



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

**CCITT**

COMITÉ CONSULTATIF  
INTERNATIONAL  
TÉLÉGRAPHIQUE ET TÉLÉPHONIQUE

**T.101**

(11/1988)

SÉRIE T: ÉQUIPEMENTS TERMINAUX ET  
PROTOCOLES POUR LES SERVICES DE  
TÉLÉMATIQUE

---

**INTERFONCTIONNEMENT INTERNATIONAL  
POUR LES SERVICES VIDÉOTEX**

Réédition de la Recommandation du CCITT T.101 publiée  
dans le Livre Bleu, Fascicule VII.5 (1988)

---

## NOTES

1 La Recommandation T.101 du CCITT a été publiée dans le fascicule VII.5 du Livre Bleu. Ce fichier est un extrait du Livre Bleu. La présentation peut en être légèrement différente, mais le contenu est identique à celui du Livre Bleu et les conditions en matière de droits d'auteur restent inchangées (voir plus loin).

2 Dans la présente Recommandation, le terme «Administration» désigne indifféremment une administration de télécommunication ou une exploitation reconnue.

© UIT 1988, 2008

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

## Recommandation T.101

### INTERFONCTIONNEMENT INTERNATIONAL POUR LES SERVICES VIDÉOTEX

(Malaga-Torremolinos, 1984; modifiée à Melbourne, 1988)

#### SOMMAIRE

#### Préambule

- 1 Objet et domaine d'application de la Recommandation
- 2 Interfonctionnement entre les services vidéotex - Considérations générales
- 3 Interfonctionnement international du service vidéotex
- 4 Interfonctionnement international entre passerelles
- 5 Interfonctionnement international entre un terminal et serveur
- 6 Attributs de classe d'architecture de contenu
- 7 Attributs du contenu
- 8 Définition formelle du type de données dépendant du vidéotex

*Annexe A* – Syntaxe de données d'interfonctionnement (SDI) décrite dans ASN.1 (Recommandation X.208)

*Annexe B* – Syntaxe de données I<sup>1)</sup>

*Annexe C* – Syntaxe de données II<sup>1)</sup>

*Annexe D* – Syntaxe de données III<sup>1)</sup>

#### **Préambule**

Le CCITT,

*considérant*

(a) que des services vidéotex ont été mis en oeuvre dans différents pays/régions en utilisant différentes syntaxes de données (syntaxe de données I, syntaxe de données II et syntaxe de données III) d'égale importance;

(b) que le CCIR étudie les normes à appliquer aux services télétexte de radiodiffusion pour réception générale et a fait savoir qu'il estimait souhaitable que les équipements terminaux des systèmes télétexte de radiodiffusion pour réception générale soient compatibles avec les systèmes à base de données fondés sur le réseau public;

(c) que différents pays/régions sont autorisés à utiliser leurs systèmes existants;

(d) que l'interfonctionnement entre les services vidéotex de différents pays peut nécessiter des techniques de transcodage et/ou de conversion;

(e) que l'interfonctionnement entre services vidéotex peut être assuré en utilisant différents types de réseaux tels que le réseau téléphonique public commuté (RTPC), le réseau public pour données avec commutation par paquets (RPDCP), le réseau public pour données avec commutation de circuits (RPDCC), le réseau numérique avec intégration de services (RNIS), etc.;

(f) que les protocoles d'interfonctionnement vidéotex devraient offrir un large degré de compatibilité avec les protocoles utilisés dans d'autres services de télématique,

---

<sup>1)</sup> Les annexes B, C et D ne seront pas publiées dans le fascicule VII.5 (Recommandations de la série T), mais feront l'objet d'une autre publication.

*recommande*

d'appliquer les dispositions techniques suivantes pour l'interfonctionnement international des services vidéotex.

## **1 Objet et domaine d'application de la Recommandation**

### *1.1 Objet*

La présente Recommandation a pour objet:

- a) de faciliter l'interfonctionnement des différents services vidéotex;
- b) d'identifier les paramètres nécessaires pour communiquer avec des terminaux vidéotex;
- c) de fournir les recommandations techniques voulues pour l'interfonctionnement éventuel d'autres services de télématique avec les services vidéotex.

### *1.2 Domaine d'application*

1.2.1 La présente Recommandation décrit les caractéristiques de l'information codée échangée entre les pays faisant partie du service vidéotex interactif international.

1.2.2 Les systèmes vidéotex sont des systèmes de communication de textes ayant en outre la possibilité d'un niveau donné de représentation picturale et un répertoire d'attributs d'affichage. Le texte et les images obtenus sur des écrans de télévision sont conformes aux normes de balayage actuellement en vigueur dans les différents pays.

1.2.3 Un choix de différentes syntaxes de données s'offre aux Administrations pour la mise en oeuvre de leurs services nationaux. Bien que la compatibilité entre ces options soit généralement bonne, il peut arriver que l'on ait à recourir à des techniques de transcodage et/ou de conversion pour faciliter l'interfonctionnement.

1.2.4 Pour les besoins du service international, différentes syntaxes de données ont été identifiées:

- a) syntaxe de données d'interfonctionnement;
- b) syntaxe de données I;
- c) syntaxe de données II;
- d) syntaxe de données III;
- e) autres syntaxes sont à l'étude.

## **2 Interfonctionnement entre les services vidéotex – Considérations générales**

2.1 Il appartient aux Administrations de décider dans quel(s) réseau(x) le ou les services vidéotex seront assurés.

2.2 Possibilités à envisager:

2.2.1 Service vidéotex fonctionnant sur le RTPC; la communication entre un terminal vidéotex et un serveur vidéotex est établie sur le RTPC.

2.2.2 Service vidéotex exploité sur le RTPC et un réseau public pour données (RPD) (généralement un RPDCP); la communication entre un terminal vidéotex relié au RTPC et un serveur vidéotex relié à un RPD est établie via un point d'accès vidéotex ou un centre de service vidéotex assurant l'interface entre les deux réseaux.

2.2.3 D'autres possibilités (RPDCC, RNIS, etc.) sont également envisageables.

2.3 L'interfonctionnement international entre des services vidéotex via des passerelles et reliés à n'importe quel réseau (RTPC, RPDCP, RPDCC, RNIS, etc.) devrait être possible. Cet interfonctionnement permet à un terminal vidéotex appartenant à un service vidéotex d'accéder à un serveur vidéotex appartenant à un autre service vidéotex. L'interfonctionnement international entre un terminal vidéotex d'un pays et un serveur vidéotex d'un autre pays peut également être possible. Tous les échanges internationaux de données devraient obéir aux spécifications contenues dans la présente Recommandation (voir la Recommandation F.300 pour la description des services).

### 3 Interfonctionnement international du service vidéotex

3.1 L'interfonctionnement vidéotex permet à un terminal vidéotex d'un pays donné d'interagir en temps réel avec une application vidéotex située dans un autre pays.

3.2 L'interfonctionnement international entre services vidéotex devrait utiliser les fonctions définies dans les syntaxes de données mises en oeuvre par les Administrations concernées: syntaxes de données I, II et III définies dans les annexes.

#### 3.3 Configurations d'interfonctionnement international

Les différentes configurations pour l'interfonctionnement international sont définies dans la Recommandation F.300. Les deux catégories principales d'interfonctionnement sont définies ci-après.

##### 3.3.1 Interfonctionnement de passerelle à passerelle

Cette catégorie d'interfonctionnement suppose une communication entre les passerelles situées dans chaque pays et où tous les processus de traitement des données qui entrent en jeu dans l'interfonctionnement sont effectués. Les protocoles et les syntaxes de données pour cette catégorie d'interfonctionnement sont spécifiés au § 4.

##### 3.3.2 Interfonctionnement de terminal à serveur

Cette catégorie d'interfonctionnement suppose une communication entre un terminal et un serveur situés dans des pays différents, soit directement, soit par l'intermédiaire d'une unité de conversion située dans le pays du terminal. Plusieurs configurations ont été identifiées. Les protocoles et les syntaxes de données pour chacune de ces configurations sont spécifiés au § 5.

### 4 Interfonctionnement international entre passerelles

L'interfonctionnement international entre passerelles permet à un terminal vidéotex situé dans un pays A d'avoir accès aux services vidéotex situés dans un pays B via un service vidéotex du pays A. La configuration de l'interfonctionnement international entre passerelles est illustrée dans la figure 1/T.101.

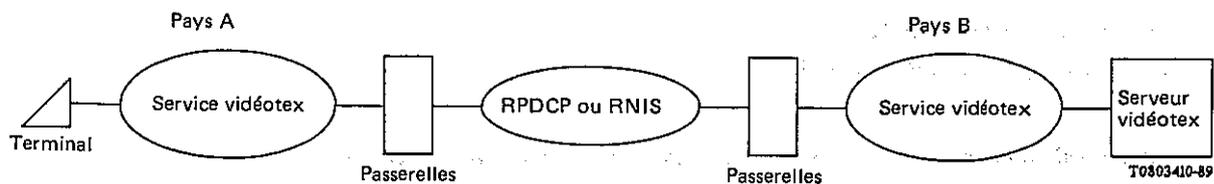


FIGURE 1/T.101

#### 4.1 Interfonctionnement international au niveau du réseau

4.1.1 L'interfonctionnement international entre services vidéotex devrait de préférence avoir lieu entre réseaux du même type (RPDCP, RPDCC, RNIS et lignes louées) lorsque ces réseaux sont fournis par les deux Administrations intéressées.

4.1.2 La définition du service de réseau de l'interconnexion des systèmes ouverts pour les applications du CCITT est donnée dans la Recommandation X.213.

4.1.3 Quand l'interfonctionnement a lieu entre services vidéotex fonctionnant sur différents types de réseaux, la Recommandation X.75 devrait s'appliquer. L'interfonctionnement avec le RNIS devrait être conforme à la Recommandation T.90.

## 4.2 *Couche transport*

Le service de couche transport de l'interconnexion des systèmes ouverts pour les applications du CCITT est défini dans la Recommandation X.214.

Le protocole de transport de l'interconnexion des systèmes ouverts pour les applications du CCITT est spécifié dans la Recommandation X.224.

Les classes 0 (correspondant à la Recommandation T.70) et 2 peuvent toutes deux être utilisées.

Lorsqu'on choisit la classe 0, le protocole utilisé est totalement compatible avec la Recommandation T.70 du CCITT. Si l'on choisit la classe 2, il faut utiliser un contrôle de flux explicite.

## 4.3 *Couche session*

Le service de couche session de l'interconnexion des systèmes ouverts pour les applications du CCITT est défini dans la Recommandation X.215. Le protocole de session de l'interconnexion des systèmes ouverts pour les applications du CCITT est spécifié dans la Recommandation X.225.

L'utilisation des protocoles de session par l'interfonctionnement vidéotex est définie dans la Recommandation T.523.

## 4.4 *Couche présentation*

### 4.4.1 *Protocole de présentation*

Le service de couche présentation de l'interconnexion des systèmes ouverts pour les applications du CCITT est défini dans la Recommandation X.216. Le protocole de présentation de l'interconnexion des systèmes ouverts pour les applications du CCITT est spécifié dans la Recommandation X.226.

L'utilisation des protocoles de présentation par l'interfonctionnement vidéotex est définie dans la Recommandation T.523.

### 4.4.2 *Codage des informations vidéotex*

#### *Codage du contenu de l'élément données d'affichage*

Le contenu vidéotex est conforme à l'une des différentes syntaxes de données. Une syntaxe de données, appelée syntaxe de données d'interfonctionnement, est décrite dans l'annexe A. Il existe trois syntaxes de données, fondées sur la Recommandation T.50, et appelées syntaxe de données I, syntaxe de données II et syntaxe de données III. Elles sont décrites dans l'annexe B, l'annexe C et l'annexe D respectivement. Les quatre annexes font partie intégrante de la présente Recommandation.

Différentes Administrations qui mettent en oeuvre un service vidéotex peuvent utiliser une des trois syntaxes de données précitées.

Si deux pays mettent en oeuvre la même syntaxe de données, celle-ci peut être utilisée pour l'interfonctionnement vidéotex entre ces deux pays.

Si un pays met en oeuvre une syntaxe de données alors qu'un autre pays met en oeuvre une syntaxe de données différente, l'interfonctionnement vidéotex entre les deux pays peut:

- i) utiliser la syntaxe de données d'interfonctionnement comme syntaxe intermédiaire. Un transcodage/conversion vers et à partir de la syntaxe de données d'interfonctionnement par les deux pays peut être nécessaire; ou
- ii) utiliser l'une des deux syntaxes de données, avec transcodage/conversion effectués soit dans le pays d'origine soit dans le pays de destination.

Pour l'identification d'une syntaxe de données particulière en cours d'utilisation (I, II ou III), il est possible d'utiliser les séquences d'échappement pour la désignation et l'invocation des «codes complets»:

ESC 2/5 4/3 pour la syntaxe de données I

ESC 2/5 4/4 pour la syntaxe de données II

ESC 2/5 4/1 pour la syntaxe de données III

L'environnement du «code complet» sera quitté à résection soit d'une séquence:

ESC 2/5 4/0

ou par la désignation et l'invocation de n'importe quel autre code complet.

