort [0] INTEGER,
 vpiValue [1] VpiValue,
 vpciValue [2] VpciValue}

ResourceStatusIndicationInfo ::= SEQUENCE {
 logicalServicePortNumber[0] INTEGER,
 logicalUserPortNumber [1] INTEGER OPTIONAL,
 vpciValue [2] VpciValue,
 resourceStatus [3] ResourceStatus}

ResourceStatus ::= INTEGER {
 fullyOperational (0),
 administratelyBlockedTestCallsAllowed (1),
 administratelyBlockedNoCellFlow (2),

fault (3)}

SnAccessLabelsInquiryResult ::= SEQUENCE {
 snLabel [0] INTEGER OPTIONAL,
 interfaceLabel [1] INTEGER OPTIONAL}

SpecifiedNniVpci ::= SEQUENCE {
 logicalUserPortNumber [0] INTEGER OPTIONAL,
 vpciValue [1] VpciValue}

SpecifiedNniVpciOrRemoteVp ::= CHOICE {
 specifiedNniVpci [0] SpecifiedNniVpci,
 specifiedRemoteVp [1] SpecifiedVp}

SpecifiedVc ::= SEQUENCE {
 physicalPort [0] INTEGER,
 vpiValue [1] VpiValue,
 vciValue [2] VciValue}

SpecifiedVp ::= SEQUENCE {
 physicalPort [0] INTEGER,
 vpiValue [1] VpiValue}

SpecifiedVpOrVc ::= CHOICE {
 specifiedVp [0] SpecifiedVp,
 specifiedVc [1] SpecifiedVc}

Vb5ProtocolType ::= INTEGER {
 rtmc (0),
 bbcc (1)}

VpciNotAuditedInfo ::= INTEGER {
 unspecified (0),
 unknownLupNumber (1),
 unknownVpci (2),
 unknownPhysicalPort (3),
 unknownVpiValue (4)}

VpProfile ::= SEQUENCE {
 maxNumVciBitsNearEnd [0] INTEGER,
 maxNumVciBitsSupported [1] INTEGER,
 maxNumActiveVccsAllowed [2] INTEGER,
 maxNumActiveVccsNearEnd [3] INTEGER}

END – de Q832-3ASN1DefinedTypesModule

7 Piles de protocoles

On peut utiliser, dans la pile de protocoles de la présente Recommandation, les piles de protocoles spécifiées dans l'UIT-T Q.811, Q.812, G.773 et dans la partie de l'UIT-T G.784 consacrée aux répartiteurs numériques SDH. On aura recours aux Recommandations suivantes pour compléter ces piles en vue d'inclure le mode ATM:

- UIT-T Q.2811, Interfaces Q3 et X large bande – Protocoles des couches inférieures.
- UIT-T Q.2812, Interfaces Q3 et X large bande – Protocoles des couches supérieures.

Conditions de gestion

Les conditions générales sont indiquées ci-dessous ainsi que dans les principes de configuration pour interfaces VB5.1 et VB5.2.

A.1 Conditions générales

Les conditions générales comprennent les fonctions de coordination de gestion générale entre le réseau d'accès et le nœud de service de part et d'autre des interfaces Q3/X.

A.1.1 Configuration coordonnée des conduits virtuels et des voies virtuelles

La fonction de gestion de configuration doit assurer l'adjonction et le retrait coordonnés de conduits et voies virtuelles aux interfaces UNI comme VB5, de façon que les conduits et voies virtuelles puissent être ajoutés et supprimés sans interruption.

A.1.2 Contrôle des connexions VPC

Un mécanisme est requis pour vérifier l'identité des connexions VPC établies entre un port utilisateur et un nœud de service de façon que l'on puisse repérer les erreurs de brassage à l'intérieur d'un réseau d'accès.

A.1.3 Coordination des données de configuration des ports

La coordination des informations de configuration relatives aux ports utilisateur et aux ports service ainsi qu'à leurs conduits virtuels et voies virtuelles est requise afin de garantir la compatibilité entre le réseau d'accès et le nœud de service.

A.1.4 Coordination des valeurs d'identificateur VPCI

Il est nécessaire d'assurer la coordination de gestion entre le nœud de service et le réseau d'accès pour ce qui est de l'attribution des valeurs d'identificateur VPCI aux connexions.

A.1.5 Cohérence de la configuration

Il est nécessaire de vérifier la cohérence de la configuration des ports utilisateur logique, des ports service logique et des accès d'interface UNI.

A.1.6 Disponibilité des informations

Les informations relatives à une interface VB5 ne devraient pas être visibles par des opérateurs autres que ceux qui sont mis en relation par cette interface VB5.

A.2 Coordination de l'interface VB5

A.2.1 Création

Il est nécessaire de coordonner la création de l'interface VB5, avec les informations suivantes:

- 1) identificateurs de VP;
- 2) connexions VCC attribuées à des protocoles B-BCC et RTMC;
- 3) série d'identificateurs VCI;
- 4) nombre maximal de connexions VCC simultanément actives;

- 5) largeur de bande maximale de la connexion VPC, indiquée par le descripteur de trafic, dans les deux sens (sortie et entrée) qui spécifie par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 6) classe de qualité de service du VP;
- 7) autres voies virtuelles configurées dans le VP;
- 8) le profil de trafic du VC, décrivant par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 9) classe de qualité de service du VC.

A.2.2 Vérification et audit

Il est nécessaire de vérifier et de contrôler la configuration correcte de l'interface VB5 (au sujet des informations correspondantes, voir création).

A.2.3 Modification

Il est nécessaire de coordonner la modification de l'interface VB5.

A.2.4 Suppression

Il est nécessaire de coordonner la suppression de l'interface VB5.

A.2.5 Configuration de VC et VP

Il est nécessaire de coordonner la configuration des VC et des VP pour les utilisateurs, avec les informations suivantes:

- 1) identificateurs VP/VC;
- 2) étendue VCI pour VP;
- 3) nombre maximal de connexions VCC simultanément actives dans un VP;
- 4) largeur de bande maximale de la connexion VPC, indiquée par le descripteur de trafic, dans les deux sens (sortie et entrée) qui spécifie par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 5) classe de qualité de service du VP;
- 6) VC configurés dans le VP;
- 7) le profil de trafic du VC, décrivant par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales.

A.2.6 Vérification et audit

Il est nécessaire de vérifier et de contrôler la configuration correcte des VC et des VP pour les utilisateurs (au sujet des informations correspondantes, voir configuration).

A.2.7 Modification des VP et des VC

Il est nécessaire de coordonner la modification des VC et des VP pour les utilisateurs.

A.2.8 Vérification et audit

Il est nécessaire de vérifier et de contrôler la configuration correcte de l'interface VB5 (au sujet des informations correspondantes, voir création).

A.2.9 Suppression des VP et des VC

Il est nécessaire de coordonner la suppression des VC et des VP pour les utilisateurs.

A.3 Coordination de l'interface UNI

A.3.1 Création

Il est nécessaire de coordonner la création de VP d'utilisateur et de VC d'utilisateur entre les utilisateurs et l'AN, avec les informations suivantes:

- 1) identificateurs VP/VC;
- 2) étendue VCI;
- 3) nombre maximal de connexions VCC simultanément actives;
- 4) largeur de bande maximale de la connexion VPC, indiquée par le descripteur de trafic, dans les deux sens (sortie et entrée) qui spécifie par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 5) classe de qualité de service du VP;
- 6) le profil de trafic du VC, décrivant par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 7) classe de qualité de service du VC.

A.3.2 Vérification et audit

Il est nécessaire de vérifier et de contrôler la configuration correcte de VP d'utilisateur et de VC d'utilisateur entre les utilisateurs et l'AN (au sujet des informations correspondantes, voir création).

A.3.3 Modification

Il est nécessaire de coordonner la modification de VP d'utilisateur et de VC d'utilisateur entre les utilisateurs et l'AN.

