|  |
| --- |
| **Recommandation UIT-R BT.1869**  **(03/2010)** |
| **Schéma de multiplexage pour paquets de longueur variable dans des systèmes de diffusion multimédias numériques** |
| **Série BT**  **Service de radiodiffusion télévisuelle** |

Avant-propos

Le rôle du Secteur des radiocommunications est d’assurer l’utilisation rationnelle, équitable, efficace et économique du spectre radioélectrique par tous les services de radiocommunication, y compris les services par satellite, et de procéder à des études pour toutes les gammes de fréquences, à partir desquelles les Recommandations seront élaborées et adoptées.

Les fonctions réglementaires et politiques du Secteur des radiocommunications sont remplies par les Conférences mondiales et régionales des radiocommunications et par les Assemblées des radiocommunications assistées par les Commissions d’études.

# Politique en matière de droits de propriété intellectuelle (IPR)

La politique de l'UIT‑R en matière de droits de propriété intellectuelle est décrite dans la «Politique commune de l'UIT‑T, l'UIT‑R, l'ISO et la CEI en matière de brevets», dont il est question dans l'Annexe 1 de la Résolution UIT-R 1. Les formulaires que les titulaires de brevets doivent utiliser pour soumettre les déclarations de brevet et d'octroi de licence sont accessibles à l'adresse <http://www.itu.int/ITU-R/go/patents/fr>, où l'on trouvera également les Lignes directrices pour la mise en oeuvre de la politique commune en matière de brevets de l'UIT‑T, l'UIT‑R, l'ISO et la CEI et la base de données en matière de brevets de l'UIT-R.

|  |  |
| --- | --- |
| Séries des Recommandations UIT-R  (Egalement disponible en ligne: <http://www.itu.int/publ/R-REC/fr>) | |
| **Séries** | Titre |
| **BO** | Diffusion par satellite |
| BR | Enregistrement pour la production, l'archivage et la diffusion; films pour la télévision |
| **BS** | Service de radiodiffusion sonore |
| **BT** | Service de radiodiffusion télévisuelle |
| **F** | Service fixe |
| **M** | Services mobile, de radiorepérage et d'amateur y compris les services par satellite associés |
| **P** | Propagation des ondes radioélectriques |
| **RA** | Radio astronomie |
| **RS** | Systèmes de télédétection |
| **S** | Service fixe par satellite |
| **SA** | Applications spatiales et météorologie |
| **SF** | Partage des fréquences et coordination entre les systèmes du service fixe par satellite et du service fixe |
| **SM** | Gestion du spectre |
| **SNG** | Reportage d'actualités par satellite |
| **TF** | Emissions de fréquences étalon et de signaux horaires |
| **V** | Vocabulaire et sujets associés |

|  |
| --- |
| ***Note****: Cette Recommandation UIT-R a été approuvée en anglais aux termes de la procédure détaillée dans la  Résolution UIT-R 1.* |

*Publication électronique*

Genève, 2011

© UIT 2011

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

RECOMMANDATION UIT-R BT.1869

Schéma de multiplexage pour paquets de longueur variable dans   
des systèmes de diffusion multimédias numériques[[1]](#footnote-1)\*

(Question UIT-R 45/6)

(2010)

Domaine d'application

La présente Recommandation porte sur les schémas de multiplexage pour paquets de longueur variable sur des canaux de diffusion. Elle spécifie les schémas de transport de paquets IP sur des canaux de télévision à savoir le format d'encapsulation, le format des paquets IP compressés avec en-tête et les signaux de contrôle de la transmission.

L'Assemblée des radiocommunications de l'UIT,

considérant

a) que divers types de signaux pour services multimédias peuvent être délivrés en diffusion numérique;

b) que des services multimédias ont été aussi mis en oeuvre dans des réseaux de télécommunication dans lesquels des paquets IP, y compris des paquets IPv4 et IPv6, sont utilisés;

c) que ces paquets IP sont par nature une longueur variable qui peut atteindre 65 535 octets au maximum;

d) qu'il est souhaitable de disposer d'un mécanisme de transport compatible IP pour les services de diffusion multimédia afin de permettre une harmonisation entre les services de diffusion multimédia et les services de télécommunication;

e) qu'un flux de transport MPEG-2 a été adopté pour la diffusion numérique comme moyen de transport des divers types de signaux;

f) que le flux de transport MPEG‑2 est composé de paquets de longueur fixe de 188 octets incluant une charge utile de 184 octets;

g) qu'il est souhaitable, pour la diffusion multimédia, de disposer d'un schéma de multiplexage qui permettrait un transport plus efficace et une réception moins complexe de paquets de longueur variable,

recommande

**1** d'utiliser le schéma de multiplexage décrit dans l'Annexe 1 pour les systèmes de diffusion multimédia;

**2** que la conformité avec la présente Recommandation est volontaire. Toutefois, la Recommandation peut contenir certaines dispositions de obligatoires (pour assurer par exemple l'interopérabilité ou l'applicabilité) et que la conformité avec la présente Recommandation est atteinte lorsque ces dispositions obligatoires sont toutes respectées. Les mots «doi(ven)t» ou toute expression de caractère obligatoire et leurs équivalents sont utilisés pour exprimer des prescriptions. L'utilisation de termes comme doit/doivent ne doit en aucune façon être interprétée comme impliquant une conformité totale ou partielle avec la présente Recommandation.