#### 4.5 Couche application

L'élément du service de contrôle d'association (ACSE) de l'interconnexion des systèmes ouverts pour les applications du CCITT est défini dans la Recommandation X.217. Le protocole d'ACSE de l'interconnexion des systèmes ouverts pour les applications du CCITT est spécifié dans la Recommandation X.227.

La couche d'application pour l'interfonctionnement vidéotex utilise les Recommandations ci-après:

- Rec. T.400: Introduction à l'architecture de documents, au transfert et à la manipulation. Introduction générale.
- Rec. T.411: Architecture des documents ouverte (ODA) et format d'échange. Introduction et principes généraux.
- Rec. T.412: Architecture des documents ouverte (ODA) et format d'échange. Structures des documents.
- Rec. T.414: Architecture des documents ouverte (ODA) et format d'échange. Profil d'un document.
- Rec. T.415: Architecture des documents ouverte (ODA) et format d'échange. Format ouvert d'échange des documents (ODIF).

La couche d'application pour l'interfonctionnement vidéotex utilise le service DTAM (transfert et manipulation des documents) et le protocole décrit dans les Recommandations T.431, T.432, T.433.

La couche d'application pour l'interfonctionnement vidéotex utilise les structures d'exploitation décrites dans la Recommandation T.441.

La Recommandation T.564 décrit le profil d'application d'interfonctionnement vidéotex et les caractéristiques des passerelles.

La Recommandation T.504 décrit le profil d'application des documents pour l'interfonctionnement vidéotex.

La Recommandation T.523 décrit le profil d'application de communication pour l'interfonctionnement vidéotex.

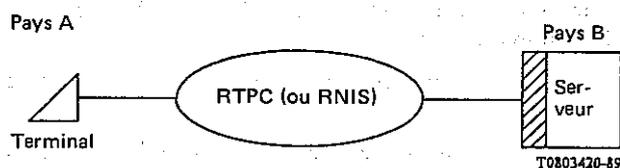
La Recommandation T.541 décrit le profil d'application des structures opérationnelles pour l'interfonctionnement vidéotex.

#### 4.6 Relation avec DTAM/ODA

Des relations avec l'architecture de document (Recommandation T.412) et le format d'échange de document (Recommandation T.415) sont exprimées par les attributs de la classe architecture de contenu et les attributs de contenu décrits aux § 6, 7 et 8.

### 5 Interfonctionnement international entre un terminal et serveur

#### 5.1 Accès via un RNIS ou un service support RTPC

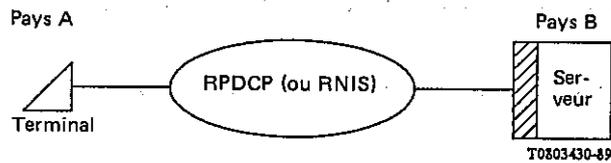


Dans cette configuration, le terminal utilise le RTPC international (ou les services supports du RNIS) pour atteindre le serveur. Sur la liaison internationale, les protocoles ci-après devraient être utilisés:

- |  |  |
|--|--|
| couches 1 à 3 via RTPC:                                | protocoles définis par le serveur;                                       |
| couches 1 à 3 via service support RNIS <sup>2)</sup> : | Recommandation T.90;   |
| couches 4 à 7:   | protocoles (le cas échéant) définis par le serveur situé dans le pays B; |
| syntaxe de données:                                    | syntaxe de données définie par le serveur;                               |
| fonctions dialogue/service:                            | fonctions définies par le serveur.                                       |

<sup>2)</sup> Les protocoles à utiliser dans le téléservice vidéotex du RNIS doivent faire l'objet d'un complément d'étude.

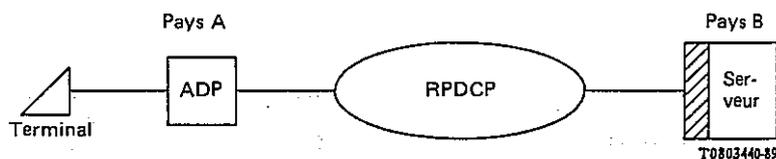
## 5.2 Accès via le RPDCP ou un service support RNIS



Dans cette configuration, le terminal utilise le RPDCP international (ou les services supports du RNIS) pour atteindre le serveur. Sur la liaison internationale, les protocoles suivants devraient être utilisés:

couches 1 à 3 via le RPDCP:	Recommandation X.75;
couches 1 à 3 via service support RNIS:	(Recommandation T.90);
couches 4 à 7:	les protocoles (le cas échéant) définis par le serveur situé dans le pays B;
syntaxe de données:	syntaxe de données définie par le serveur;
fonctions dialogue/service:	fonctions définies par le serveur.

## 5.3 Accès via le RPDCP et un ADP



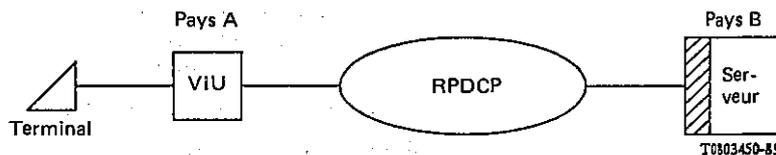
Dans cette configuration, le terminal est relié à un ADP qui donne accès au RPDCP international; le terminal et l'ADP sont tous deux situés dans le pays A. Le type de connexion entre le terminal et l'ADP relève de chaque pays (en général, il s'agit du RTPC ou d'une ligne louée).

L'accès au serveur du pays B peut se faire par l'intermédiaire du RPDCP international. Le type de connexion entre le serveur et le RPD national relève de chaque pays (généralement il s'agit d'une ligne louée).

Sur la liaison internationale, il convient d'utiliser les protocoles ci-après:

couches 1 à 3:	Recommandation X.75;
au-dessus de la couche 3:	Recommandation X.29 + Recommandation X.3;
syntaxe de données:	syntaxe de données définie par le serveur dans le pays B;
fonctions dialogue/service:	fonctions définies par le serveur.

## 5.4 Accès via le RPDCP à travers une VIU



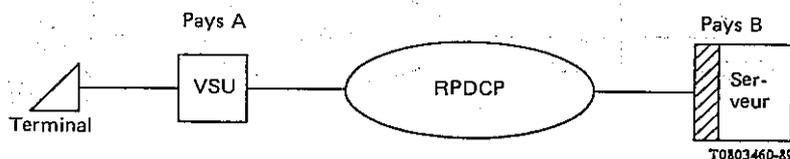
Dans cette configuration, le terminal est relié à une VIU (unité d'interface vidéotex) qui donne accès au RPDCP international; le terminal et la VIU sont situés dans le pays A. Le type de connexion entre le terminal et la VIU relève de chaque pays (généralement il s'agit du RTPC ou d'une ligne louée). La VIU remplit deux fonctions: elle supporte les terminaux et convertit les syntaxes de données. Il appartient à l'Administration du pays A de décider des modalités de mise en oeuvre d'une VIU: ce peut être sous forme d'un système séparé ou intégré avec un équipement existant (ADP ou point d'accès au vidéotex par exemple).

L'accès au serveur du pays B peut se faire via le RPDCP international. Le type de connexion entre le serveur et le RPDCP national relève de chaque pays (généralement il s'agit d'une ligne louée).

Sur la liaison internationale, les protocoles suivants devraient être utilisés:

couches 1 à 3:	Recommandation X.75;
au-dessus de la couche 3:	Recommandation X.29 + Recommandation X.3. Des protocoles basés sur la Recommandation X.200 pourraient aussi être utilisés. Dans ce cas, les profils d'application devront être définis dans la série des Recommandations T.500; pour étude ultérieure;
syntaxe de données:	celle définie par le serveur situé dans le pays B;
fonctions dialogue/service:	celles définies par le serveur.

### 5.5 Accès via un RPDCP par l'intermédiaire d'une VSU



Dans cette configuration, le terminal est relié à une VSU (unité de service vidéotex) qui donne accès au RPDCP international; le terminal et la VSU sont situés dans le pays B. Une VSU est une VIU qui est en outre chargée de s'occuper de la taxation et de la comptabilité de l'application. Il appartient à l'Administration du pays A de décider d'établir ou non une VSU et, le cas échéant, des modalités de mise en oeuvre: ce peut être sous forme de système séparé ou intégré avec un équipement existant (ADP, point d'accès vidéotex ou centre de service vidéotex).

Le serveur du pays B peut être atteint via le RPDCP international. Le type de connexion entre le serveur et le RPDCP national relève de chaque pays (généralement il s'agit d'une ligne louée).

Sur la liaison internationale, les protocoles ci-après devraient être utilisés:

couches 1 à 3:	Recommandation X.75;
au-dessus de la couche 3:	soit des protocoles basés sur la Recommandation X.200. Dans ce cas, les profils d'application devront être définis dans les Recommandations de la série T.500. Cela nécessite un complément d'étude; soit les Recommandations X.29 et X.3 peuvent être utilisées. Des extensions (règles d'application) à la Recommandation X.29 sont nécessaires (voir le § 5.6);
syntaxe de données:	celle définie par le serveur situé dans le pays B;
fonctions dialogue/service:	celles définies par le serveur.

### 5.6 Règles d'application pour que la Recommandation X.29 permette d'assurer les fonctions administratives

Lorsqu'une communication internationale est établie via une VSU utilisant la Recommandation X.29, des commandes vidéotex X.29 peuvent être utilisées pour permettre la transmission des taxes d'application (le cas échéant) du serveur.

Les commandes vidéotex doivent être renvoyées en séquences complètes de paquets avec le bit Q mis à un.

Les commandes vidéotex utilisent un codage T(type)-L(longueur)-V(valeur). Les commandes de longueur fixe n'ont pas besoin d'indicateur de longueur. Lorsqu'il est utilisé, l'indicateur de longueur est codé sur deux octets et définit la longueur totale en octets dans le champ V.

Pour distinguer les commandes vidéotex des commandes ADP actuellement définies par la Recommandation X.29, la valeur du type d'une commande vidéotex est définie avec le bit de poids le plus fort mis à un.

Pour permettre l'échange de données de taxation sur les liaisons internationales, les valeurs suivantes sont proposées:

- Administration (99H) L2 taxation (82H) L2 paramètre-de-taxation

Le paramètre de taxation peut prendre soit la valeur opération-du-service (80H), soit la valeur opération-applicative (81H). Les deux valeurs peuvent figurer dans un même paramètre de taxation:

- opération-du-service (80H) L1 paramètre-de-service
- opération-applicative (81H) L1 paramètre-applicatif

Le paramètre de service est dépendant du temps et peut correspondre soit au tarif par unité de taxation (80H) ou à la durée d'une unité de taxation ou à une combinaison des deux:

- montant (80H) L1 valeur
- durée (81H) L1 valeur

Le paramètre applicatif peut être basé sur les pages (80H), sur le temps (81H) ou sur une transaction (82H) ou sur une combinaison:

- page (80H) L1 valeur
- temps (81H) L1 paramètre-de-taxation-à-la-durée
- transaction (82H) L1 valeur

Le paramètre de taxation à la durée est organisé comme le paramètre de service.

L1 désigne les codes de longueur à 1 octet.

L2 désigne les codes de longueur à 2 octets.

Les exemples ci-dessous montrent les mécanismes de codage:

```
Administration L2 Charging L2 (Service) (Application)
(99H)          (82H)
(Service) ::= Service L1 (Amount L1 value) (Period L1 value)
            (80H)      (80H)          (81H)
(Application) ::= Application L1 (Frame) (Time) (Transaction)
              (81H)
(Frame) ::= Frame L1 value
         (80H)
(Time) ::= Time L1 (Amount L1 value) (Period L1 value)
        (81H)      (80H)          (81H)
(Transaction) ::= Transaction L1 value
              (82H)
```

## 6 Attributs de classe d'architecture de contenu

### 6.1 Classe d'architecture de contenu

La valeur de l'attribut «classe d'architecture de contenu» de la description d'un élément de base conforme à cette Recommandation T.101 est un ASN.1 OBJECT IDENTIFIER ayant la valeur:

{ 0 1 8 16 3 }

## 6.2 *Type de contenu*

L'attribut de classe d'architecture de contenu «type de contenu» ne peut pas être utilisé pour spécifier la classe d'architecture de contenu définie dans cette Recommandation.

## 7 **Attributs du contenu**

### 7.1 Type de codage

Classification	Peut prendre une valeur par défaut
Applicabilité	Classe d'architecture de contenu vidéotex
Structure	ASN.1 OBJECT IDENTIFIER
Valeurs permises	ASN.1 OBJECT IDENTIFIER
{ 0 1 8 16 4 }	pour «codage SDI»
{ 0 1 8 16 5 }	pour «codage de la syntaxe de données I»
{ 0 1 8 16 6 }	pour «codage de la syntaxe de données II»
{ 0 1 8 16 7 }	pour «codage de la syntaxe de données III»
Valeur par défaut	«Codage SDI»
Définition	Pour l'interfonctionnement vidéotex, les valeurs possibles correspondent aux syntaxes de données décrites dans les annexes A, B, C et D de cette Recommandation.