A.3.4 Suppression

Il est nécessaire de coordonner la suppression de VP d'utilisateur et de VC d'utilisateur entre les utilisateurs et l'AN.

A.4 Conditions de coordination de commande de connexion support à large bande

Les conditions sont fondées ici sur la nécessité de prendre en charge le protocole B-BCC à l'interface VB5.2 [3].

A l'interface VB5 et à l'interface UNI, les connexions VPC/VCC peuvent être sélectionnées aussi bien par le SN que par l'AN qui doivent donc, l'un et l'autre, connaître les informations correspondantes afin d'appliquer les fonctions de commande CAC et de choisir les identificateurs corrects de connexion VPC/VC.

A.4.1 Voies VC à l'interface VB5

Afin d'assurer la création, la suppression et la modification d'une voie virtuelle à l'interface VB5.2 sous la commande des procédures B-BCC, les informations suivantes doivent être disponibles dans l'AN et le SN:

- les connexions VPC associées à l'interface VB5.2; les informations associées à chaque connexion VPC doivent comprendre:
 - 1) étendue VCI;
 - 2) nombre maximal de connexions VCC simultanément actives;
 - 3) largeur de bande maximale de la connexion VPC, indiquée par le descripteur de trafic, dans les deux sens (sortie et entrée) qui spécifie par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
 - 4) classe de qualité de service du VP;
 - 5) VC attribués dans le VP;
 - 6) le profil de trafic du VC, décrivant par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
 - 7) classe de qualité de service du VC.

Ces informations sont disponibles dans le réseau d'accès et dans le nœud de service, indiquées par les classes d'objets gérés modélisant les points de connexion VP/VC, définis dans l'UIT-T I.751.

A.4.2 Voies virtuelles à l'interface UNI

Afin de prendre en charge la création, la suppression et la modification des VC à l'interface UNI sous le contrôle des procédures B-BCC, les informations suivantes doivent être disponibles dans le réseau d'accès et dans le nœud de service:

- les connexions VPC associées au port LUP; les informations associées à chaque connexion VPC doivent comprendre:
 - 1) étendue VCI;
 - 2) nombre maximal de connexions VCC simultanément actives;

- 3) largeur de bande maximale de la connexion VPC, indiquée par le descripteur de trafic dans les deux sens (sortie et entrée), qui spécifie par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 4) classe de qualité de service du VP;
- 5) VC attribués dans le VP;
- 6) le profil de trafic du VC, décrivant par exemple:
 - débit cellulaire crête;
 - tolérance de variation du temps de propagation des cellules;
 - débit cellulaire soutenu;
 - taille maximale des rafales;
- 7) classe de qualité de service du VC.

Ces informations sont disponibles dans le réseau d'accès et sont indiquées par les classes d'objets gérés modélisant les points de connexion VP/VC définis dans l'UIT-T I.751. Les mêmes informations doivent être fournies au nœud de service.

A.5 Gestion des dérangements et de la performance

Les conditions suivantes ont été relevées pour la gestion des dérangements et de la performance.

A.5.1 Signalisation des dérangements

Les dérangements doivent toujours être signalés par l'interface X lorsque les interfaces VB5 de trafic ne sont pas opérationnelles.

A.5.2 Localisation des dérangements

L'interface X doit toujours assurer les activités de coordination permettant à un nœud de service et à un réseau d'accès de coopérer pour localiser l'origine de dérangements, par exemple en cas d'utilisation de bouclages.

ANNEXE B

Conditions de transaction

Ces transactions sont destinées à être incorporées dans des protocoles d'implémentation. Elles ont été définies au moyen de la spécification d'appel de procédure à distance [7] car cela constitue un moyen simple de spécifier formellement des transactions, compatible avec le protocole CMIP mais sans y être limité. La présente annexe ne spécifie pas d'implémentation RPC obligatoire mais son utilisation en tant que telle n'est pas exclue.

```

/*****
*****
* Définitions de transaction
*****
*****/

```

```

program XVB5 {
version {
/* */
addAnLoopRequestAnswer AddAnLoopRequest (addAnLoopRequestCall) = 1
addLupsIndicationAnswer AddLupsIndication (addLupsIndicationCall) = 2
addLupsRequestAnswer AddLupsRequest (addLupsRequestCall) = 3
addVb5ProtocolVpIndicationAnswer AddVb5ProtocolVpIndication (addVb5ProtocolVpIndicationCall) = 4
addVb5ProtocolVpRequestAnswer AddVb5ProtocolVpRequest (addVb5ProtocolVpRequestCall) = 5
addVb5InterfaceRequestAnswer AddVb5InterfaceRequest (addVb5InterfaceRequestCall) = 6
addVb5InterfaceIndicationAnswer AddVb5InterfaceIndication (addVb5InterfaceIndicationCall) = 7
addVb5ProtocolIndicationAnswer AddVb5ProtocolIndication (addVb5ProtocolIndicationCall) = 8
addVb5ProtocolRequestAnswer AddVb5ProtocolRequest (addVb5ProtocolRequestCall) = 9
addVb5VcsIndicationAnswer AddVb5VcsIndication (addVb5VcsIndicationCall) = 10
addVb5VcsRequestAnswer AddVb5VcsRequest (addVb5VcsRequestCall) = 11
addVb5VpsIndicationAnswer AddVb5VpsIndication (addVb5VpsIndicationCall) = 12
addVb5VpsRequestAnswer AddVb5VpsRequest (addVb5VpsRequestCall) = 13
addVb5ConnectionIndicationAnswer AddVb5ConnectionIndication (addVb5ConnectionIndicationCall) = 14
addVb5ConnectionRequestAnswer AddVb5ConnectionRequest (addVb5ConnectionRequestCall) = 15
auditVb5ConnectionRequestAnswer AuditVb5ConnectionRequest (auditVb5ConnectionRequestCall) = 16
auditVb5VpciRequestAnswer AuditVb5VpciRequest (auditVb5VpciRequestCall) = 17
listLupsRequestAnswer ListLupsRequest (listLupsRequestCall) = 18
listVb5ProtocolDetailsRequestAnswer ListVb5ProtocolDetailsRequest (listVb5ProtocolDetailsRequestCall) = 19
listVb5InterfacesRequestAnswer ListVb5InterfacesRequest (listVb5InterfacesRequestCall) = 20
listVb5VcsRequestAnswer ListVb5VcsRequest (listVb5VcsRequestCall) = 21
listVb5VpsRequestAnswer ListVb5VpsRequest (listVb5VpsRequestCall) = 22
removeAnLoopRequestAnswer RemoveAnLoopRequest (removeAnLoopRequestCall) = 23
removeLupsIndicationAnswer RemoveLupsIndication (removeLupsIndicationCall) = 24
removeLupsRequestAnswer RemoveLupsRequest (removeLupsRequestCall) = 25
removeVb5ProtocolVpIndicationAnswer RemoveVb5ProtocolVpIndication
(removeVb5ProtocolVpIndicationCall) = 26
removeVb5ProtocolVpRequestAnswer RemoveVb5ProtocolVpRequest (removeVb5ProtocolVpRequestCall) = 27
removeVb5InterfaceIndicationAnswer RemoveVb5InterfaceIndication (removeVb5InterfaceIndicationCall) = 28
removeVb5InterfaceRequestAnswer RemoveVb5InterfaceRequest (removeVb5InterfaceRequestCall) = 29
removeVb5ProtocolIndicationAnswer RemoveVb5ProtocolIndication (removeVb5ProtocolIndicationCall) = 30
removeVb5ProtocolRequestAnswer RemoveVb5ProtocolRequest (removeVb5ProtocolRequestCall) = 31
removeVb5VcsIndicationAnswer RemoveVb5VcsIndication (removeVb5VcsIndicationCall) = 32
removeVb5VcsRequestAnswer RemoveVb5VcsRequest (removeVb5VcsRequestCall) = 33
removeVb5VpsIndicationAnswer RemoveVb5VpsIndication (removeVb5VpsIndicationCall) = 34
removeVb5VpsRequestAnswer RemoveVb5VpsRequest (removeVb5VpsRequestCall) = 35
removeVb5ConnectionIndicationAnswer RemoveVb5ConnectionIndication
(removeVb5ConnectionIndicationCall) = 36
removeVb5ConnectionRequestAnswer RemoveVb5ConnectionRequest (removeVb5ConnectionRequestCall) = 37
anServiceLabelInquiryAnswer AnServiceLabelInquiry (anServiceLabelInquiryCall) = 38
snAccessLabelsInquiryAnswer SnAccessLabelsInquiry (snAccessLabelsInquiryCall) = 39
resourceStatusIndicationAnswer ResourceStatusIndication (resourceStatusIndicationCall) = 40
/* */
} = 1 /* dénote version 1 */
} = 8323 /* dénote Q.832.3 */