Annexe 1  
  
Schéma de multiplexage de paquets de longueur variable

Références

Références normatives

[1] IETF RFC 791: Internet Protocol.

Cette norme IETF est disponible à l'adresse. <http://www.ietf.org/rfc/rfc791.txt>.

[2] IETF RFC 2460: Internet Protocol, Version 6 (IPv6) Specification.

Cette norme IETF est disponible à l'adresse. <http://www.ietf.org/rfc/rfc2460.txt>.

[3] IETF RFC 768: User Datagram Protocol.

Cette norme IETF est disponible à l'adresse: <http://www.ietf.org/rfc/rfc768.txt>.

[4] ETSI TS 102 606 v1.1.1 (2007-10): Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE) Protocol.

[5] ETSI EN 301 192 v1.4.2 (2008-04): Digital Video Broadcasting (DVB); DVB specification for data broadcasting.

Référence informative

[6] Recommandation UIT T H.222.0, 2006: Technologies de l'information – Codage générique d'images animées et information audio associée: Systèmes – *Generic coding of moving pictures and associated audio information: Systems*.

Abréviations

ACM codage et modulation adaptatifs (*adaptive coding and modulation*)

AMT table de localisation d'adresse (*address map table*)

ATM mode de transfert asynchrone (*asynchronous transfer mode*)

CID identification de contexte (*context identification*)

CRC contrôle de redondance cyclique (*cyclic redundancy check*)

DVB radiodiffusion vidéo numérique (*digital video broadcast*)

ETSI European Telecommunications Standards Institute

GSE encapsulation de flux générique (*generic stream encapsulation*)

IETF Internet Engineering Task Force

IGMP Internet Group Management Protocol

INT table de notification IP/MAC (*IP/MAC notification table*)

IP protocole Internet (*Internet protocol*)

MAC contrôle d'accès aux médias (*media access control*)

MLD identification de l'auditeur de multidiffusion (*multicast listener discovery*)

MPE encapsulation multiprotocole (*multi protocol encapsulation*)

MPEG Moving Pictures Experts Group

NIT table d'information du réseau (*network information table*)

ONU unité de réseau optique (*optical network unit*)

PES flux élémentaire en paquets (*packetized elementary stream*)

RFC appel aux commentaires (norme IETF) (*request for comment (IETF standard)*)

SN numéro de séquence (*sequence number*)

TLV valeur de longueur de type (*type length value*)

TS flux de transport (*transport stream*)

UDP protocole datagramme d'utilisateur (*user datagram protocol*)

VCM codage et modulation variables (*variable coding and modulation*)

# 1 Introduction

Divers services de diffusion multimédias devraient pouvoir être rendus en adoptant les schémas de multiplexage pour paquets TS MPEG‑2 de longueur fixe et le schéma de multiplexage de paquets de longueur variable décrit à la Fig. 1.

Figure 1

Pile protocolaire



# 2 Prescriptions relatives au schéma de multiplexage de paquets de longueur variable

Comme les services de radiodiffusion utilisent le spectre des fréquences radioélectriques, qui est une ressource finie, et que des services analogues ont été mis en place sur l'Internet, un schéma de multiplexage pour paquets de longueur variable devrait prendre en charge les prescriptions suivantes:

a) Pouvoir multiplexer des paquets de longueur variable de formats divers y compris des paquets IPv4 et IPv6.

b) Pouvoir multiplexer sans fragmentation des paquets de longueur maximale de 65 535 octets.

c) Les préfixes nécessaires pour la transmission des paquets devraient être de faible longueur.

d) Le processus de réception devrait être suffisamment simple pour traiter les paquets reçus à haut débit de paquets.

# 3 Schéma d'encapsulation pour paquets de longueur variable

## 3.1 Format des conteneurs type-longueur-valeur

Le schéma de multiplexage type-longueur-valeur (TLV) est représenté à la Fig. 2 et dans le Tableau 1. Ce schéma permet de multiplexer des paquets de longueur variable de tout format sauf si un filtrage et une fragmentation des paquets sont nécessaires. Le type de paquet est indiqué par le champ *packet\_type* et sa longueur par le champ *length*. Les paquets IP à en-tête compressée et les signaux de contrôle de transmission peuvent aussi être encapsulés dans des conteneurs TLV. Ce schéma permet de multiplexer des paquets de 65 535 octets au maximum sans fragmentation. Le préfixe de transmission est petit et le schéma de multiplexage des TLV utilise efficacement la capacité de transmission.