### 7.2 *Attributs de codage spécifiques*

Ces attributs fournissent l'information supplémentaire nécessaire au codage et au décodage de l'information du contenu, ainsi que d'autres informations intrinsèques à la partie contenu et au type de codage.

#### 7.2.1 *Sous-ensemble*

Classification	Peut avoir une valeur par défaut
Applicabilité	Classe d'architecture de contenu vidéotex Type de «codage SDI»
Valeurs	Nombre entier [0, 1 à 5, 81 à 92]
Valeur par défaut	0
Définition	Cet attribut identifie le sous-ensemble fonctionnel (rang au profil) utilisé dans l'IDS. La valeur 0 est utilisée quand il n'est spécifié aucun sous-ensemble.

#### 7.2.2 *Rang*

Classification	Peut avoir une valeur par défaut
Applicabilité	Classe d'architecture de contenu vidéotex Type de codage «syntaxe de données I»
Valeurs	Entier [0, 1 à 5]
Valeur par défaut	0
Définition	Cet attribut identifie le rang utilisé dans la syntaxe de données I. La valeur 0 est utilisée lorsque le rang n'est pas spécifié.

#### 7.2.3 *Profil*

Classification	Peut avoir une valeur par défaut
Applicabilité	Classe d'architecture de contenu vidéotex Type de codage «syntaxe de données II»
Valeurs	Entier [0, 81 à 92]
Valeur par défaut	0
Définition	Cet attribut identifie le profil utilisé dans la syntaxe de données II. La valeur 0 est utilisée lorsque le profil n'est pas spécifié.

## 8 Définition formelle du type de données dépendant du vidéotex

### 8.1 Introduction

Ce paragraphe contient la définition formelle dans la notation ASN.1 (définie dans la Recommandation X.208) du type de données correspondant aux attributs applicables au vidéotex.

Ce type de données est:

- le type de données nécessaire à la représentation des attributs de codage.

### 8.2 Représentation des attributs de codage spécifiques

```
Videotex-coding-attributes ::= CHOICE {
    subset [0] IMPLICIT subset OPTIONAL
    rank   [1] IMPLICIT rank OPTIONAL
    profile [2] IMPLICIT profile OPTIONAL }

Subset a) ::= INTEGER { undefined (0)
                    rank 1 (1)
                    rank 2 (2)
                    rank 3 (3)
                    rank 4 (4)
                    rank 5 (5)
                    profile 1 (81)
                    profile 2 (82)
                    profile 3 (83)
                    profile 4 (84)
                    profile X1-1 (85) b)
                    profile X1-2 (86)
                    profile X1-3 (87)
                    profile X1-4 (88)
                    profile X2-1 (89)
                    profile X2-2 (90)
                    profile X2-3 (91)
                    profile X2-4 (92) }

Rank ::= INTEGER { undefined (0)
                 rank 1 (1)
                 rank 2 (2)
                 rank 3 (3)
                 rank 4 (4)
                 rank 5 (5) }

Profile ::= INTEGER { undefined (0)
                   profile 1 (81)
                   profile 2 (82)
                   profile 3 (83)
                   profile 4 (84)
                   profile X1-1 (85)
                   profile X1-2 (86)
                   profile X1-3 (87)
                   profile X1-4 (88)
                   profile X2-1 (89)
                   profile X2-2 (90)
                   profile X2-3 (91)
                   profile X2-4 (92) }
```

<sup>a)</sup> L'utilisation de sous-ensemble au sein de la SID nécessite un complément d'étude.

<sup>b)</sup> Profil X<sub>i,j</sub>: profil géométrique X<sub>i</sub> avec profil alphamosaïque j.

TABLEAU 1/T.101

Profil d'application de document vidéotex	0 1 8 16 0
Profil d'application de communication DM1	0 1 8 16 1
Profil d'application opérationnel vidéotex	0 1 8 16 2
Classe d'architecture de contenu de la Recommandation T.101	0 1 8 16 3
Type de codage	
SDI	0 1 8 16 4
Syntaxe de données I	0 1 8 16 5
Syntaxe de données II	0 1 8 16 6
Syntaxe de données III	0 1 8 16 7
Contexte d'application	0 1 8 16 8

## ANNEXE A

(à la Recommandation T.101)

### **Syntaxe de données d'interfonctionnement (SDI) décrite dans ASN.1 (Recommandation X.208)**

#### *Préambule*

Pour les communications vidéotex:

- a) si deux pays appliquent la même syntaxe de données, l'interfonctionnement peut utiliser la même syntaxe de données (SD I, SD II ou SD III);
- b) si deux pays appliquent deux syntaxes de données différentes, l'interfonctionnement peut utiliser:
  - i) la syntaxe de données d'interfonctionnement (SDI) définie ci-après; ou
  - ii) l'une quelconque des trois syntaxes de données, avec conversion directe SD I/SD II/SD III. Les syntaxes de données peuvent être identifiées par le mécanisme ESC 2/5 F décrit au § 4.4.2 du corps de la Recommandation T.101.

En cas d'utilisation de la SDI, l'Administration du pays où se trouve la base de données sera responsable de la conversion vers la SDI, et l'Administration du pays où se trouve le terminal de l'utilisateur sera responsable de la conversion à partir de la SDI.

En cas d'utilisation de la méthode de conversion directe au lieu de la SDI, la SDI pourrait servir de guide technique pour la spécification du processus de conversion.

La SDI n'est pas destinée à être utilisée pour les communications entre terminal et unité centrale.

## A.1 Page vidéotex

Dans la syntaxe de données d'interfonctionnement (SDI), une page vidéotex est une séquence de commandes de présentation exprimées d'une manière indépendante de l'une quelconque des syntaxes de données de terminal. Cette manière de formuler l'information de présentation qui compose une page vidéotex a pour but de faciliter l'interfonctionnement entre des syntaxes de données de terminal fondamentalement différentes. Elle permet d'obtenir ce résultat en isolant les éléments spécifiques et les éléments communs entre les diverses syntaxes de données. La syntaxe de données d'interfonctionnement n'est pas destinée à être utilisée comme syntaxe de données de terminal en tant que telle. Un des modes de codage de la syntaxe de données d'interfonctionnement est celui défini dans la Recommandation X.209. D'autres types de codage doivent encore être étudiés:

```
Videotex-Page ::= SEQUENCE OF Presentation-Commands
Presentation-Commands ::= SEQUENCE { State-Vector, Function-&-Parameters }
```

## A.2 Vecteur d'état

On définit un vecteur d'état avec chaque commande de présentation pour établir la relation entre cette commande de présentation et chacune des autres commandes de présentation. L'information contenue explicitement dans le vecteur d'état est aussi contenue implicitement dans chaque commande de présentation; toutefois, il faudrait que l'appareil de conversion comprenne parfaitement chacune des trois syntaxes de données axées sur les terminaux pour faire apparaître cette information. C'est pourquoi un vecteur d'état est inclus dans chaque commande de présentation dans laquelle un état global est influencé ou dans laquelle on rencontre une valeur limite, afin que le processus de conversion puisse s'effectuer à un niveau général.

```
State-Vector ::= CHOICE { [1] Vector-Definition,
                          [2] Reset-State-Vector,
                          [3] NULL }
```

### A.2.1 Définition vecteur

```
Vector-Definition ::= SEQUENCE { Global-State-Affected-Indicator,
                                  Terminal-Model-Precedence,
                                  Boundary-Condition-Definition }
```

-- Seule doit être communiquée l'information qui varie entre les vecteurs d'état. Si une composante du vecteur ne varie pas, il est inutile de communiquer cette composante. Cela signifie que le vecteur d'état n'est pas communiqué souvent et n'introduit pas une marge notable.

#### A.2.1.1 Indicateur d'état global influencé

L'indicateur d'état global influencé véhicule l'information relative aux états globaux de la syntaxe de données de présentation. Les variables d'état global sont des variables représentant les états de la syntaxe de données de présentation qui sont établis par les commandes de présentation et qui influencent les résultats des commandes de présentation subséquentes. En déclarant explicitement les variables d'état global, le processus de conversion n'a pas besoin de comprendre les relations mutuelles existant entre les commandes de présentation. Cela signifie que le processus de conversion n'a pas besoin de simuler un terminal de la syntaxe des données de source pour effectuer la conversion de ses éléments.

L'indicateur d'état global influencé ne transporte pas d'information sur la valeur qui a été donnée à un état global. Cette information est contenue dans l'élément `fonction et paramètre` de la SDI. L'indicateur signale simplement les états qui ont subi des changements. Cela est d'une grande importance dans les cas où le processus de conversion doit faire le tri des commandes de présentation pour expliquer les différences intervenant dans le modèle de terminal utilisé à la source et à la destination de l'échange. En cas de changement de l'ordre de succession des commandes de présentation, le processus de conversion doit établir les variables globales appropriées avant chaque commande présente dans la séquence modifiée. En consultant l'indicateur d'état global influencé, le processus de conversion peut identifier les états globaux qui doivent être rétablis. Par exemple, s'il est nécessaire de trier les commandes de présentation pour passer d'un modèle de terminal multiplan à un modèle à un seul plan, et s'il est nécessaire d'utiliser des commandes de couleur, l'indicateur d'état global influencé indiquera que l'état de couleur approprié doit être établi avant chaque paquet des données triées.

Dans certaines syntaxes de données de terminal, les attributs ont des effets globaux; dans d'autres syntaxes de données, les effets sont localisés, selon le type de la primitive de visualisation utilisé. Par exemple, dans la syntaxe de données III, une commande de couleur demeure opérante pour toutes les primitives, jusqu'à la commande de couleur suivante; par contre, dans la syntaxe de données II, il existe plusieurs états de couleur qui s'appliquent indépendamment à des primitives différentes, par exemple couleur LIGNE, couleur ZONE REMPLIE, etc. L'indicateur d'état global influencé transporte une référence à un nombre de `vecteurs d'état d'attributs` indépendants qui définissent le contexte des attributs. Dans les syntaxes de données qui utilisent seulement des paramètres globaux, il suffit de référencer un seul vecteur d'état d'attribut`. Pour d'autres syntaxes de données, qui utilisent plusieurs attributs localisés, il est possible de référencer plusieurs `vecteurs d'état d'attributs`.

```

Global-State-Affected-Indicator ::= SEQUENCE {
attribute-state-vector-reference      INTEGER,
attribute-affected-indicators        SEQUENCE OF {
                                        INTEGER {
current-text-position                (1),
current-foreground-colour            (2),
current-auxiliary-colour             (3),
lining-state                         (4),
flash-blink-state                   (5),
basic-char-size-state               (6),
conceal-state                       (7),
char-invert-box-state               (8),
char-marking-state                 (9),
screen-protection-state             (10),
display-control-state               (11),
device-control-state                (12),
cursor-control-state                (13),
geometric-control-1-state           (14),
geometric-control-2-state           (15),
wait-state                          (16),
general-text-state                  (17),
p-text-state                        (18),
geometric-text-state                (19),
DRCS-definition-state               (20),
macro-definition-state              (21),
texture-pattern-state               (22),
music-part-memory-state              (23),
animation-configuration-state       (24),
workstation-configuration-state     (25)
                                        }
}

```

-- L'indicateur d'état global influencé est constitué par un ensemble de drapeaux indicateurs qui identifient des états globaux particuliers ou, dans certains cas, des catégories d'états globaux qui peuvent être modifiés par des commandes de présentation et d'autres commandes. Pour plus de simplicité, certains états, par exemple toutes les catégories de processus de clignotement, sont regroupés.

### A.2.1.2 *Modèle de terminal*

Le modèle de terminal diffère notablement entre les diverses syntaxes de données de terminal. Pour la présentation des images statiques, cette différence se manifeste par la manière dont les commandes de présentation se superposent. Une image créée pour un modèle de terminal multiplan peut être représentée sur un terminal utilisant un modèle de terminal à un seul plan ou un terminal multiplan ayant un ordre de priorité différent pour les plans, en triant les commandes de présentation de telle manière qu'elles constituent une image équivalente. L'opération de tri est nécessaire car, faute de ce tri, l'ordre de constitution pourrait être contradictoire avec l'ordre de priorité dans le nouvel environnement. L'indicateur de priorité de modèle de terminal est simplement un indicateur numérique de la priorité de superposition pour les commandes de présentation prévues par la syntaxe de données de terminal source. Le processus de conversion est indépendant du modèle de terminal d'une syntaxe de données particulière; il a simplement pour effet de trier les commandes de présentation sur la base de cet indicateur. A noter que certaines commandes, comme les réinitialisations, qui produisent un effet dans plusieurs plans, pourraient devoir être répétées dans différentes parties de la séquence de présentation après le tri. L'indicateur de priorité du modèle de terminal est constitué par une séquence de chiffres qui indiquent l'effet d'une commande dans le modèle de terminal.