```

```

/*****
*****
* Définitions des paramètres
*****
*****/

```

/******

*** Définitions des paramètres pour la transaction 1**

Cette transaction est utilisée par le système d'exploitation du SN afin de demander au système d'exploitation du réseau d'accès de boucler une connexion de façon que les cellules envoyées au réseau d'accès soient renvoyées.

*/

```
struct addAnLoopRequestCall {
    unsigned int LogicalServicePort;
    union switch (bool VpcReachesUserPort) {
        case TRUE:
            unsigned int LogicalUserPort;
        case FALSE:
            void;
    } OptionalUserPort;
    unsigned int VpciValue;
    union switch (bool VccLoopback) {
        case TRUE:
            unsigned int VciValue;
        case FALSE:
            void;
    } OptionalVciValue;
};

struct addAnLoopRequestAnswer {
    union switch (bool Failure) {
        case TRUE:
            enum {
                Unspecified                = 0,
                UnknownLogicalServicePort  = 1,
                UnknownLogicalUserPort     = 2,
                UnknownVpciValue           = 3,
                UnknownVciValue            = 4,
                LoopAlreadyPresent         = 5
            } FailureReason;
        case FALSE:
            void;
    } OptionalFailureReason;
};
```

/******

*** Définitions des paramètres pour la transaction 2**

Cette transaction est utilisée pour signaler au système d'exploitation homologue l'adjonction d'un port utilisateur logique à une interface VB5.

*/

```
struct addLupsIndicationCall {
    unsigned int LogicalServicePortNumber;
    unsigned int LogicalUserPortNumber<>;
};

struct addLupsIndicationAnswer {
    bool Acknowledge;
};
```

/******

*** Définitions des paramètres pour la transaction 3**

Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter un port utilisateur logique à une interface VB5.

*/

```
struct addLupsRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int LogicalUserPortNumber<>;
};
```

```

};
struct addLupsRequestAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 4
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue l'adjonction d'un protocole VP à une interface VB5.
*/
struct addVb5ProtocolVpIndicationCall {
    unsigned int LogicalServicePortNumber;
    unsigned int VpciValue;
    union switch (bool ProfileSupported) {
        case TRUE:
            struct {
                unsigned int maxNumVciBitsNearEnd;
                unsigned int maxNumVciBitsSupported;
                unsigned int maxNumActiveVccsAllowed;
                unsigned int maxNumActiveVccsNearEnd;
            } Profile;
        case FALSE:
            void;
    } OptionalProfile
};
struct addVb5ProtocolVpIndicationAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 5
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter un protocole VP à une interface VB5.
*/
struct addVb5ProtocolVpRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int VpciValue;
    union switch (bool ProfileSupported) {
        case TRUE:
            struct {
                unsigned int maxNumVciBitsNearEnd;
                unsigned int maxNumVciBitsSupported;
                unsigned int maxNumActiveVccsAllowed;
                unsigned int maxNumActiveVccsNearEnd;
            } Profile;
        case FALSE:
            void;
    } OptionalProfile
};
struct addVb5ProtocolVpRequestAnswer {
    bool Acknowledge;
};

```

```

/*****
Définitions des paramètres pour la transaction 6
*****/
Cette transaction est utilisée pour informer le système d'exploitation homologue qu'une nouvelle interface VB5 a
été ajoutée.
*/
struct addVb5InterfaceIndicationCall {
    unsigned int LogicalServicePortNumber;
};
struct addVb5InterfaceIndicationAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 7
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter une interface VB5.
*/
struct addVb5InterfaceRequestCall {
    unsigned int LogicalServicePortNumber;
};
struct addVb5InterfaceRequestAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 8
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue l'adjonction d'un protocole à une
interface VB5 existante.
*/
struct addVb5ProtocolIndicationCall {
    unsigned int LogicalServicePortNumber;
    unsigned int VpciValue;
    unsigned int VciValue;
    enum {
        RTMC      = 1,
        BBCC      = 2
    } ProtocolType;
    struct {
        /* Le débit cellulaire est exprimé en cellules par seconde. */
        /* La variation du délai cellulaire est exprimé en microsecondes. */
        /* La taille des rafales est exprimée en nombre de cellules. */
        unsigned int EgressPeakCellRateCLP0Plus1;
        unsigned int EgressPeakCellRateCLP0;
        unsigned int IngressPeakCellRateCLP0Plus1;
        unsigned int IngressPeakCellRateCLP0;
        unsigned int EgressSustainableCellRateCLP0Plus1;
        unsigned int EgressSustainableCellRateCLP0;
        unsigned int IngressSustainableCellRateCLP0Plus1;
        unsigned int IngressSustainableCellRateCLP0;
        unsigned int EgressCDVToleranceCLP0Plus1;
        unsigned int EgressCDVToleranceCLP0;
        unsigned int IngressCDVToleranceCLP0Plus1;
        unsigned int IngressCDVToleranceCLP0;
        unsigned int EgressMaxBurstSizeCLP0Plus1;
        unsigned int EgressMaxBurstSizeCLP0;
        unsigned int IngressMaxBurstSizeCLP0Plus1;
        unsigned int IngressMaxBurstSizeCLP0;
    } QosParameters;
    struct {
        bool BufferRelease;      /* à la libération de connexion */
        unsigned int MaxCc;     /* valeur maximale dans les unités PDU de la variable d'état VT(CC) */
    };
};