FIGURE 2

Format du conteneur TLV



TABLEAU 1

TLV container

|  |  |  |
| --- | --- | --- |
| Syntaxe | Nbre de bits | Mnémonique |
| TLV { |  |  |
| '01' | 2 | bslbf |
| reserved\_future\_use | 6 | bslbf |
| packet\_type | 8 | bslbf |
| length | 16 | uimsbf |
| if (packet\_type==0x01) |  |  |
| IPv4\_packet ( ) |  |  |
| else if (packet\_type==0x02) |  |  |
| IPv6\_packet ( ) |  |  |
| else if (packet\_type==0x02) |  |  |
| compressed\_ip\_packet( ) |  |  |
| else if (packet\_type==0xFE) |  |  |
| signalling\_packet ( ) |  |  |
| else if (packet\_type==0xFF) |  |  |
| for(i=0;i<N;i++){ |  |  |
| NULL | 8 | bslbf |
| } |  |  |
| } |  |  |
| } |  |  |

**reserved\_future\_use** –Indique que la valeur peut être utilisée pour des extensions futures. Sauf indication contraire dans le présent document, tous les bits réservés sont mis à «1».

**packet\_type** – Indique le type de paquet encapsulé. Il est codé conformément au Tableau 2.

TABLEAU 2

Valeurs attribuées aux types de paquet

|  |  |
| --- | --- |
| Valeur | Description |
| 0x00 | Réservé |
| 0x01 | Paquet IPv4 |
| 0x02 | Paquet IPv6 |
| 0x03 | Paquet IP avec compression d'en-tête |
| 0x04 – 0xFD | Réservé |
| 0xFE | Paquet de signalisation |
| 0xFF | Paquet NULL |

**length** –Ce champ spécifie le nombre d'octets suivant immédiatement le champ longueur jusqu'à la fin du conteneur TLV.

**IPv4\_packet ( )** – Indique un paquet IPv4 qui comporte un en-tête défini dans la RFC 791 [1].

**IPv6\_packet ( )** – Indique un paquet IPv4 qui comporte un en-tête défini dans la RFC 2460 [2].

**compressed\_ip\_packet ( )** – Indique un paquet qui comporte des en-têtes compressés présenté au § 4.

**signalling\_packet ( )** – Indique des signaux de contrôle de transmission présentés au § 5.

**NULL** – Ce sont des octets de bourrage fixes de 8 bits dont la valeur est «0xFF».

## 3.2 Format du paquet d'encapsulation de flux générique

L'encapsulation de flux générique (GSE, *generic stream encapsulation*) spécifié dans la norme ETSI TS 102 606 [4] permet d'encapsuler des paquets de longueur variable tels les paquets IP. Chaque paquet GSE peut comporter un champ étiquette et un champ CRC (contrôle de redondance cyclique). Les récepteurs peuvent filtrer les paquets qu'ils reçoivent en utilisant le champ étiquette de chaque paquet. Lorsque les paquets GSE sont fragmentés en éléments à insérer dans les intervalles de transmission, l'intégrité des paquets reconstitués peut être assurée par contrôle CRC.

Le protocole GSE a été conçu comme une couche adaptation assurant les fonctions d'encapsulation et de fragmentation des paquets de la couche réseau dans un Flux Générique. La GSE assure une encapsulation efficace des paquets IP dans des paquets de la couche longueur variable 2, qui sont alors directement séquencés sur la couche physique en trames en bande de base.

La GSE maximise l'efficacité du transport de paquets IP par diminution du préfixe par un facteur allant de 2 à 3 comparativement à l'encapsulation MPE dans un flux de transport MPEG. Cette diminution est obtenue sans aucun compromis sur les fonctionnalités offertes par le protocole, en raison de la taille des paquets de la couche longueur variable 2, adapté aux caractéristiques du trafic IP.

La GSE offre également des possibilités supplémentaires qui augmentent la souplesse et l'applicabilité du protocole. Certaines fonctions/caractéristiques-clés de la GSE sont:

1) Prise en charge de l'encapsulation multiprotocole (par exemple, IPv4, IPv6, MPEG, ATM, Ethernet et VLAN).

2) Transparence par rapport aux fonctions de la couche réseau, y compris le cryptage IP et la compression d'en-tête IP.

3) Prise en charge de plusieurs modes d'adressage. Outre l'adresse MAC de 6 octets (y compris la multidiffusion et la monodiffusion), elle prend en charge le mode MAC sans adresse et un mode adresse à 3 octets facultatif.