Terminal-Model-Precedence ::= INTEGER

- L'indicateur de priorité du modèle de terminal est un chiffre qui indique l'ordre de priorité de l'information de présentation identifiée. Le chiffre `1` indique que l'information identifiée a la priorité la plus élevée et qu'elle devrait être placée devant toutes les autres informations actuellement affichées. Le chiffre `2` indique que l'information identifiée est au deuxième niveau de priorité et qu'elle devrait être placée après l'information du niveau `1` mais devant tous les autres niveaux d'information. Par exemple, les données de texte et mosaïque de la syntaxe de données II sont une information du niveau `1`, tandis que l'information géométrique peut être du niveau `1` ou du niveau `2`. Dans la syntaxe de données III, toute l'information est au niveau `1`, parce que l'ordre de priorité dans lequel elle est affichée est déterminé exclusivement par l'ordre dans lequel l'information est communiquée. Dans la syntaxe de données I, l'ordre n'est pas fixé, parce que la priorité de certains `plans` de mémoire peut être modifiée par la commande ATTRIBUER UNE TRAME.
- La valeur `0` a une signification particulière pour l'indicateur de priorité du modèle de terminal. Elle indique que l'information identifiée nécessite une interprétation spéciale. Cette information spéciale englobe des commandes de réinitialisation partielle qui influencent plusieurs couches du modèle de terminal, ainsi que des commandes qui ont un effet fonction du temps, à savoir ATTENTE, caractère BEL et APPARITION D'AFFICHAGE.

### A.2.1.3 *Conditions aux limites*

Les variables conditions aux limites représentent les limites à l'intérieur desquelles la commande de présentation considérée a été définie. Chaque commande de présentation admet son interprétation normale seulement dans un certain intervalle de valeurs. Par exemple, le nombre de caractères pouvant être affiché sur l'écran varie selon les syntaxes de données des terminaux sources; en conséquence, la présentation d'un caractère unique ne peut pas être considérée comme identique dans toutes les syntaxes de données de terminal. Pour mettre en facteur la communauté de conception, on fait des identifications séparées pour la condition limite de rencontrer le bord de la zone d'affichage pour la présentation d'un caractère. Cela facilite la conversion, car cela signifie que les conditions aux limites applicables à chaque commande de présentation sont spécifiées explicitement. De ce fait, le processus de conversion est indépendant des conditions aux limites internes à l'intérieur de chacune des syntaxes de données des terminaux aux sources.

Boundary-Condition-Definition ::= SET { [1] Screen-Dimensions,  
[2] Colour-Map-Limit,  
[3] Presentation-Sub-Area,  
[4] Char-Mode-Constraints,  
[5] Coordinate-Limit-Polygon,  
[6] Coordinate-Limit-Spline,  
[7] Presentation-Resolution,  
[8] Macro-Seg-Memory-Limit,  
[9] DRCS-Memory-Limit,  
[10] Direct-Colours-Limit }

#### A.2.1.3.1 Dimensions d'écran

Screen-Dimensions ::= SEQUENCE { INTEGER, INTEGER }

- Dimensions d'écran indique le rapport des dimensions de l'écran de visualisation, exprimé par une fraction entre les dimensions Y et X, le nombre ENTIER représentant une fraction binaire avec un point binaire implicite devant le bit de plus grand poids. A noter que la dimension (1,1) signifie qu'il n'y a pas de restriction géométrique. Un service avec mode de caractères pourrait utiliser (1,1) pour indiquer l'absence de restriction.

#### A.2.1.3.2 Limite de la carte de couleurs

Colour-Map-Limit ::= INTEGER

- La limite de la carte de couleurs indique le nombre maximal des couleurs pouvant être stockées dans une seule carte de couleurs, ou le total combiné pour des cartes de couleurs multiples. Elle représente le nombre maximal d'états de couleurs que l'on peut rencontrer dans une page de présentation donnée. Si on n'utilise pas de carte de couleurs, l'entier spécifie le nombre de couleurs fixes.

#### A.2.1.3.3 Sous-zone de présentation

Presentation-Sub-Area ::= SEQUENCE { Abs-Coord, Rel-Coord, INTEGER, INTEGER }

- Les deux coordonnées donnent les dimensions aux limites d'une sous-zone de l'écran d'affichage, en termes de dimensions de la sous-zone et par le nombre de caractères par rangée et nombre de colonnes. La coordonnée absolue spécifie l'origine de la sous-zone, la coordonnée relative la dimension de la sous-zone, et les coordonnées ENTIER la limite imposée respectivement au nombre de caractères par rangée et au nombre de ligne.

#### A.2.1.3.4 Contraintes sur le mode caract.

Char-Mode-Constraints ::= SEQUENCE { INTEGER, INTEGER }

- Les deux paramètres indiquent la limite imposée au nombre de caractères par rangée et au nombre de rangées du texte qui peut être présenté sur l'écran d'affichage, c'est-à-dire les limites pour lesquelles il y aura enroulement et défilement des caractères (ou des mots).

#### A.2.1.3.5 Polygone limite de coordonnées

Coordinate-Limit-Polygon ::= INTEGER

- La limite de coordonnées par polygone spécifie le nombre maximal des coordonnées pouvant être spécifiées pour un polygone rempli.

#### A.2.1.3.6 Courbe de lissage limite de coordonnées

Coordinate-Limit-Spline ::= INTEGER

- La limite de coordonnées par courbe de lissage spécifie le nombre maximal des coordonnées pouvant être spécifiées.

#### A.2.1.3.7 *Résolution de présentation*

Presentation-Resolution ::= SEQUENCE { INTEGER, INTEGER }

- La résolution de présentation spécifie la résolution nominale de l'écran d'affichage qui a été utilisé par la source d'information.

#### A.2.1.3.8 *Limite de mémoire macro-seg.*

Macro-Seg-Memory-Limit ::= INTEGER

- La limite de mémoire macro spécifie la limite supérieure imposée à la capacité de mémoire disponible pour stocker les macro-instructions ou les segments. Le paramètre ENTIER représente la mémoire disponible exprimée en multipléts.

#### A.2.1.3.9 *Limite de mémoire de JCDR*

DRCS-Memory-Limit ::= INTEGER

- La limite de mémoire de JCDR spécifie la limite supérieure imposée à la capacité de mémoire disponible pour stocker le JCDR. Le paramètre ENTIER représente la mémoire disponible exprimée en multipléts.

#### A.2.1.4 *Identificateur de syntaxe de données (ISD)*

SID ::= IMPLICIT INTEGER { data-syntax-I (1),  
data-syntax-II (2),  
data-syntax-III (3) }

- ISD est un identificateur qui est référencé dans un certain nombre de commandes de primitive et qui identifie la syntaxe de données source de la commande.

#### A.2.2 *Réinitialisation du vecteur d'état*

Reset-State-Vector ::= SEQUENCE { SID, Vector-Definition }

- La commande réinitialisation vecteur d'état sert à établir l'état initial pour la syntaxe de données d'interfonctionnement. L'état par défaut peut être choisi dans la table correspondant à la syntaxe ou au profil de données du terminal source qui figure dans l'appendice II. D'autres paramètres peuvent être spécifiées, par le recours aux définitions explicites du vecteur, des fonctions et des paramètres d'état.

#### A.2.3 *NUL*

NUL signifie que le vecteur d'état n'a pas varié depuis la commande de présentation précédente.

#### A.3 *Fonctions et paramètres*

Les fonctions et paramètres constituant les commandes de présentation sont groupés en catégories qui dépendent de leur communauté de spécification entre les diverses syntaxes de données de terminal. Les fonctions compatibles, par exemple le répertoire de base de caractères alphanumériques spécifié dans la Recommandation T.51, définissent des groupes séparés. Les fonctions uniques, par exemple certains caractères spécifiques spéciaux, définissent aussi des groupes séparés, pour pouvoir être converties ou traitées d'une autre manière spéciale. Des fonctions telles que le JCDR et les commandes de dessin de caractères graphiques, qui diffèrent fondamentalement entre les diverses syntaxes de données de terminal, sont organisées de telle sorte que les possibilités sous-jacentes communes puissent être exploitées dans le processus de conversion nécessaire.

```

Functions-&-Parameters ::= CHOICE {
    [0] Alpha-Char-String,
    [1] Special-Char-String,
    [2] Kana-Char-String,
    [3] Kanji-Char-String,
    [4] Block-Mosaic-String,
    [5] Smooth-Mosaic-String,
    [6] Special-Mosaic-String,
    [7] Format-Effector-C0-Chars,
    [8] Special-Format-C0-Characters,
    [9] General-Control-Characters,
    [10] Geometric-String,
    [11] Animation-Control-String,
    [12] Segment-Control-String,
    [13] Colour-Control-String,
    [14] Text-Control-String,
    [15] Photo-Graphic-String-Syntetic-Image,
    [16] Photo-Graphic-String-Natural-Image,
    [17] MACRO-String,
    [18] DRCS-String,
    [19] Fill Pattern-Control-String,
    [20] Music-String,
    [21] Tele-Software-String,
    [22] Audio-Data-String,
    [23] Greek-Char-String }

```

Les six premières catégories de fonctions et la dernière portent sur divers caractères de texte ou mosaïques. Aucune des syntaxes de données de terminal définies dans la Recommandation T.101 ne couvre la totalité de ces caractères. Il existe plusieurs caractères spécifiques dans chacune de ces syntaxes. Toutefois, une grande partie du répertoire est commune aux différentes syntaxes de données de terminal, même si les caractères sont susceptibles d'être codés différemment. Etant donné que la question du codage n'est pas traitée ici et que l'emploi de tableaux de codage pourrait en fait créer une grande confusion, les caractères extraits des différents répertoires de caractères seront repérés par les codes de nom d'identificateur pour chaque caractère, comme défini dans la Recommandation T.51. Dans la Recommandation T.101, aucune des syntaxes de données axées sur les terminaux n'utilise explicitement ces codes de nom dans le corps de la Recommandation; en conséquence, la totalité du répertoire de caractères, avec les noms de code pour chaque caractère, est jointe ici en appendice.

### A.3.0 Chaîne de caract. alpha

Alpha-Char-String ::= GRAPHICSTRING

- Caractères (LA01 à LZ30, ND01 à ND09 et ND10, SC01 à SC05, SP01 à SP22, SA01 à SA07, NS01 à NS03, NF01 à NF21, SM01 à SM44 et SM47 à SM49, et SD11 à SD43) pris dans le répertoire 1; ce sont des caractères faisant partie des jeux de caractères primaires et supplémentaires de la Recommandation T.51, complétés par le caractère ESPACEMENT (SP01) et OBLITÉRATION (SM34).
- Le codage des caractères faisant partie d'une chaîne de caractères alpha sera pris dans le tableau de code des caractères primaires IRV (numéro d'enregistrement 2 de l'ISO 2375), et le tableau de code secondaire à utiliser avec IRV dans ISO 6937/2 (numéro d'enregistrement 90 de l'ISO).
- *Remarque* – Le codage du caractère \$ «symbole du dollar» (SC02) sera pris dans le jeu de caractères supplémentaire.
- *Remarque* – Le codage du caractère ## «signe numéro» (SM01) sera pris dans le jeu de caractères supplémentaire.
- *Remarque* – Le codage du caractère «symbole monétaire universel» (SC01) sera pris dans le jeu de caractères primaire.

### A.3.1 *Chaîne de caractères spéciaux*

Special-Char-String ::= INTEGER { non-spacing-vector-overbar (1),  
non-spacing-slant (2),  
left-vertical-bar-jointive (3),  
right-vertical-bar-jointive (4) }

- Barre vectorielle autre que d'espacement est un caractère (SM50) du répertoire 2.
- Barre oblique autre que d'espacement est un caractère (SM51) du répertoire 2.
- Barre verticale jointive à gauche est un caractère (SM45) du répertoire 2.
- Barre verticale jointive à droite est un caractère (SM46) du répertoire 2.

### A.3.2 *Chaîne de caract. Kana*

Kana-Char-String ::= GRAPHICSTRING

- Ces caractères (JA01 à JA63) sont pris dans le répertoire 3.
- Le codage des caractères faisant partie d'une chaîne de caractères kana sera pris dans le tableau de code des caractères kana (numéro d'enregistrement 56 de l'ISO dans ISO 2375).

### A.3.3 *Chaîne de caract. Kanji*

Kanji-Char-String ::= GRAPHICSTRING

- Ces caractères (JK01 à JK2980, HK01 à HK83 et JS01 à JS366) sont pris dans le répertoire 4.
- Le codage des caractères faisant partie d'une chaîne de caractères kanji sera pris dans le tableau de code à deux multiplats des caractères kanji (numéro d'enregistrement 87 de l'ISO dans l'ISO 2375).
- *Remarque* – Les caractères de ce tableau de code à deux multiplats qui chevauchent d'autres jeux de caractères vidéotex définis ne sont pas considérés comme faisant partie du répertoire 4; ils sont donc communiqués comme des caractères du répertoire 1, du répertoire 3 ou du répertoire 8, selon le cas. Il s'agit en l'occurrence des caractères alphanumériques latins (LA01 à LZ30) et des caractères non alphabétiques (ND01 à ND09 et ND00, SC01 à SC05, SP01, SP02, SP04 à SP15, SP17 à SP22, SA01 à SA07, NS02 à NS03, NF01 à NF05, SM01 à SM14, SM19, SM24 à SM34, SM38, SM43, SM44, SM47, SM48 et SD11 à SD43) du répertoire 1, des caractères kana (JA01 à JA63) du répertoire 3 et des caractères de dessin (DG01 à DG04, DG13 à DG24 et DG32) du répertoire 8, qui ont un autre codage dans le tableau de code à deux multiplats, mais qui figurent dans d'autres répertoires.