```



```

    unsigned int MaxInformationFieldLength;    /* en octets */
    unsigned int MaxLengthSscopUuField;      /* en octets */
    unsigned int MaxPd;                       /* valeur maximale dans les unités PDU de la variable d'état VT(PD) */
    unsigned int MaxSscopCreditToPeer;       /* valeur absolue dans les unités PDU de la fenêtre de
                                             réception */
    unsigned int MaxStat;                     /* nombre maximal d'éléments énumérés dans une unité PDU STAT */
    unsigned int SscopTimerCc;                /* intervalle de temps entre transmissions en millisecondes */
    unsigned int SscopTimerIdle;             /* phase de repos en millisecondes */
    unsigned int SscopTimerKeepAlive;        /* phase de survie en millisecondes */
    unsigned int SscopTimerNoResponse;       /* temps de non-réponse en millisecondes */
    unsigned int SscopTimerPoll;            /* phase d'interrogation active en millisecondes */
} SaalParameters;
};
struct addVb5ProtocolIndicationAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 9
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter un protocole à une
interface VB5 existante.
*/
struct addVb5ProtocolRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int VpciValue;
    unsigned int VciValue;
    enum {
        RTMC      = 1,
        BBCC      = 2
    } ProtocolType;
    struct {
        /* Le débit cellulaire est exprimé en cellules par seconde. */
        /* La variation du délai cellulaire est exprimé en microsecondes. */
        /* La taille des rafales est exprimée en nombre de cellules. */
        unsigned int EgressPeakCellRateCLP0Plus1;
        unsigned int EgressPeakCellRateCLP0;
        unsigned int IngressPeakCellRateCLP0Plus1;
        unsigned int IngressPeakCellRateCLP0;
        unsigned int EgressSustainableCellRateCLP0Plus1;
        unsigned int EgressSustainableCellRateCLP0;
        unsigned int IngressSustainableCellRateCLP0Plus1;
        unsigned int IngressSustainableCellRateCLP0;
        unsigned int EgressCDVToleranceCLP0Plus1;
        unsigned int EgressCDVToleranceCLP0;
        unsigned int IngressCDVToleranceCLP0Plus1;
        unsigned int IngressCDVToleranceCLP0;
        unsigned int EgressMaxBurstSizeCLP0Plus1;
        unsigned int EgressMaxBurstSizeCLP0;
        unsigned int IngressMaxBurstSizeCLP0Plus1;
        unsigned int IngressMaxBurstSizeCLP0;
    } QosParameters;
    struct {
        bool BufferRelease;    /* à la libération de connexion */
        unsigned int MaxCc;    /* valeur maximale dans les unités PDU de la variable d'état
                               VT(CC) VT(CC) */
        unsigned int MaxInformationFieldLength;    /* en octets */
        unsigned int MaxLengthSscopUuField;      /* en octets */
        unsigned int MaxPd;                       /* valeur maximale dans les unités PDU de la variable d'état VT(PD) */
        unsigned int MaxSscopCreditToPeer;       /* valeur absolue dans les unités PDU de la fenêtre de
                                             réception */
        unsigned int MaxStat;                     /* nombre maximal d'éléments énumérés dans une unité PDU STAT */
        unsigned int SscopTimerCc;                /* intervalle de temps entre transmissions en millisecondes */

```

```

        unsigned int SscopTimerIdle;    /* phase de repos en millisecondes */
        unsigned int SscopTimerKeepAlive;    /* phase de survie en millisecondes */
        unsigned int SscopTimerNoResponse;    /* durée de non-réponse en millisecondes */
        unsigned int SscopTimerPoll;    /* phase d'interrogation active en millisecondes */
    } SaalParameters;
};
struct addVb5ProtocolRequestAnswer {
    bool Acknowledge;

/*****
* Définitions des paramètres pour la transaction 10
*****/
    Cette transaction est utilisée pour signaler au système d'exploitation homologue l'adjonction de voies virtuelles à
    un conduit virtuel qui est associé à une interface VB5.
    */
    struct addVb5VcsIndicationCall {
        unsigned int LogicalServicePortNumber;
        unsigned int *LogicalServiceSubport;
            /* présent si VPC atteint un port utilisateur */
        unsigned int VpciValue;
        unsigned int VciValue<>;
    };
    struct addVb5VcsIndicationAnswer {
        bool Acknowledge;
    };

/*****
* Définitions des paramètres pour la transaction 11
*****/
    Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter des voies virtuelles
    à un conduit virtuel qui est associé à une interface VB5.
    */
    struct addVb5VcsRequestCall {
        unsigned int LogicalServicePortNumber;
        unsigned int *LogicalServiceSubport;
            /* présent si VPC atteint un port utilisateur */
        unsigned int VpciValue;
        unsigned int VciValue<>;
    };
    struct addVb5VcsRequestAnswer {
        bool Acknowledge;
    };

/*****
* Définitions des paramètres pour la transaction 12
*****/
    Cette transaction est utilisée pour signaler au système d'exploitation homologue l'adjonction de conduits virtuels
    qui sont associés à une interface VB5.
    */
    struct addVb5VpsIndicationCall {
        unsigned int LogicalServicePortNumber;
        unsigned int *LogicalServiceSubport;
            /* présent si VPC atteint un port utilisateur */
        struct {
            unsigned int PhysicalPort;
            unsigned int VpiValue;
            unsigned int VpciValue;
            union switch (bool ProfileSupported) {
                case TRUE:
                    struct {
                        unsigned int maxNumVciBitsNearEnd;
                        unsigned int maxNumVciBitsSupported;
                        unsigned int maxNumActiveVccsAllowed;
                    };
            };
        };
    };

```

```

        unsigned int maxNumActiveVccsNearEnd;
    } Profile;
    case FALSE:
        void;
    } OptionalProfile
} VpInfo<>;
};
struct addVb5VpsIndicationAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 13
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter des conduits virtuels qui sont associés à une interface VB5.
*/
struct addVb5VpsRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int *LogicalServiceSubport;
    /* présent si VPC atteint un port utilisateur */
    struct {
        unsigned int PhysicalPort;
        unsigned int VpiValue;
        unsigned int VpciValue;
        union switch (bool ProfileSupported) {
            case TRUE:
                struct {
                    unsigned int maxNumVciBitsNearEnd;
                    unsigned int maxNumVciBitsSupported;
                    unsigned int maxNumActiveVccsAllowed;
                    unsigned int maxNumActiveVccsNearEnd;
                } Profile;
            case FALSE:
                void;
        } OptionalProfile
    } VpInfo<>;
};
struct addVb5VpsRequestAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 14
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue l'adjonction d'un brassage de conduit virtuel ou de voie virtuelle associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.
*/
struct addVb5ConnectionIndicationCall {
    /* Le débit cellulaire est exprimé en cellules par seconde. */
    /* La variation du délai cellulaire est exprimée en microsecondes. */
    /* La taille des rafales est exprimée en nombre de cellules. */
    unsigned int EgressPeakCellRateCLP0Plus1;
    unsigned int EgressPeakCellRateCLP0;
    unsigned int IngressPeakCellRateCLP0Plus1;
    unsigned int IngressPeakCellRateCLP0;
    unsigned int EgressSustainableCellRateCLP0Plus1;
    unsigned int EgressSustainableCellRateCLP0;
    unsigned int IngressSustainableCellRateCLP0Plus1;
    unsigned int IngressSustainableCellRateCLP0;
    unsigned int EgressCDVToleranceCLP0Plus1;
    unsigned int EgressCDVToleranceCLP0;
};

```

```

unsigned int IngressCDVToleranceCLP0Plus1;
unsigned int IngressCDVToleranceCLP0;
unsigned int EgressMaxBurstSizeCLP0Plus1;
unsigned int EgressMaxBurstSizeCLP0;
unsigned int IngressMaxBurstSizeCLP0Plus1;
unsigned int IngressMaxBurstSizeCLP0;
enum {
    Class0 = 0, /* au mieux sans paramètres spécifiés */
    Class1 = 1, /* paramètres comme spécifié pour CBR */
    Class2 = 2, /* paramètres comme spécifié pour VBR */
    Class3 = 3, /* paramètres comme spécifié pour données en mode connexion */
    Class4 = 4 /* paramètres comme spécifié pour données en mode non connecté */
 } EgressQosClass;
enum {
    Class0 = 0, /* au mieux sans paramètres spécifiés */
    Class1 = 1, /* paramètres comme spécifié pour CBR */
    Class2 = 2, /* paramètres comme spécifié pour VBR */
    Class3 = 3, /* paramètres comme spécifié pour données en mode connexion */
    Class4 = 4 /* paramètres comme spécifié pour données en mode non connecté */
 } IngressQosClass;
union switch (bool VcCrossConnection) {
    case TRUE:
        struct {
            unsigned int VciValueA; /* VCI du côté A */
            unsigned int VciValueB; /* VCI du côté B */
         } VciPoints;
    case FALSE:
        void;
     } ConnectionTypeDetails;
unsigned int PhysicalPortA;
unsigned int VpiValueA;
union switch (bool DetailsForPortA) {
    case TRUE:
        struct {
            unsigned int *LogicalServicePort;
            unsigned int *VpciValue;
            /* ne peut pas être spécifié pour les deux accès */
         } Details;
    case FALSE:
        void;
     } OptionalDetailsForPortA;
unsigned int PhysicalPortB;
unsigned int VpiValueB;
union switch (bool DetailsForPortB) {
    case TRUE:
        struct {
            unsigned int *LogicalUserPort;
            unsigned int *VpciValue;
            /* ne peut pas être spécifié pour les deux accès */
         } Details;
    case FALSE:
        void;
     } OptionalDetailsForPortB;
 } AddVb5ConnectionIndicationCall;
};
struct addVb5ConnectionIndicationAnswer {
    bool Acknowledge;
};

```

/******

*** Définitions des paramètres pour la transaction 15**

Cette transaction est utilisée afin de demander au système d'exploitation homologue d'ajouter un brassage de conduit virtuel ou de voie virtuelle associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.