4) Mécanisme de fragmentation de paquets IP ou d'autres paquets de la couche réseau sur des trames en bande de base en vue de la prise en charge de l'ACM/VCM.

5) Prise en charge du filtrage matériel.

6) Extensibilité: des protocoles de liaisons supplémentaires peuvent être inclus via des valeurs types de protocole spécifiques (par exemple. couche sécurité 2, compression d'en-tête IP, etc.).

7) Faible complexité.

# 4 Compression d'en-tête IP (compression d'en-tête pour la diffusionHCfB)

Lorsque les paquets IP doivent être acheminés sous forme de paquets de longueur variable, il est pratique pour les services de diffusion d'avoir une grande compatibilité avec divers services utilisant des réseaux de télécommunication. Outre l'en-tête UDP de 8 octets, chaque paquet IP a généralement au moins un en-tête IPv4 de 20 octets ou un en-tête IPv6 de 40 octets. Sur la base de ces en-têtes, il est nécessaire pour les routeurs des réseaux de télécommunication de décider de la façon dont chaque paquet doit être transféré. En conséquence, ces en-têtes ont une très grande importance pour les réseaux de télécommunication. Par ailleurs, ils ne sont jamais nécessaires dans les canaux de diffusion étant donné que tous les paquets dans ces canaux sont tout simplement transférés vers les récepteurs. Le débit de transfert peut être augmenté si cette information d'en-tête non utilisée est compressée.

Le format d'un paquet IP à en-tête compressé est illustré à la Fig. 3 et dans le Tableau 3. Cela ramène la longueur des en-têtes IP et UDP à 3 ou 5 octets d'en-tête compressé pour la plupart des paquets. Lorsque le contenu est transféré sur des paquets IP, la plupart des champs dans ces en-tête restent inchangés pendant la connexion. Une fois l'en-tête compressé envoyé, ces champs avec les mêmes valeurs dans les paquets subséquents peuvent ne pas nécessairement être envoyés. Sur la base de ce principe, les en-têtes IP et UDP avec toute l'information sont envoyés à intervalles longs et les en-tête compressés sont envoyés pour la plupart de tous les paquets. Les en-têtes compressés sont rétablis au niveau du récepteur en les remplissant avec l'en-tête d'un paquet précédent qui contient toute l'information.

FIGURE 3

Format des paquets IP à en-tête compressé



TABLEAU 3

Paquet IP à en-tête compressé

|  |  |  |
| --- | --- | --- |
| Syntaxe | Nbre de bits | Mnémonique |
| compressed\_ip\_packet ( ) { |  |  |
| CID | 12 | uimsbf |
| SN | 4 | uimsbf |
| CID\_header\_type | 8 | uimsbf |
| If (CID\_header\_type==0x20) { |  |  |
| IPv4\_header\_wo\_length ( ) |  |  |
| UDP\_header\_wo\_length ( ) |  |  |
| for(i=0;i<N;i++){ |  |  |
| packet\_data\_byte | 8 | bslbf |
| } |  |  |
| } |  |  |
| else if (CID\_header\_type==0x21) { |  |  |
| Identification | 16 | bslbf |
| for(i=0;i<N;i++){ |  |  |
| packet\_data\_byte | 8 | bslbf |
| } |  |  |
| } |  |  |
| else if(CID\_header\_type==0x60) { |  |  |
| IPv6\_header\_wo\_length ( ) |  |  |
| UDP\_header\_wo\_length ( ) |  |  |
| for(i=0;i<N;i++){ |  |  |
| packet\_data\_byte | 8 | bslbf |
| } |  |  |
| } |  |  |
| else if(CID\_header\_type==0x61) { |  |  |
| for(i=0;i<N;i++){ |  |  |
| packet\_data\_byte | 8 | bslbf |
| } |  |  |
| } |  |  |
| } |  |  |

**CID** **– Context IDentification** – Indique le flux IP qui est identifié par combinaison des champs subséquents. Pour l'IPv4, il s'agit de l'adresse IP de la source, de l'adresse IP de la destination, du protocole, du numéro du port source et du numéro du port de destination. Pour l'IPv6, il s'agit de l'adresse IP de la source, de l'adresse IP de la destination, de next\_header, du numéro du port source et du numéro du port de destination.

**SN** **– Sequence Number** – Champ de 4 bits qui s'incrémente avec chaque packet avec le même identificateur CID. Le numéro de séquence SN revient à 0 après avoir atteint sa valeur maximale.