### A.3.4 *Chaîne de mosaïques de bloc*

Block-Mosaic-String ::= GRAPHICSTRING

- Caractères mosaïques de bloc (MG01 à MG63) pris dans le répertoire 7.
- Le codage des caractères du sous-répertoire des caractères mosaïques de bloc est le même dans les trois syntaxes de données de terminal définies dans la Recommandation T.101 du CCITT. Le jeu de caractères est enregistré avec le numéro 129 de l'ISO dans l'ISO 2375.

### A.3.5 *Chaîne de mosaïques lissées*

Smooth-Mosaic-String ::= CHOICE { [1] Sub-Cell-Aligned-Smooth-Mosaics,  
[2] General-Smooth-Mosaics }

#### A.3.5.1 *Mosaïques lissées alignées en sous-cellules*

Sub-Cell-Aligned-Smooth-Mosaics ::= GRAPHICSTRING

- Caractères mosaïques lissés (SG01 à SG56) pris dans le répertoire 8.

- Le codage des caractères du sous-répertoire des mosaïques lissées alignées en sous-cellules est le même dans les deux syntaxes de données de terminal définies dans la Recommandation T.101 du CCITT, qui contiennent ces caractères. Ceux-ci sont enregistrés sous les numéros 71 et 72 dans ISO 2375.

#### A.3.5.2 Mosaïques lissées générales

General-Smooth-Mosaics ::= GRAPHICSTRING

- Caractères mosaïques lissés (MS01 à MS28) pris dans le répertoire 9.
- Le codage des caractères du sous-répertoire des mosaïques lissées générales est pris dans la syntaxe de données de terminal de la Recommandation T.101 du CCITT, qui contient ces caractères. Ce tableau de code est enregistré sous le numéro 137 dans ISO 2375.

#### A.3.6 Chaîne de mosaïques spéciales

Special-Mosaic-String ::= CHOICE { [1] Drawing-Characters,  
[2] Other-Special-Mosaics }

##### A.3.6.1 Caractères de dessin

Drawing-Characters ::= GRAPHICSTRING

- Caractères de dessin (DG01 à DG50) pris dans le répertoire 10.

##### A.3.6.2 Autres mosaïques spéciales

Other-Special-Mosaics ::= INTEGER { open-left-half-oval (1),  
open-right-half-oval (2),  
filled-left-half-oval (3),  
filled-right-half-oval (4),  
reverse-left-half-oval (5),  
reverse-right-half-oval (6) }

- Demi-ovale ouverte à gauche est un caractère mosaïque spécial (MS13) du répertoire 11.
- Demi-ovale ouverte à droite est un caractère mosaïque spécial (MS14) du répertoire 11.
- Demi-ovale à gauche remplie est un caractère mosaïque spécial (MS30) du répertoire 11.
- Demi-ovale à droite remplie est un caractère mosaïque spécial (MS30) du répertoire 11.
- Demi-ovale à gauche inverse est un caractère mosaïque spécial (MS15) du répertoire 11.
- Demi-ovale à droite inverse est un caractère mosaïque spécial (MS31) du répertoire 11.

Les catégories de fonctions et de paramètres 7 et 8 contiennent des caractères de commande de base qui servent à commander l'état de présentation des caractères de texte alphanumérique et mosaïques (y compris le JCDR). Ces caractères de commande peuvent être rangés dans deux catégories: caractères de commande de mise en page et caractères de commande de format spécial. Les caractères de commande de mise en page ont fondamentalement la même signification dans chacune des syntaxes de données de terminal. La seule différence réside dans la manière dont les fonctions appelées par ces caractères de commande interagissent avec le modèle de terminal et les conditions de visualisation des diverses syntaxes de données de terminal; par exemple, elles peuvent s'appliquer à un seul plan de l'affichage dans un modèle de terminal multiplan ou à tous les plans de l'affichage. Le codage des caractères de mise en page est aussi compatible entre les syntaxes de données de terminal.

En général, les caractères de commande de format spécial de la catégorie 8 ont une signification spéciale qui ne se retrouve pas dans toutes les syntaxes de données. Ces fonctions doivent être spécialement converties pendant l'interfonctionnement, même entre des syntaxes de données qui semblent conférer la même signification à une fonction de commande particulière. La raison en est qu'il existe de grandes différences entre les modèles de terminal et les conditions d'affichage des diverses syntaxes de données de terminal. Le caractère Bell fait partie de cette catégorie parce qu'il exige un traitement spécial dû au moment de la présentation. S'il est nécessaire de trier les commandes de présentation dans le processus de conversion de l'interfonctionnement, pour prendre en compte les différences dans un modèle de terminal - par exemple le traitement destiné à un terminal multiplan ou à un terminal à un seul plan - il faut changer le moment de présentation du caractère Bell.

### A.3.7 Caractère C0 de mise en page

Format-Effector-C0-Char ::= GRAPHICSTRING

- Caractères C0 de mise en page (APB, APF, APD, APU, CS, APR, APH) pris dans la Recommandation T.101 SD I, II, III; (position dans le tableau de code C0: 0/8 à 0/13 et 1/14 respectivement).
- APB – Position active vers l'arrière - analogue à ISO 646 (FE<sub>0</sub> BS)
- APF – Position active vers l'avant – (FE<sub>1</sub> HT)
- APD – Position active vers le bas – (FE<sub>2</sub> LF)
- APU – Position active vers le haut – (FE<sub>3</sub> VT)
- CS – Effaçage d'écran – (FE<sub>4</sub> FF)
- APR – Retour position active – (FE<sub>5</sub> CR)
- APH – Position active vers la position de départ

### A.3.8 Caractère C0 spécial de format

Special-Format-C0-Char ::= CHOICE { [1] Bell-Character,  
[2] Position-Set,  
[3] Cancel-Macro,  
[4] Non-Selective-Reset,  
[5] Cancel-Row }

#### A.3.8.1 Caractère Bell

Bell-Character ::= GRAPHICSTRING

- Caractère C0 spécial (BEL) pris dans la Recommandation T.101 SD I, III (position dans le jeu C0: 0/7).
- *Remarque* – Cette fonction fournit un signal audio à l'utilisateur du dispositif terminal. Elle n'est pas disponible dans toutes les syntaxes de données de terminal et ne peut être simulée d'une manière raisonnable.

#### A.3.8.2 Ensemble de positions

Position-set ::= SEQUENCE { INTEGER, INTEGER }

- Cette fonction fournit la possibilité équivalente de la commande d'ensemble de positions actives (APS) de la Recommandation T.101 SD I, III, et de la partie positions de la commande d'adresse de position active (APA) de SD II.
- Paramètres servant à établir la nouvelle position active sur l'écran comme comptage des cellules de caractères «de taille actuelle» de la position supérieure gauche «de départ».

### A.3.8.3 *Suppression de macro-instruction*

Cancel-Macro ::= GRAPHICSTRING

-- Caractère C0 spécial [CAN (Macro)] pris de la Recommandation T.101 SD I, III (position dans le jeu C0: 1/8).

### A.3.8.4 *Réinitialisation non sélective*

Non-Selective-Reset ::= SEQUENCE { [1] NSR-Code,  
[2] Position-Set OPTIONAL }

NSR-Code ::= GRAPHICSTRING

-- Caractère C0 spécial (NSR) pris de la Recommandation T.101 SD I, III (position dans le jeu C0: 1/15). La séquence de paramètres de positionnement est facultative.

### A.3.8.5 *Suppression de rangée*

Cancel-Row ::= GRAPHICSTRING

-- Caractères C0 spéciaux [CAN (Row)] pris de la Recommandation T.101 SD II (position dans le jeu C0: 1/8).

### A.3.9 *Caractères de commande généraux*

La catégorie 9 de fonctions et paramètres contient des fonctions de commande générales qui servent à commander l'état général de la présentation. La signification de ces caractères dépend étroitement du modèle de terminal et des conditions de visualisation de la syntaxe de données de terminal dans lesquelles ils sont utilisés. Un transcodage et une conversion sont nécessaires pour chacune des fonctions appelées par ces caractères de commande. Ceux-ci ont été organisés en un certain nombre de sous-catégories qui correspondent au domaine de fonctionnalité considéré.

General-Control-Characters ::= CHOICE { [1] Other-Format-Effectors,  
[2] Lining-Control,  
[3] Character-Size-Control,  
[4] Flash-Control,  
[5] Conceal-Control,  
[6] Invert-Control,  
[7] Window/Box-Control,  
[8] Marking-Control,  
[9] Protection-Control,  
[10] Display-Control,  
[11] Device-Control,  
[12] Cursor-Control,  
[13] Reset-Control }

Le présent paragraphe traite des caractères additionnels de mise en page qui doivent faire l'objet d'un traitement spécial dans la conversion entre les diverses syntaxes de données.

Une fonction de répétition existe dans toutes les syntaxes de données; toutefois, les effets secondaires de cette fonction diffèrent d'une syntaxe de données à une autre. La syntaxe de données de terminal SD I fournit une fonction qui permet de répéter le jeu de caractères G qui précède immédiatement, ou le couple de caractères qui précède immédiatement dans le cas d'un caractère graphique codé composite ou d'un caractère d'accent sans espacement. Dans les syntaxes de données de terminal SD II et SD III, le caractère considéré est exclusivement un caractère graphique (caractère de texte alphanumérique ou caractère mosaïque du répertoire, ou encore caractère du JCDR). Ces restrictions doivent être prises en considération lorsqu'on spécifie le processus de conversion. Ici, on considérera que la fonction de répétition répète tout caractère précédent du jeu G et des tests doivent être effectués dans l'interprétation de la SDI pour éliminer les cas erronés.

Les fonctions maintien mosaïque et effaçage mosaïque existent dans une seule syntaxe de données. Elles nécessitent une interprétation spéciale car il n'existe pas de fonctions analogues dans les autres syntaxes de données de terminal.

#### A.3.9.1 *Autres caractères de mise en page*

**Other-Format-Effectors ::= CHOICE { [1] Repeat-N,  
[2] Repeat-EOL,  
[3] Hold-Mosaic,  
[4] Release-Mosaic }**

##### A.3.9.1.1 *Répétition-N*

**Repeat-N ::= SEQUENCE { SID, RPT-Par }**

-- Caractère spécial indiquant la fonction RÉPÉTITION de la Recommandation T.101 SD I [position dans le jeu C1: 5/8, (9/8)], SD II [position dans le jeu C0: 1/2], SD III [position dans le jeu C1: 4/6, (8/6)].

**RPT-Par ::= INTEGER**

-- Comptage des répétitions.

##### A.3.9.1.2 *Répétition-EOL*

**Repeat-EOL ::= SID**

-- Caractère spécial indiquant la fonction RÉPÉTITION JUSQU'À Fin de ligne, de la Recommandation T.101 SD I [position dans le jeu C1: 5/8, (9/8) avec paramètre 0], SD III [position dans le jeu C1: 4/7, (8/7)].

##### A.3.9.1.3 *Maintien mosaïque*

**Hold-Mosaic ::= SID**

-- Caractère spécial indiquant la fonction maintien mosaïque (HMS) de la Recommandation T.101 SD II [position (série) dans le jeu C1: 5/14, (9/14)].

##### A.3.9.1.4 *Effaçage mosaïque*

**Release-Mosaic ::= SID**

-- Caractère spécial indiquant la fonction effaçage mosaïque (RMS) de la Recommandation T.101 SD II [position (série) dans le jeu C1: 5/15, (9/15)].

#### A.3.9.2 *Commande de renforcement*

La fonction de renforcement permet d'afficher un soulignement comme partie de la forme d'un caractère graphique, pour les caractères du répertoire 1. Ce soulignement est considéré comme faisant partie de l'image de la cellule du caractère, avant application d'une rotation. Dans le cas particulier de l'affichage des caractères mosaïques, la fonction de renforcement établit une police «mosaïque séparée». La possibilité de traitement des mosaïques séparées existe dans les trois syntaxes de données de terminal; toutefois, il y a des différences entre les niveaux de possibilité. Dans la syntaxe SD II, on dispose directement d'une seule taille de séparation pour les mosaïques séparées. Dans les syntaxes SD I et SD III, la valeur de la séparation est définie par le paramètre (pel logique) largeur du trait (taille du point de dessin) dans les commandes de dessin géométrique. Il est possible de convertir directement les mosaïques séparées de base entre les différentes syntaxes de données. Etant donné que la variation de séparation ne peut pas être obtenue directement dans l'une des syntaxes de données de terminal, elle doit être simulée à l'aide du JCDR. Bien entendu, cette simulation des mosaïques séparées consommerait des ressources du JCDR, qui sont limitées; il faut par conséquent tenir compte des conditions aux limites.