*/

```
struct addVb5ConnectionRequestCall {
    /* Le débit cellulaire est exprimé en cellules par seconde. */
    /* La variation du délai cellulaire est exprimé en microsecondes. */
    /* La taille des rafales est exprimée en nombre de cellules. */
    unsigned int EgressPeakCellRateCLP0Plus1;
    unsigned int EgressPeakCellRateCLP0;
    unsigned int IngressPeakCellRateCLP0Plus1;
    unsigned int IngressPeakCellRateCLP0;
    unsigned int EgressSustainableCellRateCLP0Plus1;
    unsigned int EgressSustainableCellRateCLP0;
    unsigned int IngressSustainableCellRateCLP0Plus1;
    unsigned int IngressSustainableCellRateCLP0;
    unsigned int EgressCDVToleranceCLP0Plus1;
    unsigned int EgressCDVToleranceCLP0;
    unsigned int IngressCDVToleranceCLP0Plus1;
    unsigned int IngressCDVToleranceCLP0;
    unsigned int EgressMaxBurstSizeCLP0Plus1;
    unsigned int EgressMaxBurstSizeCLP0;
    unsigned int IngressMaxBurstSizeCLP0Plus1;
    unsigned int IngressMaxBurstSizeCLP0;
    enum {
        Class0    = 0, /* au mieux sans paramètres spécifiés */
        Class1    = 1, /* paramètres comme spécifié pour CBR */
        Class2    = 2, /* paramètres comme spécifié pour VBR */
        Class3    = 3, /* paramètres comme spécifié pour données en mode connexion */
        Class4    = 4 /* paramètres comme spécifié pour données en mode non connecté */
    } EgressQosClass;
    enum {
        Class0    = 0, /* au mieux sans paramètres spécifiés */
        Class1    = 1, /* paramètres comme spécifié pour CBR */
        Class2    = 2, /* paramètres comme spécifié pour VBR */
        Class3    = 3, /* paramètres comme spécifié pour données en mode connexion */
        Class4    = 4 /* paramètres comme spécifié pour données en mode non connecté */
    } IngressQosClass;
    union switch (bool VcCrossConnection) {
        case TRUE:
            struct {
                unsigned int VciValueA; /* VCI du côté A */
                unsigned int VciValueB; /* VCI du côté B */
            } VciPoints;
        case FALSE:
            void;
    } ConnectionTypeDetails;
    unsigned int PhysicalPortA;
    unsigned int VpiValueA;
    union switch (bool DetailsForPortA) {
        case TRUE:
            struct {
                unsigned int *LogicalServicePort;
                unsigned int *VpciValue;
                /* ne peut pas être spécifié pour les deux accès */
            } Details;
        case FALSE:
            void;
    } OptionalDetailsForPortA;
    unsigned int PhysicalPortB;
};
```

```

unsigned int VpiValueB;
union switch (bool DetailsForPortB) {
    case TRUE:
        struct {
            unsigned int *LogicalUserPort;
            unsigned int *VpciValue;
            /* ne peut pas être spécifié pour les deux accès */
        } Details;
    case FALSE:
        void;
    } OptionalDetailsForPortB;
} AddVb5ConnectionRequestCall;
};
struct addVb5ConnectionRequestAnswer {
    bool Acknowledge;
};

/*
*****
* Définitions des paramètres pour la transaction 16
*****
Cette transaction est utilisée afin de demander au système d'exploitation homologue de contrôler une connexion
qui est associée à une interface VB5.
*/
struct auditVb5ConnectionRequestCall {
    unsigned int LogicalServicePortNumber;
    union switch (bool VpCrossConnection) {
        case TRUE:
            struct {
                unsigned int PhysicalPort;
                unsigned int VpiValue;
            } KnownVpEnd;
        case FALSE:
            struct {
                unsigned int PhysicalPort;
                unsigned int VpiValue;
                unsigned int VciValue;
            } KnownVcEnd;
        } KnownEnd;
};
struct auditVb5ConnectionRequestAnswer {
    union switch (bool Success) {
        case FALSE:
            enum {
                Unspecified = 0,
                UnknownPhysicalPort = 1,
                UnknownVpiValue = 2,
                UnknownVciValue = 3
            } FailureReason;
        case TRUE:
            union switch (bool VpCrossConnection) {
                case TRUE:
                    struct {
                        unsigned int PhysicalPort;
                        unsigned int VpiValue;
                    } OtherVpEnd;
                case FALSE:
                    struct {
                        unsigned int PhysicalPort;
                        unsigned int VpiValue;
                        unsigned int VciValue;
                    } OtherVcEnd;
                } OtherEnd;
            } AuditVb5ConnectionRequestAnswer;

```

```
};
```

```
/******  
* Définitions des paramètres pour la transaction 17  
*****
```

```
Cette transaction est utilisée afin de demander au système d'exploitation homologue de contrôler un identificateur VPCI qui est associé à une interface VB5.  
*/
```

```
struct auditVb5VpciRequestCall {  
    unsigned int LogicalServicePortNumber;  
    union switch (bool KnownVpci) {  
        case TRUE:  
            struct {  
                unsigned int *LogicalUserPortNumber;  
                    /* présent si VPC atteint un port utilisateur */  
                unsigned int VpciValue;  
            } KnownNniVpci;  
        case FALSE:  
            struct {  
                unsigned int PhysicalPort;  
                unsigned int VpiValue;  
            } KnownRemoteVp;  
    } KnownEnd;  
};  
struct auditVb5VpciRequestAnswer {  
    union switch (bool Success) {  
        case FALSE:  
            enum {  
                Unspecified           = 0,  
                UnknownLupNumber      = 1,  
                UnknownVpci           = 2,  
                UnknownPhysicalPort   = 3,  
                UnknownVpiValue       = 4  
            } FailureReason;  
        case TRUE:  
            union switch (bool ReturnedVpci) {  
                case TRUE:  
                    struct {  
                        unsigned int *LogicalUserPortNumber;  
                            /* présent si VPC atteint un port utilisateur */  
                        unsigned int VpciValue  
                    } ReturnedNniVpci;  
                case FALSE:  
                    struct {  
                        unsigned int PhysicalPort;  
                        unsigned int VpiValue;  
                    } ReturnedRemoteVp;  
            } ReturnedEnd;  
    } AuditVb5VpciRequestAnswer;  
};
```

```
/******  
* Définitions des paramètres pour la transaction 18  
*****
```

```
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'énumérer les ports utilisateur logique associés à une interface VB5 entre un réseau d'accès et un nœud de service que les deux systèmes d'exploitation commandent ensemble.  
*/
```

```
struct listLupsRequestCall {  
    unsigned int LogicalServicePortNumber;  
};  
struct listLupsRequestAnswer {  
    union switch (bool Success) {
```

```

    case FALSE:
        enum {
            Unspecified          = 0,
            UnknownLspNumber    = 1
        } FailureReason;
    case TRUE:
        unsigned int LogicalUserPortNumber<>;
} ListLupsRequestAnswer;
};