**CID\_header\_type** –Indique le type d'en-tête du paquet. Il est codé conformément au Tableau 4.

TABLEAU 4

Valeur attribuée au CID\_header\_type

|  |  |
| --- | --- |
| Valeur | Description |
| 0x00 – 0x1F | Réservé |
| 0x20 | En-tête complet de paquet avec en-têtes IPv4 et UDP |
| 0x21 | En-tête compressé de paquet avec en-têtes IPv4 et UDP |
| 0x22 – 0x5F | Réservé |
| 0x60 | En-tête complet de paquet avec en-têtes IPv6 et UDP |
| 0x61 | En-tête compressé de paquet avec en-têtes IPv6 et UDP |
| 0x62 – 0xFF | Réservé |

**Identification** –Ce champ contient l'identification IP de l'en-tête IPv4.

**IPv4\_header\_wo\_length ( )** – En-tête IPv4 sans champ total\_length ou checksum\_header représenté à la Fig. 4 et dans le Tableau 5.

FIGURE 4

Structure de l'en-tête IPv4\_header\_wo\_length ( )



TABLEAU 5

En-tête IPv4\_header\_wo\_length

|  |  |  |
| --- | --- | --- |
| Syntaxe | Nbre de bits | Mnémonique |
| IPv4\_header\_wo\_length ( ) { |  |  |
| version | 4 | uimsbf |
| IHL | 4 | uimsbf |
| type\_of\_service | 8 | bslbf |
|  |  |  |
| identification | 16 | bslbf |
| flags | 3 | bslbf |
| fragment\_offset | 13 | uimsbf |
| time\_to\_live | 8 | uimsbf |
| protocol | 8 | bslbf |
|  |  |  |
| source\_address | 32 | bslbf |
| destination\_address | 32 | bslbf |
| } |  |  |

**IPv6\_header\_wo\_length ( )** – En-tête IPv6 sans le champ payload\_length représenté à la Fig. 5 et dans le Tableau 6.

FIGURE 5

Structure de l'en-tête IPv6\_header\_wo\_length ( )



TABLEAU 6

En-tête IPv6\_header\_wo\_length

|  |  |  |
| --- | --- | --- |
| Syntaxe | Nbre de bits | Mnémonique |
| IPv6\_header\_wo\_length ( ) { |  |  |
| version | 4 | uimsbf |
| traffic\_class | 8 | bslbf |
| flow\_label | 20 | bslbf |
|  |  |  |
| next\_header | 8 | bslbf |
| hop\_limit | 8 | uimsbf |
| source\_address | 128 | bslbf |
| destination\_address | 128 | bslbf |
| } |  |  |

**UDP\_header\_wo\_length ( )** –En-tête UDP [3] sans champ longueur totale ou checksum\_header représenté à la Fig. 6 et dans le Tableau 7.

FIGURE 6

Structure de l'en-tête UDP\_header\_wo\_length ( )



TABLEAU 7

En-tête UDP\_header\_wo\_length

|  |  |  |
| --- | --- | --- |
| Syntaxe | Nbre de bits | Mnémonique |
| UDP\_header\_wo\_length ( ) { |  |  |
| source\_port | 16 | uimsbf |
| destination\_port | 16 | uimsbf |
| } |  |  |

# 5 Signaux de contrôle pour le multiplexage de paquets IP

Il est nécessaire pour le récepteur d'identifier un flux de données IP utile pour pouvoir effectuer le démultiplexage dans les signaux de diffusion.

## 5.1 Signaux de contrôle pour des paquets IP acheminés sur des paquets MPEG‑2 TS

Pour des paquets IP acheminés dans des paquets MPEG‑2 TS, en utilisant par exemple une encapsulation multiprotocole, la table de notification IP/MAC (INT) conformément à la norme ETSI EN 301 192 [5] peut être utilisée pour réaliser la résolution d'adresse IP. Avec la table INT, les récepteur sont capables d'identifier le flux de données IP utile dans les signaux de diffusion.

## 5.2 Signaux de contrôle pour des paquets IP acheminés sur des conteneurs TLV

Pour les paquets IP non acheminés dans des paquets MPEG‑2 TS, une table de localisation d'adresse AMT (Address Map Table) et une table d'information de réseau TLV TLV‑NIT (TLV‑Network Information Table) sont définies.

La table AMT est utilisée pour lister les adresses des groupes de multidiffusion IP associées à un identificateur **service\_id** identifiant le service que les canaux de diffusion proposent. La table TLV‑NIT est utilisée pour associer l'identificateur **service\_id** à l'identificateur **TLV\_stream\_id** ou autres organisations des signaux acheminés via un réseau donné et les caractéristiques du réseau lui‑même. La table TLV‑NIT est la même que la table NIT dans les systèmes MPEG‑2 sauf qu'elle est transmise par le paquet de signalisation dans le conteneur TLV.