Lining-Control ::= INTEGER { start-lining (1),  
stop-lining (2) }

- Début de renforcement est une fonction prise de la Recommandation T.101 SD I et II [position dans le jeu C1: 5/10, (9/10)] et (DÉBUT DE SOULIGNEMENT) de SD III [position dans le jeu C1: 5/9, (9/9)].
- Fin de renforcement est une fonction prise de la Recommandation T.101 SD I et II [position dans le jeu C1: 5/9, (9/9)] et (FIN DE SOULIGNEMENT) de SD III [position dans le jeu C1: 5/10, (9/10)].

### A.3.9.3 Commande de taille des caractères

Les diverses syntaxes de données de terminal offrent la possibilité d'établir une large gamme de tailles de caractère pour le texte alphanumérique de base, les mosaïques et les caractères JCDR. De plus, la syntaxe SD II permet de définir séparément des tailles de caractère totalement variables pour des textes définis comme une partie de la section géométrique de SD II. Ces données de «texte géométrique» servent uniquement à annoter les images géométriques dans la partie géométrique facultative de SD II; il est donc inutile de les considérer comme faisant partie de la traduction du texte alphanumérique de base. De son côté, SD III fournit une seule forme de texte. En conséquence, des opérations telles que les tailles dynamiques et les rotations de texte doivent être traitées comme des parties du processus de conversion entre syntaxes de données.

La possibilité de modifier les textes, les caractères mosaïques et les caractères JCDR pour leur donner des tailles arbitraires n'existe pas dans toutes les syntaxes de données; l'image affichée subira donc une certaine dégradation dans la conversion entre une syntaxe de données et une autre. Il faut éviter de perdre de l'information de texte dans le processus de conversion, car cette information pourrait avoir une importance capitale pour la compréhension de la page de vidéotex. De même, il faut éviter d'enrouler ou de faire défiler arbitrairement l'information de texte, car cela aurait pour effet de mutiler l'information mosaïque. Dans certains cas, le processus de conversion doit choisir automatiquement une taille plus petite pour la cellule de caractère, afin d'éviter la perte d'information. Les commandes relatives à la taille des caractères indiquent la taille de la cellule de caractère prévue dans la syntaxe de données de terminal pour représenter les données relatives à la source. La cellule de caractère résultante, dans la forme convertie, peut être plus petite, selon les possibilités de la syntaxe de données de terminal et selon la condition aux limites appliquée.

Il existe deux fonctions distinctes pour définir les caractères à hauteur double. Cela est dû à une différence dans la définition de la relation entre la cellule de caractère à hauteur double et la position de la ligne de base dans une partie d'une des syntaxes de données de source. La syntaxe de données SD II offre la possibilité de définir des caractères à hauteur double qui, d'une part, ont une hauteur double au-dessus de la ligne de base et, d'autre part, se prolongent au-dessous de cette ligne; c'est la raison pour laquelle on dispose de deux fonctions dans ce cas. Comme les deux autres syntaxes de données de terminal offrent une seule possibilité de hauteur double, il est nécessaire d'appliquer une conversion avec repositionnement de la ligne de base.

Character-Size-Control ::= CHOICE { [1] Normal-Size,  
[2] Double-Size-Up,  
[3] Double-Width,  
[4] Double-Height-Up,  
[5] Double-Height-Down,  
[6] Small-Size,  
[7] Medium-Size,  
[8] Double-Size-Down }

#### A.3.9.3.1 Taille normale

Normal-Size ::= SID

- Fonction prise de la Recommandation T.101, SD I [position dans le jeu C1: 4/10, (8/10)], SD II [position dans le jeu C1: 4/12, (8/12)], SD III [position dans le jeu C1: 4/12, (8/12)].
- *Remarque* – La «taille normale» du texte est définie par les conditions aux limites de chacune des syntaxes de données de terminal; elle diffère d'une syntaxe à une autre. La largeur de la cellule de caractère «normale» est égale à 1/40 de la largeur de l'écran dans SD II et SD III, mais les largeurs d'écran ne sont pas exactement les mêmes. La valeur par défaut de la taille «normale» de cellule de caractère dans SD I est de 1/31 de la largeur; elle peut toutefois être redéfinie par la commande P-TEXT de SD I. De la même manière, la hauteur verticale d'une cellule de caractère diffère entre les diverses syntaxes de données de terminal. Cette commande indique que la syntaxe de données de terminal source implique l'utilisation de taille «normale» implicite dans cette syntaxe. La valeur de «taille normale» doit être communiquée explicitement dans le vecteur d'état associé à cette commande.

#### A.3.9.3.2 Taille double vers le haut

Double-Size-Up ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/15, (8/15)], (TAILLE DOUBLE) SD III [position dans le jeu C1: 4/15, (8/15)], et (DBS 4/5) de SD I [position dans le jeu C1: 4/11, (8/11) suivi du paramètre 4/5].
- *Remarque* – La largeur et la hauteur de la cellule de caractère sont deux fois plus grandes que celles définies par la commande «taille normale».

#### A.3.9.3.3 Largeur double

Double-Width ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/14, (8/14)], et (DBW 4/4) de SD I [position dans le jeu C1: 4/11, (8/11) suivi du paramètre 4/4].
- *Remarque* – La largeur de la cellule de caractère est deux fois plus grande que celle définie par la commande «taille normale».

#### A.3.9.3.4 Hauteur double vers le haut

Double-Height-Up ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: (8/13)], (HAUTEUR DOUBLE) de SD III [position dans le jeu C1: 4/13, (8/13)], (DBH 4/1) dans SD I [position dans le jeu C1: 4/11, (8/11) suivi du paramètre 4/1].
- *Remarque* – La hauteur de cellule de caractère est deux fois plus grande que celle définie par la commande «taille normale»; elle s'élève jusqu'à deux hauteurs de cellule de caractère au-dessus de la ligne de base.

#### A.3.9.3.5 Hauteur double vers le bas

Double-Height-Down ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/13, (8/13)].
- *Remarque* – La hauteur de cellule de caractère est deux fois plus grande que celle définie par la commande «taille normale»; le caractère à hauteur double s'élève jusqu'à une hauteur de cellule au-dessus de la ligne de base et s'abaisse jusqu'à une hauteur de cellule au-dessous de la ligne de base.

#### A.3.9.3.6 Petite taille

Small-Size ::= SID

- Fonction prise de la Recommandation T.101 SD I [position dans le jeu C1: 4/8, (8/8)] et (PETIT TEXTE) de SD III [position dans le jeu C1: 4/10, (8/10)].
- *Remarque* – La largeur et la hauteur de la cellule de caractère sont celles définies par la commande «taille normale».

#### A.3.9.3.7 Taille moyenne

Medium-Size ::= SID

- Fonction prise de la Recommandation T.101 SD I [position dans le jeu C1: 4/9, (8/9)] et (TEXTE MOYEN) de SD III [position dans le jeu C1: 4/11, (8/11)].
- *Remarque* – La taille de la cellule de caractère est définie comme étant une taille intermédiaire. Cette taille intermédiaire est définie par les conditions aux limites de chacune des syntaxes de données de source qui utilisent cette fonction de commande. Dans les données provenant de la syntaxe de données SD III, la taille moyenne est définie comme égale à 1/32 de la largeur normalisée de la plage d'affichage et 3/64 de la hauteur

de la zone unitaire normalisée. Dans les données provenant de la syntaxe de données SD I, le texte de taille moyenne prend la moitié de la hauteur de cellule de caractère et la largeur totale, définies par la commande «taille normale».

#### A.3.9.3.8 Taille double vers le bas

Double-Size-Down ::= SID

-- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/15, (8/15)].

#### A.3.9.4 Commande de clignotement

Le fonctionnement de la commande de clignotement dépend du modèle de terminal correspondant à la syntaxe de données de source considérée. Dans un terminal «multiplan», les cellules de caractère peuvent comporter un premier plan et un fond implicites qui alternent pendant le clignotement. Dans un terminal «à un seul plan», la possibilité clignotement est réalisée par le recours à la mise en correspondance des couleurs. Il est possible d'opérer une conversion entre ces deux variantes au moment du clignotement. En plus d'une possibilité de clignotement de base déclenchée par les caractères de commande, chacune des syntaxes de données de terminal offre aussi la possibilité d'établir un clignotement dynamique complexe à l'aide d'autres commandes. Dans la conversion de la commande de clignotement, il est important d'inclure une référence à la condition aux limites imposée par le nombre de couleurs dans la carte de couleurs et la structure de plans du modèle de terminal.

Flash-Control ::= SEQUENCE { Flash-Rate, Flash-Mode }

Flash-Rate ::= CHOICE { [1] Flash,  
[2] Steady,  
[3] Phase1-Flash,  
[4] Phase2-Flash,  
[5] Phase3-Flash,  
[6] Increment-Flash,  
[7] Decrement-Flash,  
[8] Blink-Stop }

Flash-Mode ::= CHOICE { [1] Normal,  
[2] Inverted-Flash,  
[3] Reduced-Intensity-Flash }

##### A.3.9.4.1 Clignotement

Flash ::= SID

-- Fonction prise de la Recommandation T.101 SD II [position dans le C1: 4/8, (8/8)], (FLC 4/0) de SD I [position dans le jeu C1: 5/1, (9/1) suivi du paramètre 4/0] et (DÉBUT DE CLIGNOTEMENT) de SD III [position dans le jeu C1: 4/14, (8/14)].

-- *Remarque* – Etablissement d'un clignotement avec cycle de 50%, du premier plan vers le fond ou entre deux adresses de la carte de couleurs choisies implicitement pour produire un effet équivalent à celui du clignotement premier plan/fond. La fonction clignotement est semblable dans les trois syntaxes de données de source mais le rythme de clignotement n'est pas nécessairement la même.

##### A.3.9.4.2 Stable

Steady ::= SID

-- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/9, (8/9)], (FLC 4/15) de SD I [position dans le jeu C1: 5/1, (9/1) suivi du paramètre 4/15].

-- *Remarque* – Neutralise l'application de tout attribut de clignotement.

#### A.3.9.4.3 Clignotement inversé

Inverted-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/0 4/1), (FLC 4/7) de SD I [position dans le jeu C1: 5/1, (9/1) suivi du paramètre 4/7].
- *Remarque* – Etablissement d'un clignotement inversé avec cycle de 50%, du premier plan vers le fond.

#### A.3.9.4.4 Clignotement à intensité réduite

Reduced-Intensity-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/1 4/1), (FLC 4/7) de SD I [position dans le jeu C1: 5/1, (9/1) suivi du paramètre 4/7].
- *Remarque* – Etablissement d'un clignotement à intensité réduite entre des adresses de la carte de couleurs.

#### A.3.9.4.5 Clignotement phase 1

Phase1-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/2 4/1), (FLC 4/4) de SD I [position dans le jeu C1: 5/1, (9/1) suivi du paramètre 4/4].
- *Remarque* – Etablissement d'un clignotement avec cycle de 33%, du premier plan vers le fond, commençant en phase 1.

#### A.3.9.4.6 Clignotement phase 2

Phase2-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/3 4/1), (FLC 4/2) de SD I [position dans le jeu C1 5/1, (9/1) suivi du paramètre 4/2].
- *Remarque* – Etablissement d'un clignotement avec cycle de 33%, du premier plan vers le fond, commençant en phase 2.

#### A.3.9.4.7 Clignotement phase 3

Phase3-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/4 4/1), (FLC 4/1) de SD I [position dans le jeu C1: 5/1, (9/1) suivi du paramètre 4/1].
- *Remarque* – Etablissement d'un clignotement avec cycle de 33%, du premier plan vers le fond, commençant en phase 3.

#### A.3.9.4.8 Incrémentation clignotement

Increment-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/5 4/1).
- *Remarque* – Etablissement d'un clignotement avec cycle de 33%, du premier plan vers le fond, avec incrémentation de la référence de phase.

#### A.3.9.4.9 Décrémentation clignotement

Decrement-Flash ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/6 4/1).

- *Remarque* – Etablissement d'un clignotement avec cycle de 33%, du premier plan vers le fond, avec décrémentation de la référence de phase.

#### A.3.9.4.10 *Fin de clignotement*

Blink-Stop ::= SID

- Fonction prise de la Recommandation T.101 SD III [position dans le jeu C1: 5/14, (9/14)].
- *Remarque* – Arrêt de tous les processus de clignotement.