```

```

/*****

```

*** Définitions des paramètres pour la transaction 19**

```

*****

```

Cette transaction est utilisée afin de demander au système d'exploitation homologue d'énumérer les détails des protocoles d'une interface VB5 entre un réseau d'accès et un nœud de service que les deux systèmes d'exploitation commandent ensemble.

```

*/

```

```

struct listVb5ProtocolDetailsRequestCall {
    unsigned int LogicalServicePortNumber;
};

```

```

struct listVb5ProtocolDetailsRequestAnswer {

```

```

    union switch (bool Success) {

```

```

        case FALSE:

```

```

            enum {

```

```

                Unspecified          = 0,

```

```

                UnknownLspNumber    = 1

```

```

            } FailureReason;

```

```

        case TRUE:

```

```

            struct {

```

```

                enum {

```

```

                    RTMC            = 1,

```

```

                    BBCC            = 2

```

```

                } ProtocolType;

```

```

                unsigned int VpciValue;

```

```

                unsigned int VciValue;

```

```

                struct {

```

```

                    /* Le débit cellulaire est exprimé en cellules par seconde. */

```

```

                    /* La variation du délai cellulaire est exprimé en microsecondes. */

```

```

                    /* La taille des rafales est exprimée en nombre de cellules. */

```

```

                    unsigned int EgressPeakCellRateCLP0Plus1;

```

```

                    unsigned int EgressPeakCellRateCLP0;

```

```

                    unsigned int IngressPeakCellRateCLP0Plus1;

```

```

                    unsigned int IngressPeakCellRateCLP0;

```

```

                    unsigned int EgressSustainableCellRateCLP0Plus1;

```

```

                    unsigned int EgressSustainableCellRateCLP0;

```

```

                    unsigned int IngressSustainableCellRateCLP0Plus1;

```

```

                    unsigned int IngressSustainableCellRateCLP0;

```

```

                    unsigned int EgressCDVToleranceCLP0Plus1;

```

```

                    unsigned int EgressCDVToleranceCLP0;

```

```

                    unsigned int IngressCDVToleranceCLP0Plus1;

```

```

                    unsigned int IngressCDVToleranceCLP0;

```

```

                    unsigned int EgressMaxBurstSizeCLP0Plus1;

```

```

                    unsigned int EgressMaxBurstSizeCLP0;

```

```

                    unsigned int IngressMaxBurstSizeCLP0Plus1;

```

```

                    unsigned int IngressMaxBurstSizeCLP0;

```

```

                } QosParameters;

```

```

                struct {

```

```

                    bool BufferRelease; /* à la libération de connexion */

```

```

                    unsigned int MaxCc;

```

```

                    /* valeur maximale dans les unités PDU de la variable d'état VT(CC) */

```

```

                    unsigned int MaxInformationFieldLength; /* en octets */

```

```

                    unsigned int MaxLengthScopUuField; /* en octets */

```

```

                    unsigned int MaxPd;

```



```

        /* valeur maximale dans les unités PDU de la variable d'état VT(PD) */
        unsigned int MaxSscopCreditToPeer;
        /* valeur absolue dans les unités PDU de la fenêtre de réception */
        unsigned int MaxStat;
        /* nombre maximal d'éléments énumérés dans une unité PDU STAT */
        unsigned int SscopTimerCc;
        /* intervalle de temps entre transmissions en millisecondes */
        unsigned int SscopTimerIdle; /* phase de repos en millisecondes */
        unsigned int SscopTimerKeepAlive;
        /* phase de survie en millisecondes */
        unsigned int SscopTimerNoResponse;
        /* temps de non-réponse en millisecondes */
        unsigned int SscopTimerPoll;
        /* phase d'interrogation active en millisecondes */
    } SaalParameters;
} ProtocolDetails<>;
} ListProtocolDetailsRequestAnswer;
};

/*****
* Définitions des paramètres pour la transaction 20
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'énumérer les identités des
interfaces VB5 entre réseau(x) d'accès et le(s) nœud(s) de service que les deux systèmes d'exploitation
commandent ensemble.
*/
struct listVb5InterfacesRequestCall {
    void;
};
struct listVb5InterfacesRequestAnswer {
    unsigned int LogicalServicePortNumber<>;
};

/*****
* Définitions des paramètres pour la transaction 21
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue d'énumérer les voies
virtuelles associées à une interface VB5.
*/
struct listVb5VcsRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int *LogicalUserPortNumber;
        /* présent en cas de contrainte de réponse à un certain LUP */
    unsigned int *VpciValue;
        /* présent en cas de contrainte de réponse à un certain VPCI */
};
struct listVb5VcsRequestAnswer {
    union switch (bool Success) {
        case FALSE:
            enum {
                Unspecified                = 0,
                UnknownLspNumber           = 1,
                UnknownLupNumber           = 2,
                UnknownVpciValue           = 3,
                UnknownLupVpciCombination   = 4
            } FailureReason;
        case TRUE:
            struct {
                unsigned int *LogicalUserPort;
                    /* présent si VCC atteint LUP */
                unsigned int VpciValue;
                unsigned int VciValue<>;
            } VpciVcs<>;
    };
};

```

```

    } ListVb5VcsRequestAnswer;
};

```

```

/*****

```

```

* Définitions des paramètres pour la transaction 22

```

```

*****

```

Cette transaction est utilisée afin de demander au système d'exploitation homologue d'énumérer les conduits virtuels associés à une interface VB5.

```

*/

```

```

struct listVb5VpsRequestCall {
    unsigned int LogicalServicePortNumber;
    union switch (bool AllVps)
        case TRUE:
            void;
        case FALSE:
            unsigned int *LogicalUserPortNumber;
                /* absent pour la sélection de connexions VPC aboutissant au réseau d'accès */
    } Filter;
    unsigned int *LogicalUserPortNumber;
};

```

```

struct listVb5VpsRequestAnswer {
    union switch (bool Success) {
        case FALSE:
            enum {
                Unspecified                = 0,
                UnknownLspNumber           = 1,
                UnknownLupNumber           = 2
            } FailureReason;
        case TRUE:
            struct
                unsigned int *LogicalUserPort;
                    /* present si VPC atteint LUP */
                struct {
                    unsigned int PhysicalPort;
                    unsigned int VpiValue;
                    unsigned int VpciValue;
                    union switch (bool ProfileSupported) {
                        case TRUE:
                            struct {
                                unsigned int maxNumVciBitsNearEnd;
                                unsigned int maxNumVciBitsSupported;
                                unsigned int maxNumActiveVccsAllowed;
                                unsigned int maxNumActiveVccsNearEnd;
                            } Profile;
                        case FALSE:
                            void;
                    } OptionalProfile
                } VpInfo;
            } SupportVps<>;
    } ListVb5VpsRequestAnswer;
};

```

```

/*****

```

```

* Définitions des paramètres pour la transaction 23

```

```

*****

```

Cette transaction est utilisée par le système d'exploitation du SN afin de demander au système d'exploitation du réseau d'accès de supprimer un circuit d'une connexion.

```

*/

```

```

struct removeAnLoopRequestCall {
    unsigned int LogicalServicePort;
    union switch (bool VpcReachesUserPort) {
        case TRUE:
            unsigned int LogicalUserPort;
    }
};

```

```

        case FALSE:
            void;
    } OptionalUserPort;
    unsigned int VpciValue;
    union switch (bool VccLoopback) {
        case TRUE:
            unsigned int VciValue;
        case FALSE:
            void;
    } OptionalVciValue;
};
struct removeAnLoopRequestAnswer {
    union switch (bool Failure) {
        case TRUE:
            enum {
                Unspecified                = 0,
                UnknownLogicalServicePort  = 1,
                UnknownLogicalUserPort     = 2,
                UnknownVpciValue           = 3,
                UnknownVciValue            = 4,
                NoLoopPresent               = 5
            } FailureReason;
        case FALSE:
            void;
    } OptionalFailureReason;
};