Lorsqu'est notifié à un récepteur le flux de données IP utile, ce récepteur identifie le signal de diffusion dans lequel ce flux de données IP est multiplexé en consultant les tables AMT et TLV‑NIT, et se cale ensuite sur ce signal. Pour la notification du flux de données IP utile, les applications peuvent utiliser la MLD ou l'IGMP, qui sont largement utilisés dans les réseaux de télécommunication pour contrôler les paquets IP de multidiffusion reçus. Comme le mécanisme utilisant l'AMT ou la TLV‑NIT, les applications peuvent acquérir le flux de données IP voulu sans devoir distinguer s'il provient de canaux de diffusion ou de réseaux de télécommunication comme le montre la Fig. 7.

FIGURE 7

Applications acquérant du contenu sans faire la distinction entre canaux



### 5.2.1 Structure du format de section étendu

Les structures des signaux de contrôle de transmission sont conformes au format de section étendu représenté à la Fig. 8 et dans le Tableau 8.

FIGURE 8

Structure du format de section étendu



TABLEAU 8

Format de section étendu

|  |  |  |
| --- | --- | --- |
| Syntaxe | Nbre de bits | Mnémonique |
| signalling\_packet ( ) { |  |  |
| table\_id | 8 | uimsbf |
| section\_syntax\_indicator | 1 | bslbf |
| '1' | 1 | bslbf |
| '11' | 2 | bslbf |
| section\_length | 12 | uimsbf |
| table\_id\_extension | 16 | uimsbf |
| '11' | 2 | bslbf |
| version\_number | 5 | umisbf |
| current\_next\_indicator | 1 | bslbf |
|  |  |  |
| section\_number | 8 | uimsbf |
| last\_section\_number | 8 | uimsbf |
| for(i=0; i<N; i++) { |  |  |
| signalling\_data\_byte | 8 | bslbf |
| } |  |  |
| CRC\_32 | 32 | rpchof |
| } |  |  |

**table\_id** –Champ de 8 bits identifiant la table à laquelle appartient la section. La valeur de ce champ est indiquée dans le Tableau 9.

TABLEAU 9

Valeurs de Table\_id assignment

|  |  |
| --- | --- |
| Valeur | Description |
| 0x00 – 0x3F | Réservé |
| 0x40 | TLV-NIT (TLV-Network Information Table) (réseau réel) |
| 0x41 | TLV-NIT (TLV-Network Information Table) (tout autre réseau) |
| 0x42 – 0xFD | Réservé |
| 0xFE | La table est indiquée par la valeur du champ table\_id\_extension |
| 0xFF | Réservé |

**section\_syntax\_indicator** – Champ déterminant si un format normal ou un format d'extension est utilisé et représente des formats normaux ou d'extension respectivement, lorsque ce champ contient «0» et «1».

**section\_length** – Champ qui indique le nombre d'octets de données qui suivent ce champ et qui ne doit pas dépasser 4093.

**table\_id\_extension** – Champ étendant l'identificateur de table. Lorsque la valeur de ce champ est 0xFE, il est utilisé pour identifier la table, comme illustré dans le Tableau 10.

TABLEAU 10

Valeurs attribuées au champ table\_id\_extension

|  |  |
| --- | --- |
| Valeur | Description |
| 0x0000 | AMT (Address Map Table) |
| 0x0001 – 0xFFFF | Réservé |

**version\_number** – Champ indiquant le numéro de version de la table.

**current\_next\_indicator** –Champ contenant «0» et «1» respectivement lorsque la table est actuellement utilisée ou lorsque la table ne peut être utilisée à présent, mais qui sera valide prochainement.

**section\_number** – Champ indiquant le numéro de la première section contenant la table.

**last\_section\_number** – Champ indiquant le numéro de la dernière section contenant la table.

**signalling\_data\_byte** – Champ contenant les signaux de contrôle de transmission.

**CRC\_32** – Champ conforme à la Recommandation UIT‑T H.222.0.

### 5.2.2 Structure des signaux de contrôle de la transmission

Tous les signaux multiplexés avec des conteneurs TLV sont contrôlés par les signaux de contrôle de transmission.

– Une table TLV NIT acheminant des informations corrélant les fréquences de modulation et autres informations sur les canaux de transmission avec des programmes de diffusion.

– Une table AMT qui associe des adresses IP en spécifiant les flux de données IP avec leurs services de diffusion.

#### 5.2.2.1 Table TLV-NIT

La Fig. 9 et le Tableau 11 montrent la structure de la table TLV‑NIT.