#### A.3.9.5 Commande de masquage

La fonction masquage affichage est conçue pour être appliquée à un modèle de terminal doté de plusieurs plans indépendants. Les données stockées dans les cellules de caractère peuvent être marquées masquées, auquel cas le fond de la cellule sera affiché avec la même couleur que le premier plan. Une commande locale apparition d'affichage aurait pour effet d'afficher le premier plan dans les couleurs définies initialement. Une conversion est nécessaire pour appliquer cette fonction à un terminal à un seul plan. Cette possibilité peut être simulée soit par l'utilisation d'une macro-instruction déclenchée par touche, qui contient une définition du premier plan des cellules de caractère masquées, soit par la carte de couleurs. La définition de la séquence de macro-instructions déclenchée par touche doit être établie au cours d'un processus de tri dans la procédure de conversion; elle est limitée par la disponibilité de la mémoire de macros. L'emploi de la carte de couleurs pour simuler cette fonction a pour effet de consommer très rapidement les ressources de la carte. De ce fait, l'application de la fonction de masquage doit avoir la dernière priorité dans l'utilisation de ces ressources. Les fonctions de commande masquage et fin de masquage sont incluses ici, afin qu'elles puissent être traitées de la façon la plus efficace par le processus de conversion.

Conceal-Control ::= CHOICE { [1] Conceal-Display,  
[2] Stop-Conceal-Display }

##### A.3.9.5.1 *Masquage d'affichage*

Conceal-Display ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 5/8, (9/8)] et SD I [position dans le jeu C1: 5/2, (9/2) suivi du paramètre 4/0].
- *Remarque* – Etablissement de l'attribut d'état de masquage.

##### A.3.9.5.2 *Fin de masquage d'affichage*

Stop-Conceal-Display ::= SID

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 4/2) et SD I [position dans le jeu C1: 5/2, (9/2) suivi du paramètre 4/15].
- *Remarque* – Arrêt de l'application de l'attribut d'état de masquage.

#### A.3.9.6 *Commande d'inversion*

Invert-Control ::= CHOICE { [1] Invert-Polarity,  
[2] Normal-Polarity }

- *Remarque* – Inversion de l'application des attributs de couleur du premier plan et du fond dans le cas d'un modèle de terminal multiplan, et inversion des couleurs superposées (premier plan) et sous-jacentes (fond) dans le cas d'un terminal à un seul plan. Ces commandes produisent des effets essentiellement identiques lorsqu'on génère une présentation avec chacun des modèles de terminal identifiés; toutefois, l'effet est très différent quand cette commande est utilisée pour modifier l'attribut d'un caractère graphique déjà affiché. Dans la conversion, ce caractère doit être traité par le processus qui convertit les effets des différents plans du modèle de terminal.

#### A.3.9.6.1 *Inversion polarité*

Invert-Polarity ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 5/12, (9/12)] and (VIDÉO INVERSE) SD III [position dans le jeu C1: 4/8, (8/8)].
- *Remarque* – Etablissement d'un attribut d'inversion de polarité.

#### A.3.9.6.2 *Polarité normale*

Normal-Polarity ::= SID

- Fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 5/13, (9/13)] et (VIDÉO NORMALE) SD III [position dans le jeu C1: 4/9, (8/9)].
- *Remarque* – Etablissement d'un attribut de polarité normale.

#### A.3.9.7 *Commande de fenêtre/cases*

La possibilité fenêtre/cases établit une couleur spéciale de fond pour une cellule de caractère transparente à une image vidéo qui peut être sous-jacente à l'affichage. Cette possibilité est fournie directement par deux commandes dans l'une des syntaxes de données de terminal. La même possibilité est fournie, d'une manière plus complexe, dans toutes les syntaxes de données, moyennant l'établissement d'une couleur transparente spéciale qui peut être utilisée avec d'autres commandes de présentation.

Window/Box-Control ::= INTEGER { start-box (1),  
end-box (2) }

- Début de cases est une fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/10, (8/10)].
- *Remarque* – Etablissement de l'attribut de formation de cases.
- Fin de cases est une fonction prise de la Recommandation T.101 SD II [position dans le jeu C1: 4/11, (8/11)].
- *Remarque* – Arrêt de l'application de l'attribut de formation de cases.

#### A.3.9.8 *Commande de marquage*

La possibilité de commande de marquage a pour effet de marquer les positions de cellule de caractère pour les opérations futures. Cette fonction dépend de l'existence d'une mémoire axée sur les cellules de caractère dans le modèle de terminal. Elle ne peut pas être convertie à d'autres syntaxes de données.

Marking-Control ::= INTEGER { marked-mode-start (1),  
marked-mode-stop (2) }

- Début de mode marqué est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/0 5/3, CSI 3/1 5/3 ou CSI 3/2 5/3).
- *Remarque* – Application de l'attribut de marquage.
- Fin de mode marqué est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/0 5/4, CSI 3/1 5/4 ou CSI 3/2 5/4).
- *Remarque* – Fin de l'application de l'attribut de marquage.

### A.3.9.9 *Commande de protection*

Il y a de grandes différences dans la manière dont la commande d'entrée sélective est traitée dans les trois syntaxes de données de terminal source. Non seulement les procédures diffèrent, mais les processus d'entrée sont bornés par des conditions aux limites différentes. Par exemple, dans un certain cas, l'entrée est associée à la mémoire de cellules de caractère du modèle de terminal multiplan; dans un autre cas, les données d'entrée sont bornées par une limite de capacité de mémoire (nombre et taille cumulative des champs de données d'entrée). Comme ces processus d'entrée sont fondamentalement différents, les commandes qui agissent sur eux sont indiquées ici séparément. De cette façon, le processus de conversion pourra simuler un ensemble de fonctions pour un terminal différent.

```
Protection-Control ::= INTEGER { unprotect-field      (1),
                                protect-field       (2),
                                protect-mode-start  (3),
                                protect-mode-cancel (4),
                                protect-mode-idle   (5),
                                unprotect-block     (6),
                                protect-block       (7) }
```

- Ne pas protéger champ est une fonction prise de la Recommandation T.101 SD III [position dans le jeu C1: 5/15, (9/15)].
- *Remarque* – Ne pas protéger une plage donnée de l'écran d'affichage, définie par la commande géométrique CHAMP, pour permettre l'entrée des caractères dans le tampon du champ non protégé quand le curseur se trouve dans la plage non protégée.
- Protéger champ est une fonction prise de la Recommandation T.101 SD III [position dans le jeu C1: 5/0, (9/0)].
- *Remarque* – Protéger une plage donnée de l'écran d'affichage pour empêcher l'entrée des caractères dans le tampon du champ non protégé quand le curseur se trouve dans la plage protégée. La totalité de l'écran est protégée par défaut.
- Début de mode de protection est une fonction prise de la Recommandation T.101, SD II (position dans le jeu C1: CSI 3/0 5/0, CSI 3/1 5/0 ou C SI 3/2 5/0).
- *Remarque* – Appliquer l'attribut protégé aux positions de cellule de caractère qui empêchent la surimpression.
- Suppression mode de protection est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/0 5/1, CSI 3/1 5/1 ou CSI 3/2 5/1).
- *Remarque* – Supprimer l'attribut protégé des positions de cellule de caractère qui permettent la surimpression.
- Mode de protection neutralisé est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/2 5/2).
- *Remarque* – Arrêt de l'application de l'attribut mode de protection.
- Ne pas protéger bloc est une fonction prise de la Recommandation T.101 SD I [position dans le jeu C1: 5/14, (9/14)].
- *Remarque* – Supprimer la protection des positions de cellule de caractère contre l'altération.
- Protéger bloc est une fonction prise de la Recommandation T.101 SD I [position dans le jeu C1: 5/15, (9/15)].
- *Remarque* – Protéger les positions de cellule de caractère contre l'altération.

### A.3.9.10 *Commande d'affichage*

La sous-catégorie commande d'affichage contient des fonctions qui agissent sur la façon dont le dispositif d'affichage présente l'information. Elle comprend les éléments suivants: configuration de la mémoire d'affichage disponible dans un modèle de terminal donné; s'il faut faire défiler le contenu de cette mémoire d'affichage; et la surimpression de l'information contenue dans cette mémoire.

```
Display-Control ::= CHOICE { [1] Plane-Configuration-Control,
                              [2] Scroll-Control,
                              [3] Overwrite-Mode }
```

Les modèles de terminal utilisés dans chacune des trois syntaxes de données de terminal diffèrent notablement d'une syntaxe à une autre. Dans deux des cas, les structures de modèle de terminal sont fixes. Dans le cas de la syntaxe SD I, la structure peut être modifiée dynamiquement. On peut faire varier la quantité de mémoire assignée à chaque plan d'affichage, ainsi que l'ordre de présentation (superposition) des plans. Ces fonctions dépendent étroitement du matériel d'affichage utilisé pour réaliser le modèle d'affichage particulier qui est sous-jacent à la syntaxe de données SD I; par ailleurs, les effets dynamiques que l'on peut obtenir avec ces fonctions ne sont pas convertibles dans les autres syntaxes de données. Cependant, le processus de conversion doit interpréter ces commandes pour déterminer les critères de tri d'autres informations d'affichage, afin de mettre en correspondance le modèle de terminal multiplan de SD I avec celui de SD II ou avec celui de SD III.

#### A.3.9.10.1 *Commande de configuration de plan*

```
Plane-Configuration-Control ::= CHOICE { [1] Frame-Area,
                                           [2] Set-Frame,
                                           [3] Assign-Frame,
                                           [4] Header-Area,
                                           [5] Body-Area }
```

##### A.3.9.10.1.1 *Plage de trame*

```
Frame-Area ::= SEQUENCE { Area-Origin, Area-Dimensions }
```

-- La fonction plage de trame est prise de la Recommandation T.101 SD I [position dans l'ensemble des commandes d'affichage: 2/5, (10/5)].

-- *Remarque* – L'ensemble G des commandes d'affichage a comme caractère final 3/8 dans SD I.

```
Area-Origin ::= SEQUENCE { REAL, REAL }
```

-- Spécification de l'origine de la plage de trame.

```
Area-Dimensions ::= SEQUENCE { REAL, REAL }
```

-- Spécification des dimensions de la plage de trame.

-- Les coordonnées sont spécifiées sous forme de fractions normées de la plage d'écran unitaire représentée dans un champ de nombre entier avec signe, avec un point binaire implicite dans la position de plus fort poids.

##### A.3.9.10.1.2 *Fixer trame*

```
Set-Frame ::= SEQUENCE OF { Set-Frame-Index,
                             Set-Frame-Memory-Assignment }
```

-- La fonction plage de trame est prise de la Recommandation T.101 SD I [position dans l'ensemble des commandes d'affichage: 2/6, (10/6)].

```
Set-Frame-Index ::= INTEGER
```

-- Index de plage de trame.

```
Set-Frame-Dimensions ::= INTEGER
```

-- Nombre de bits de mémoire de trame de fond attribués à la trame.

##### A.3.9.10.1.3 *Assigner trame*

```
Assign-Frame ::= INTEGER
```

-- Fonction prise de la Recommandation T.101 SD I [position dans l'ensemble des commandes d'affichage: 2/7, (10/7)].

#### A.3.9.10.1.4 *Plage d'en-tête*

Certaines syntaxes de données vidéotex axées sur les terminaux offrent la possibilité de présenter de l'information dans une plage de message spéciale, en plus de la plage d'affichage principale. Cette plage de message contiendrait des messages relatifs au service. Il n'est pas douteux que le contenu de ces messages, en interfonctionnement international, varierait entre les systèmes vidéotex. La syntaxe de données SD I fournit des commandes spéciales qui agissent sur cet en-tête de message. Dans SD I, les commandes de trame de fond et de trame de fond d'en-tête agissent sur l'affichage de l'information de présentation dans la plage d'affichage principale ou la plage de message d'en-tête. Les commandes de trame de fond établissent également les valeurs de couleurs initiales dans la syntaxe de données SD I. Ces commandes sont mentionnées ici pour permettre d'identifier et de convertir comme il convient l'information d'en-tête.

Header-Area ::= SEQUENCE { Raster-Colour-Value }

-- Fonction prise de la Recommandation T.101 SD I [position dans l'ensemble des commandes d'affichage: 3/9, (11/9)].

-- *Remarque* – L'ensemble G des commandes d'affichage a comme caractère 3/8 dans SD I.

Raster-Colour-Values ::= SEQUENCE { INTEGER, INTEGER, INTEGER }

-- Spécification de la couleur initiale de l'en-tête de trame de fond, respectivement pour vert, rouge et bleu.

-- Les valeurs de couleurs sont spécifiées sous forme de fractions normées de la gamme unitaire des couleurs, représentée dans un champ d'entier avec signe, avec un point binaire implicite dans la position de plus fort poids.

#### A.3.9.10.1.5 *Plage de corps*

Body-Area ::= SEQUENCE { Body-Opcode, Raster-Colour-Values }

-- Fonction prise de la Recommandation T.101 SD I [position dans l'ensemble des commandes d'affichage: 3/8, (11/8)].