```

```

/*****

```

*** Définitions des paramètres pour la transaction 24**

```

*****/

```

Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression d'un port utilisateur logique d'une interface VB5.

```

*/

```

```

struct removeLupsIndicationCall {
    unsigned int LogicalServicePortNumber;
    unsigned int LogicalUserPortNumber<>;
};
struct removeLupsIndicationAnswer {
    bool Acknowledge;
};

```

```

/*****

```

*** Définitions des paramètres pour la transaction 25**

```

*****/

```

Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer un port utilisateur logique d'une interface VB5.

```

*/

```

```

struct removeLupsRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int LogicalUserPortNumber<>;
};
struct removeLupsRequestAnswer {
    bool Acknowledge;
};

```

```

/*****

```

*** Définitions des paramètres pour la transaction 26**

```

*****/

```

Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression du conduit virtuel de protocole d'une interface VB5.

```

*/

```

```

struct removeVb5ProtocolVpIndicationCall {
    unsigned int LogicalServicePortNumber;
};

```

```

    unsigned int VpciValue;
};
struct removeVb5ProtocolVpIndicationAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 27
*****/

```

Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer le conduit virtuel de protocole d'une interface VB5.

```

*/
struct removeVb5ProtocolVpRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int VpciValue;
};
struct removeVb5ProtocolVpRequestAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 28
*****/

```

Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression d'une interface VB5.

```

*/
struct removeVb5InterfaceIndicationCall {
    unsigned int LogicalServicePortNumber;
};
struct removeVb5InterfaceIndicationAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 29
*****/

```

Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer une interface VB5.

```

*/
struct removeVb5InterfaceRequestCall {
    unsigned int LogicalServicePortNumber;
};
struct removeVb5InterfaceRequestAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 30
*****/

```

Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression de voies virtuelles d'un conduit virtuel qui est associé à une interface VB5.

```

*/
struct removeVb5VcsIndicationCall {
    unsigned int LogicalServicePortNumber;
    unsigned int *LogicalServiceSubport;
    /* présent si VPC atteint un port utilisateur */
    unsigned int VpciValue;
    unsigned int VciValue<>;
};
struct removeVb5VcsIndicationAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 31
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer des voies
virtuelles d'un conduit virtuel qui est associé à une interface VB5.
*/
struct removeVb5VcsRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int *LogicalServiceSubport;
        /* présent si VPC atteint un port utilisateur */
    unsigned int VpciValue;
    unsigned int VciValue<>;
};
struct removeVb5VcsRequestAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 32
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression d'un protocole
d'une interface VB5.
*/
struct removeVb5ProtocolIndicationCall {
    unsigned int LogicalServicePortNumber;
    enum {
        RTMC      = 1,
        BBCC      = 2
    } ProtocolType;
};
struct removeVb5ProtocolIndicationAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 33
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer un protocole
d'une interface VB5.
*/
struct removeVb5ProtocolRequestCall {
    unsigned int LogicalServicePortNumber;
    enum {
        RTMC      = 1,
        BBCC      = 2
    } ProtocolType;
};
struct removeVb5ProtocolRequestAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 34
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression de conduits
virtuels qui sont associés à une interface VB5.
*/
struct removeVb5VpsIndicationCall {
    unsigned int LogicalServicePortNumber;
    unsigned int *LogicalServiceSubport;
        /* présent si VPC atteint un port utilisateur */
    struct {
        unsigned int PhysicalPort;
        unsigned int VpiValue;
    };
};

```

```

        unsigned int VpciValue;
    } VpInfo<>;
};
struct removeVb5VpsIndicationAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 35
*****/
Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer des conduits
virtuels qui sont associés à une interface VB5.
*/
struct removeVb5VpsRequestCall {
    unsigned int LogicalServicePortNumber;
    unsigned int *LogicalServiceSubport;
    /* présent si VPC atteint un port utilisateur */
    struct {
        unsigned int PhysicalPort;
        unsigned int VpiValue;
        unsigned int VpciValue;
    } VpInfo<>;
};
struct removeVb5VpsRequestAnswer {
    bool Acknowledge;
};

/*****
* Définitions des paramètres pour la transaction 36
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue la suppression d'un brassage de
conduit virtuel ou de voie virtuelle associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud
de service. Le sens d'entrée va du nœud de service au réseau d'accès.
*/
struct removeVb5ConnectionIndicationCall {
    union switch (bool VcCrossConnection) {
        case TRUE:
            struct {
                unsigned int VciValueA; /* VCI du côté A */
                unsigned int VciValueB; /* VCI du côté B */
            } VciPoints;
        case FALSE:
            void;
    } ConnectionTypeDetails;
    unsigned int PhysicalPortA;
    unsigned int VpiValueA;
    union switch (bool DetailsForPortA) {
        case TRUE:
            struct {
                unsigned int *LogicalServicePort;
                unsigned int *VpciValue;
                /* ne peut pas être spécifié pour les deux accès */
            } Details;
        case FALSE:
            void;
    } OptionalDetailsForPortA;
    unsigned int PhysicalPortB;
    unsigned int VpiValueB;
    union switch (bool DetailsForPortB) {
        case TRUE:
            struct {
                unsigned int *LogicalUserPort;
                unsigned int *VpciValue;
            }

```

```

                /* ne peut pas être spécifié pour les deux accès */
            } Details;
        case FALSE:
            void;
        } OptionalDetailsForPortB;
    } RemoveVb5ConnectionIndicationCall;
};
struct removeVb5ConnectionIndicationAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 37
*****/

```

Cette transaction est utilisée afin de demander au système d'exploitation homologue de supprimer un brassage de conduit virtuel ou de voie virtuelle associée à une interface VB5. Le sens de sortie va du réseau d'accès au nœud de service. Le sens d'entrée va du nœud de service au réseau d'accès.

```

*/
struct removeVb5ConnectionRequestCall {
    union switch (bool VcCrossConnection) {
        case TRUE:
            struct {
                unsigned int VciValueA; /* VCI du côté A */
                unsigned int VciValueB; /* VCI du côté B */
            } VciPoints;
        case FALSE:
            void;
        } ConnectionTypeDetails;
    unsigned int PhysicalPortA;
    unsigned int VpiValueA;
    union switch (bool DetailsForPortA) {
        case TRUE:
            struct {
                unsigned int *LogicalServicePort;
                unsigned int *VpciValue;
                /* ne peut pas être spécifié pour les deux accès */
            } Details;
        case FALSE:
            void;
        } OptionalDetailsForPortA;
    unsigned int PhysicalPortB;
    unsigned int VpiValueB;
    union switch (bool DetailsForPortB) {
        case TRUE:
            struct {
                unsigned int *LogicalUserPort;
                unsigned int *VpciValue;
                /* ne peut pas être spécifié pour les deux accès */
            } Details;
        case FALSE:
            void;
        } OptionalDetailsForPortB;
    } RemoveVb5ConnectionRequestCall;
};
struct removeVb5ConnectionRequestAnswer {
    bool Acknowledge;
};