FIGURE 9

Structure de la table TLV-NIT



TABLEAU 11

TLV-NIT

| Syntaxe | Nbre de bits | Mnémonique |
| --- | --- | --- |
| TLV\_network\_information\_table ( ){ |  |  |
| table\_id | 8 | uimsbf |
| section\_syntax\_indicator | 1 | bslbf |
| '1' | 1 | bslbf |
| '11' | 2 | bslbf |
| section\_length | 12 | uimsbf |
| network\_id | 16 | uimsbf |
| '11' | 2 | bslbf |
| version\_number | 5 | uimsbf |
| current\_next\_indicator | 1 | bslbf |
|  |  |  |
| section\_number | 8 | uimsbf |
| last\_section\_number | 8 | uimsbf |
|  |  |  |
| reserved\_future\_use | 4 | bslbf |
| network\_descriptors\_length | 12 | bslbf |
| for(i=0;i<N;i++){ |  |  |
| descriptor ( ) |  |  |
| } |  |  |
| reserved\_future\_use | 4 | bslbf |
| TLV\_stream\_loop\_length | 12 | uimsbf |
|  |  |  |

TABLEAU 11 (*fin*)

| Syntaxe | Nbre de bits | Mnémonique |
| --- | --- | --- |
| for(i=0;i<N;i++){ |  |  |
| TLV\_stream\_id | 16 | uimsbf |
| original\_network\_id | 16 | uimsbf |
| reserved\_future\_use | 4 | bslbf |
| TLV\_stream\_descriptors\_length | 12 | uimsbf |
| for(j=0;j<N;j++){ |  |  |
| descriptor ( ) |  |  |
| } |  |  |
| } |  |  |
| CRC\_32 | 32 | rpchof |
| } |  |  |

**table\_id** – Champ de 8 bits identifiant la table à laquelle la section appartient. La valeur de ce champ est montrée dans le Tableau 9.

**section\_syntax\_indicator** – Champ mis à «1» qui représente le format de section étendu.

**section\_length** – Champ de 12 bits, dont les deux premiers bits sont «00». Il spécifie le nombre d'octets de la section, commençant immédiatement après le champ section\_length et incluant le CRC. La valeur du champ section\_length ne doit pas être supérieur à 1021, de sorte que la section entière a une longueur maximale de 1 024 octets.

**network\_id** –Champ de 16 octets servant d'étiquette identifiant le système de remise, sur lequel la table TLV‑NIT donne des informations, par rapport à tout autre système de remise.

**version\_number** –Champ indiquant le numéro de version de la table.

**current\_next\_indicator** –Champ contenant «0» et «1» respectivement lorsque la table est actuellement utilisée ou lorsque la table ne peut être actuellement utilisée, mais qui sera valide prochainement.

**section\_number** – Champ indiquant le numéro de la première section contenant la table.

**last\_section\_number** – Champ indiquant le numéro de la dernière section contenant la table.

**network\_descriptor length** – La valeur des deux premiers bits de ce champ est «00». Les 10 bits restants constituent un champ qui indique le nombre d'octets dans le descripteur qui suit le champ network\_descriptors\_length.

**TLV\_stream\_loop\_length** – La valeur des deux premiers bits de ce champ est «00». les dix bits restants constituent un champ qui indique le nombre d'octets de données qui suivent ce champ.

**TLV\_stream\_id** – Champ représentant le numéro d'identification du flux TLV applicable.

**original\_network\_id** – Champ représentant le numéro d'identification du réseau d'origine du flux TLV applicable.

**TLV\_stream\_descriptors\_length** – Champ représentant le nombre d'octets dans tous les descripteurs du flux TLV applicables immédiatement après ce champ. A noter que la valeur des deux premiers bits est «00»*.*

**CRC\_32** – Champ conforme à la Recommandation UIT‑T H.222.0.

#### 5.2.2.2 Table cartographie d'adresse

La table AMT (adress map table) offre un mécanisme souple permettant d'acheminer des informations sur les services que les flux de données IP offrent dans les réseaux transférés TLV. Cette table donne une liste des adresses IP qui constituent chaque service. La Fig. 10 et le Tableau 12 montrent la structure de la table AMT.

FIGURE 10

Structure de la table AMT



TABLEAU 12

Table AMT

| Syntaxe | Nbre de bits | Mnémonique |
| --- | --- | --- |
| address\_map\_table ( ) { |  |  |
| table\_id | 8 | uimsbf |
| section\_syntax\_indicator | 1 | bslbf |
| '1' | 1 | bslbf |
| '11' | 2 | bslbf |
| section\_length | 12 | uimsbf |
| table\_id\_extension | 16 | uimsbf |
| '11' | 2 | bslbf |
| version\_number | 5 | uimsbf |
| current\_next\_indicator | 1 | bslbf |
|  |  |  |
| section\_number | 8 | uimsbf |
| last\_section\_number | 8 | uimsbf |
| num\_of\_service\_id | 10 | uimsbf |
| reserved\_future\_use | 6 | bslbf |
|  |  |  |