```

```

/*****
* Définitions des paramètres pour la transaction 38
*****/
Cette transaction est utilisée par le système d'exploitation d'un réseau d'accès afin de demander l'étiquette qu'un
SN utilise pour le réseau d'accès.
*/
struct anServiceLabelInquiryCall {
    void;
};
struct anServiceLabelInquiryAnswer {
    union switch (bool AnLabelKnown) {
        case TRUE:
            unsigned int AnLabel;
        case FALSE:
            void;
    } AnServiceLabelInquiryAnswer;
};

/*****
* Définitions des paramètres pour la transaction 39
*****/
Cette transaction est utilisée par le système d'exploitation of an SN afin de demander l'étiquette qu'utilise un
réseau d'accès pour le SN et la référence locale utilisée par le réseau d'accès pour l'interface VB5.
*/
struct snAccessLabelsInquiryCall {
    void;
};
struct snAccessLabelsInquiryAnswer {
    union switch (bool SnAndIntLabelsKnown) {
        case TRUE:
            unsigned int SnLabel;
            unsigned int InterfaceLabel;
        case FALSE:
            void;
    } snAccessLabelsInquiryAnswer;
};

/*****
* Définitions des paramètres pour la transaction 40
*****/
Cette transaction est utilisée pour signaler au système d'exploitation homologue le changement de statut d'une
ressource.
*/
struct resourceStatusIndicationCall {
    unsigned int LogicalServicePort;
    union switch (bool SpecifiedLogicalUserPort) {
        case TRUE:
            unsigned int LogicalServicePort;
        case FALSE:
            void;
    } OptionalLogicalServicePort;
    unsigned int VpciValue<>;
    enum {
        FullyOperational = 0,
        AdministratelyBlockedTestCallsAllowed = 1,
        AdministratelyBlockedNoCellFlow = 2,
        Fault = 3
    } Status;
};
struct resourceStatusIndicationAnswer {
    bool Acknowledge;
};

```


APPENDICE I

Bibliographie

- [1] UIT-T G.773 (1993), *Suites de protocoles aux interfaces Q pour la gestion de systèmes de transmission.*
- [2] UIT-T G.774 (2001), *Modèle d'information de gestion de la hiérarchie numérique synchrone du point de vue des éléments de réseau.*
- [3] UIT-T G.803 (2000), *Architecture des réseaux de transport à hiérarchie numérique synchrone.*
- [4] UIT-T I.211 (1993), *Aspects service du RNIS à large bande.*
- [5] UIT-T I.311 (1996), *Aspects généraux réseau du RNIS à large bande.*
- [6] UIT-T I.327 (1993), *Architecture fonctionnelle du RNIS à large bande.*
- [7] UIT-T I.356 (2000), *Caractéristiques du transfert de cellules de la couche ATM du RNIS-LB.*
- [8] UIT-T I.371 (2000), *Gestion du trafic et des encombrements dans le RNIS-LB.*
- [9] UIT-T I.413 (1993), *Interface usager-réseau du RNIS à large bande.*
- [10] UIT-T de la série I.432.x, *Interface utilisateur-réseau du RNIS-LB – Spécification de la couche physique.*
- [11] UIT-T I.580 (1995), *Dispositions générales d'interfonctionnement entre le RNIS à large bande et le RNIS à 64 kbit/s.*
- [12] UIT-T I.610 (1999), *Principes et fonctions d'exploitation et de maintenance du RNIS à large bande.*
- [13] UIT-T I.732 (2000), *Caractéristiques fonctionnelles des équipements ATM.*
- [14] UIT-T M.3200 (1997), *Services de gestion du réseau de gestion des télécommunications et domaines gérés des télécommunications: aperçu général.*
- [15] UIT-T M.3207.1 (1996), *Service de gestion RGT: aspects maintenance de la gestion du RNIS-LB.*
- [16] UIT-T M.3400 (2000), *Fonctions de gestion du réseau de gestion des télécommunications.*
- [17] UIT-T M.3610 (1996), *Principes d'application du concept de réseau de gestion des télécommunications à la gestion du RNIS-LB.*
- [18] UIT-T Q.821 (2000), *Description d'étape 2 et d'étape 3 pour l'interface Q3 – Supervision des alarmes.*
- [19] UIT-T Q.822 (1994), *Description d'étape 1, d'étape 2 et d'étape 3 de l'interface Q3 – Gestion de la qualité de fonctionnement.*
- [20] UIT-T X.208 (1988), *Spécification de la syntaxe abstraite numéro un (ASN.1).*
- [21] UIT-T X.701 (1997) | ISO/CEI 10040:1998, *Technologies de l'information – Interconnexion des systèmes ouverts – Aperçu général de la gestion-systèmes.*
- [22] UIT-T X.722 (1992) | ISO/CEI 10165-4:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: directives pour la définition des objets gérés.*

- [23] UIT-T X.733 (1992) | ISO/CEI 10164-4:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de signalisation des alarmes.*
- [24] UIT-T X.734 (1992) | ISO/CEI 10164-5:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de gestion des rapports d'événement.*
- [25] UIT-T X.735 (1992) | ISO/CEI 10164-6:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de commande des registres de consignation.*
- [26] UIT-T X.737 (1995) | ISO/CEI 10164-14:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: catégories de tests de confiance et de diagnostic.*
- [27] UIT-T X.738 (1993) | ISO/CEI 10164-13:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de récapitulation.*
- [28] UIT-T X.739 (1993) | ISO/CEI 10164-11:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: objets et attributs métriques.*
- [29] UIT-T X.745 (1993) | ISO/CEI 10164-12:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: fonction de gestion des tests.*
- [30] UIT-T X.746 (1995) | ISO/CEI 10164-15:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de programmation.*
- [31] ATM Forum Specification, *M4 Interface Requirements and Logical MIB, ATM Network Element View, Version 1.0.*
- [32] ATM Forum Specification, *CMIP Specification for the M4 Interface, Version 1.0.*
- [33] ATM Forum Specification, *ATM User-Network Interface Specification, Version 3.0.*
- [34] ATM Forum Specification, *ATM User-Network Interface Specification, Version 3.1.*
- [35] ATM Forum Specification, *ATM User-Network Interface Specification, Version 4.0.*

APPENDICE II

Résumé des transactions

Catégorie	Nom	Type	Origine
Création VB5	anServiceLabelInquiry	Action	AN
	snAccessLabelsInquiry	Action	SN
	addVb5InterfaceRequest	Action	Les 2
	addVb5InterfaceIndication	Notification	Les 2
Suppression VB5	removeVb5InterfaceRequest	Action	Les 2
	removeVb5InterfaceIndication	Notification	Les 2
Modification – VP de protocole VB5	addVb5ProtocolVpRequest	Action	Les 2
	addVb5ProtocolIndication	Notification	Les 2
	removeVb5ProtocolVpRequest	Action	Les 2
	removeVb5ProtocolVpIndication	Notification	Les 2
Modification – Protocoles VB5	addVb5ProtocolVpRequest	Action	Les 2
	addVb5ProtocolVpIndication	Notification	Les 2
	removeVb5ProtocolRequest	Action	Les 2
	removeVb5ProtocolIndication	Notification	Les 2
Modification – Port utilisateur logique	addLupsRequest	Action	Les 2
	addLupsIndication	Notification	Les 2
	removeLupsRequest	Action	Les 2
	removeLupsIndication	Notification	Les 2
Modification – Conduits virtuels	addVb5VpsRequest	Action	Les 2
	addVb5VpsIndication	Notification	Les 2
	removeVb5VpsRequest	Action	Les 2
	removeVb5VpsIndication	Notification	Les 2
Modification – Voies virtuelles	addVb5VcsRequest	Action	Les 2
	addVb5VcsIndication	Notification	Les 2
	removeVb5VcsRequest	Action	Les 2
	removeVb5VcsIndication	Notification	Les 2
Modification – Connexions	addVb5ConnectionRequest	Action	Les 2
	addVb5ConnectionIndication	Notification	Les 2
	removeVb5ConnectionRequest	Action	Les 2
	removeVb5ConnectionIndication	Notification	Les 2
Audit – Demandes de listes	listVb5InterfacesRequest	Action	Les 2
	listVb5ProtocolDetailsRequest	Action	Les 2
	listLupsRequest	Action	Les 2
	listVb5VpsRequest	Action	Les 2
	listVb5VcsRequest	Action	Les 2
Audit – Demandes d'audit	auditVb5ConnectionRequest	Action	Les 2
	auditVb5VpciRequest	Action	Les 2
Signalisation des dérangements	resourceStatusIndication	Notification	Les 2
Localisation des dérangements	addAnLoopRequest	Action	SN
	removeAnLoopRequest	Action	SN

SÉRIES DES RECOMMANDATIONS UIT-T

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