TABLEAU 12 (*fin*)

| Syntaxe | Nbre de bits | Mnémonique |
| --- | --- | --- |
| for (i=0; i<num\_of\_service\_id; i++) { |  |  |
| service\_id | 16 | uimsbf |
| ip\_version | 1 | bslbf |
| reserved\_future\_use | 5 | bslbf |
| service\_loop\_length | 10 | uimsbf |
| if (ip\_version=='0'){ /\*IPv4\*/ |  |  |
| src\_address\_32 | 32 | bslbf |
| src\_address\_mask\_32 | 8 | uimsbf |
| dst\_address\_32 | 32 | bslbf |
| dst\_address\_mask\_32 | 8 | uimsbf |
| } |  |  |
| else if (ip\_version=='1') { /\*IPv6\*/ |  |  |
| src\_address\_128 | 128 | bslbf |
| src\_address\_mask\_128 | 8 | uimsbf |
| dst\_address\_128 | 128 | bslbf |
| dst\_address\_mask\_128 | 8 | uimsbf |
| } |  |  |
| for (j=0; i<N; j++) { |  |  |
| private\_data\_byte | 8 | bslbf |
| } |  |  |
| } |  |  |
| CRC\_32 | 32 | rpchof |
| } |  |  |

**table\_id** – La valeur de ce champ est 0xFE, indiquant que la table est identifiée par la valeur du champ table\_id\_extension.

**section\_syntax\_indicator** – Ce champ est mis à «1» ce qui représente le format section étendu.

**section\_length** – Ce champ indique le nombre d'octets de données qui suit ce champ et ne dépasse pas 4093.

**table\_id\_extension** – La valeur de ce champ est mis à 0x0000, représentant la table Address Map Table.

**version\_number** –Champ indiquant le numéro de version de la table.

**current\_next\_indicator** –Champ contenant «0» et «1» respectivement lorsque la table est actuellement utilisée ou lorsque la table ne peut être actuellement utilisée, mais qui sera valide prochainement.

**section\_number** – Champ indiquant le numéro de la première section contenant la table.

**last\_section\_number** – Champ indiquant le numéro de la dernière section contenant la table.

**num\_of\_service\_id** – Champ indiquant le numéro d'identificateur service\_id listé dans cette table Address Map Table.

**service\_id** – Champ de 16 bits identifiant le service offert par le flux de données IP.

**ip\_version** – Champ indiquant la version de l'IP et représentant l'IPv4 et l'IPv6 respectivement lorsque ce champ contient «0» et «1».

**service\_loop\_length** – Champ représentant le nombre d'octets qui suivent ce champ jusqu'au prochain champ service\_id listé ou juste avant le champ CRC\_32.

**src\_address\_32** – Champ spécifiant une adresse IPv4 source. L'adresse IPv4 est fragmentée en 4 champs de 8 bits où le premier octet contient l'octet de plus fort poids de l'adresse IPv4 source.

**src\_address\_mask\_32** – Champ spécifiant un masque IPv4 qui indique quels octets de l'adresse IPv4 source sont utilisés pour des comparaisons. Ce nombre de bits spécifié depuis l'octet de plus fort poids est comparé aux bits dans la position équivalente du champ src\_address\_32.

**dst\_address\_32** – Champ spécifiant une adresse de destination IPv4. L'adresse IPv4 est fragmentée en 4 champs de 8 bits dont le premier octet contient l'octet de plus fort poids de l'adresse de destination IPv4.

**dst\_address\_mask\_32** – Champ spécifiant un masque IPv4 qui indique quels octets de l'adresse IPv4 de destination sont utilisés pour des comparaisons. Ce nombre de bits spécifié depuis l'octet de plus fort poids et comparé aux bits dans la position équivalente du champ dst\_address\_32.

**src\_address\_128** – **dst\_address\_32** – Champ spécifiant une adresse source IPv6. L'adresse IPv6 est fragmentée en 8 champs de 16 bits dont le premier octet contient l'octet de plus fort poids de l'adresse source IPv6.

**src\_address\_mask\_128** – Champ spécifiant un masque IPv6 qui indique quels octets de l'adresse IPv6 source sont utilisés pour des comparaisons. Ce nombre de bits spécifié depuis l'octet de plus fort poids et comparé aux bits dans la position équivalente du champ src\_address\_128.

**dst\_address\_128** – Champ spécifiant une adresse de destination IPv6. L'adresse IPv6 est fragmentée en 8 champs de 16 bits dont le premier octet contient l'octet de plus fort poids de l'adresse de destination IPv6.

**dst\_address\_mask\_128** – Champ spécifiant un masque IPv6 qui indique quels octets de l'adresse IPv6 de destination sont utilisés pour des comparaisons. Ce nombre de bits spécifié depuis l'octet de plus fort poids et comparé aux bits occupant la position équivalente du champ dst\_address\_128.

**private\_data\_byte** – La valeur de ce champ est définie de manière privée.

1. \* Il convient de porter la présente Recommandation à l'attention des Commisssions d'études 9 et 16 de l'UIT‑T. [↑](#footnote-ref-1)