#### A.3.9.10.2 *Commande de défilement*

Le défilement peut s'effectuer sur la totalité ou sur une partie de l'écran. Il y a une grande différence entre le défilement sur un modèle de terminal multiplan et le défilement sur un modèle de terminal à un seul plan. L'assignation des fonctions aux divers plans, dans un modèle de terminal multiplan, influe aussi énormément sur le résultat du défilement. Dans certains cas, l'information graphique sous-jacente se déplace avec les caractères en défilement; dans d'autres cas, elle reste immobile. Dans SD I, la possibilité de déplacement multiplan permet d'avoir des mouvements dynamiques et des assignations aux plans qui influent profondément sur les caractéristiques du défilement. Il n'est pas possible, en général, de convertir toutes les opérations dynamiques telles que le défilement entre syntaxes de données de terminal; néanmoins, les résultats du défilement ont une influence sur la présentation finale. Le processus de conversion doit mettre les données en mémoire-tampon et leur appliquer un post-traitement afin que l'image finale soit correcte. Les opérations de défilement sont fondamentalement différentes entre les trois syntaxes de données de terminal; elles sont donc mentionnées ici afin de pouvoir être traitées par le processus de conversion.

Scroll-Control ::= CHOICE { scroll-on [1] NULL,  
scroll-off [2] NULL,  
scroll-up [3] NULL,  
scroll-down [4] NULL,  
activate-implicit-scrolling [5] NULL,  
deactivate-implicit-scrolling [6] NULL,  
create-scroll-area [7] Create-Scroll-Area,  
delete-scroll-area [8] Delete-Scroll-Area,  
scroll-display-mode-on [9] NULL,  
scroll-display-mode-off [10] NULL }

-- Défilement en marche est une fonction prise de la Recommandation T.101 SD III [position dans le jeu C1: 5/7, (9/7)].

-- *Remarque* – Déclencher le défilement dans un seul plan, à l'intérieur d'un champ d'affichage actif.

- Arrêt de défilement est une fonction prise de la Recommandation T.101 SD III [position dans le jeu C1: 5/8, (9/8)].
- *Remarque* – Neutraliser le défilement dans un seul plan.
- Défilement vers le haut est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/0 6/0).
- *Remarque* – Faire défiler la plage de défilement vers le haut.
- Défilement vers le bas est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/1 6/0).
- *Remarque* – Faire défiler la plage de défilement vers le bas.
- Activer défilement implicite est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/2 6/0).
- *Remarque* – Appliquer un défilement implicite à la plage de défilement sur la limite de cette plage.
- Désactiver défilement implicite est une fonction prise de la Recommandation T.101 SD II (position dans le jeu C1: CSI 3/3 6/0).
- *Remarque* – Empêcher que la plage de défilement ne subisse un défilement ne subisse un défilement implicite.
- Mode affichage défilement présent est une fonction prise de la Recommandation T.101 SD I (position dans l'ensemble G des commandes d'affichage: 2/4, avec paramètre b6 = 1).
- *Remarque* – L'ensemble G des commandes d'affichage a comme caractère final 3/8 dans SD I.
- *Remarque* – Etablir l'attribut de défilement du mode d'affichage.
- Arrêt mode d'affichage défilement est une fonction prise de la Recommandation T.101 SD I (position dans l'ensemble G des commandes d'affichage: 2/4, avec paramètre b6 = 0).
- *Remarque* – Neutraliser l'attribut de défilement du mode d'affichage.

#### A.3.9.10.2.1 Créer plage de défilement

Create-Scroll-Area ::= SEQUENCE { Upper-Par, Lower-Par }

- Fonction prise de la Recommandation T.101 SD II (5/5).

-- *Remarque* – Créer une plage de défilement.

Upper-Par ::= SEQUENCE { INTEGER, INTEGER, INTEGER }

- Paramètres <URH> <URT> <URU> définissant la rangée limite supérieure de la plage de défilement.

Lower-Par ::= SEQUENCE { INTEGER, INTEGER, INTEGER }

- Paramètres <LRH> <LRT> <LRU> définissant la rangée limite inférieure de la plage de défilement.

#### A.3.9.10.2.2 Supprimer plage de défilement

Delete-Scroll-Area ::= SEQUENCE { Upper-Par, Lower-Par }

- Fonction prise de la Recommandation T.101 SD II (5/6).

-- *Remarque* – Supprimer une plage de défilement.

#### A.3.9.10.3 Mode surimpression

En liaison avec la commande de la configuration de mémoire du modèle de terminal, une des syntaxes de données de terminal offre une possibilité spécifique de commander le mode d'accumulation des données dans un plan d'affichage particulier. Avec la syntaxe de données SD I, il est possible de faire en sorte que la surimpression de la mémoire dépende du contenu actuel de la mémoire. Les nouvelles données peuvent soit remplacer l'ancien contenu de la mémoire, soit appliquer une opération «OU» logique, «ET» logique ou «XOU» (exclusif OU) logique à l'ancien contenu de la mémoire avant de la remplacer. Il est extrêmement difficile de simuler cette fonction dans les deux autres syntaxes

de données, dans le cas général, car elle nécessite des opérations au niveau des bits dans une mémoire dépendant d'un modèle de terminal particulier. Elle est mentionnée ici afin que le processus de conversion puisse effectuer la meilleure simulation possible.

Overwrite-Mode ::= SEQUENCE { Overwrite-Par }

-- Fonction de la Recommandation T.101 SD I (position dans l'ensemble G des commandes d'affichage: 2/4).

-- *Remarque* – L'ensemble G des commandes d'affichage a comme caractère final 3/8 dans SD I.

Overwrite-Par ::= INTEGER { replace (1),  
or (2),  
and (3),  
xor (4) }

#### A.3.9.11 *Commande de dispositif*

A l'exception des commandes dispositif d'affichage en marche et arrêt dispositif d'affichage, les commandes de dispositif autres que les fonctions d'affichage de présentation sortant du cadre des syntaxes de données d'interfonctionnement.

Device-Control ::= INTEGER { display-device-on (1),  
display-device-off (2) }

-- Dispositif d'affichage en marche est une fonction prise de la Recommandation T.101 SD II (séquence de commande ESC 3/12).

-- Arrêt dispositif d'affichage est une fonction prise de la Recommandation T.101 SD II (séquence de commande ESC 3/13).

#### A.3.9.12 *Commande de curseur*

Le curseur d'affichage est soumis à une commande explicite dans chacune des syntaxes de données de terminal, ainsi qu'à une commande implicite. Par ailleurs, les commandes explicites ne sont pas codées de la même façon dans les diverses syntaxes. Dans le cas de la commande implicite, le curseur d'affichage est commandé par la commande de mode de protection du champ non protégé, dans la syntaxe de données de terminal SD III. Une conversion est nécessaire entre ces diverses fonctions de commande.

Cursor-Control ::= CHOICE { Cursor-On (1),  
Cursor-Flash (2),  
Cursor-Off (3) }

##### A.3.9.12.1 *Curseur en fonction*

Cursor-On ::= SID

-- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C0: 1/1), SD I [position dans le jeu C1: 4/14, (8/14)] et (CURSOR STABLE) de SD III [position dans le jeu C1: 5/12, (9/12)].

##### A.3.9.12.2 *Clignotement curseur*

Cursor-Flash ::= SID

-- Fonction prise de la Recommandation T.101 SD III [position dans le C1: 5/11, (9/11)].

##### A.3.9.12.3 *Curseur arrêté*

Cursor-Off ::= SID

-- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C0: 1/14), SD I [position dans le jeu C1: 4/15, (8/15)] et (CURSEUR ARRÊTÉ) de SD III [position dans le jeu C1: 5/13, (9/13)].

### A.3.9.13 Commande de réinitialisation

Chacune des syntaxes de données de source en vidéotex offre la possibilité de réinitialiser les états d'affichage faisant partie d'une syntaxe donnée, et de les ramener à un ensemble prédéfini de valeurs. Les paramètres susceptibles d'être modifiés par les diverses fonctions de réinitialisation des différentes syntaxes de données de source sont très variés. Certaines de ces fonctions offrent la possibilité de réinitialiser sélectivement des paramètres particuliers, d'autres appliquent la réinitialisation à une liste prédéfinie de paramètres qui dépend de la syntaxe de données. La syntaxe de données d'interfonctionnement doit mettre en oeuvre les fonctions de réinitialisation de deux manières différentes. D'une part, une indication de la commande de réinitialisation considérée doit être communiquée en tant qu'élément syntaxique dans la SDI. Les diverses fonctions de réinitialisation sont indiquées ici afin que l'effet de présentation d'une telle fonction puisse être produit dans la conversion. D'autre part, une fonction de réinitialisation agit fortement sur les états globaux de présentation. Ces états sont repérés dans le processus de conversion, de telle manière que ce processus n'ait pas besoin de comprendre les relations mutuelles entre les commandes de présentation. Cela signifie que le processus de conversion n'a pas besoin de simuler un terminal de la syntaxe de données de source pour effectuer la conversion de ses éléments. C'est pourquoi, à côté de la commande de réinitialisation de la SDI, il faut introduire un type particulier de vecteur d'état qui rétablit les variables globales.

```
Reset-Control ::= CHOICE { [1] Reset-Type-I,  
                           [2] Reset-Type-II,  
                           [3] Reset-Type-III }
```

#### A.3.9.13.1 Réinitialisation type I

```
Reset-Type-I ::= SEQUENCE { P-Reset-Par OPTIONAL }
```

- Fonction prise de la Recommandation T.101 SD I [position dans l'ensemble G des commandes d'affichage: 2/1, (10/1)].
- *Remarque* – L'ensemble G des commandes d'affichage a comme caractère final 3/8 dans SD I.

##### A.3.9.13.1.1 Par réinitialisation P

```
P-Reset-Par ::= SEQUENCE { macro-reset BOOLEAN,  
                           blink-reset BOOLEAN,  
                           lut-reset BOOLEAN,  
                           screen-reset BOOLEAN }
```

- Réinitialisation sélective des paramètres identifiés.
- *Remarque* – La syntaxe de données SD I comprend aussi la fonction de réinitialisation NSR qui est identifiée séparément plus haut.

#### A.3.9.13.2 Réinitialisation type II

```
Reset-Type-II ::= SEQUENCE { US-Reset-Operation,  
                             US-Reset-Parameter }
```

- Fonction prise de la Recommandation T.101 SD II (position dans le jeu C0: 1/15) suivi du caractère fixe 2/15.

##### A.3.9.13.2.1 Opération de réinitialisation US

```
US-Reset-Operation ::= CHOICE { us-reset-mosaic-1      [1] NULL,  
                                us-reset-mosaic-2      [2] NULL,  
                                us-reset-mosaic-1-limited [3] NULL,  
                                us-reset-mosaic-2-limited [4] NULL,  
                                us-reset-service-break   [5] US-Reset-Service-Break,  
                                us-reset-to-previous-state [6] NULL }
```

- Réinitialisation US mosaïque 1 est représenté par le caractère d'identification de réinitialisation US (4/1), réinitialisation aux valeurs par défaut et appel du jeu C1 en mode série.

- Réinitialisation US mosaïque 2 est représenté par le caractère d'identification de réinitialisation US (4/2), réinitialisation aux valeurs par défaut et appel du jeu C1 en mode parallèle.
- Réinitialisation US mosaïque 1 limité est représenté par le caractère d'identification de réinitialisation US (4/3), réinitialisation aux valeurs par défaut limitées et appel du jeu C1 en mode série.
- Réinitialisation US mosaïque 2 limité est représenté par le caractère d'identification de réinitialisation US (4/4), réinitialisation aux valeurs par défaut limitées et appel du jeu C1 en mode parallèle.
- Réinitialisation US à l'état précédent est représenté par le caractère d'identification de réinitialisation US (4/15), réinitialisation à l'état après une réinitialisation à l'interruption du service.

#### A.3.9.13.2 Réinitialisation US après interruption du service

US-Reset-Service-Break ::= SEQUENCE { INTEGER { break-to-row-serial (1),  
break-to-row-parallel (2) }, row-designator }

- Interruption en mode série rangée est représenté par le caractère US réinitialisation identificateur (4/0), le service s'interrompt en mode C1 série par rangée.
- Interruption en mode parallèle rangée est représenté par le caractère US réinitialisation identificateur (4/5), le service s'interrompt en mode C1 parralèle par rangée.
- Désignation de rangée est représenté par le caractère US réinitialisation du paramètre de désignation de rangée, où la rangée désignée est codée de la colonne 4 à la colonne 7 du tableau de code. Le numéro de rangée est indiqué par la valeur binaire des 6 bits de plus faible poids.

#### A.3.9.13.3 Réinitialisation type III

Reset-Type-III ::= SEQUENCE { [1] Reset-Par1 OPTIONAL,  
[2] Reset-Par2 OPTIONAL }

- Fonction prise de la Recommandation T.101 SD III [position dans le jeu G PDI: G 2/4, (10/0)].

Reset-Par1 ::= SEQUENCE { INTEGER {  
colour-mode-1 (1),  
colour-mode-2 (2),  
colour-mode-3 (3), }  
INTEGER {  
display-to-nominal-black (1),  
display-to-current-colour (2),  
border-to-nominal-black (3),  
border-to-current-colour (4),  
display-and-border-to-current-colour (5),  
display-to-current-colour-and-border-to-nominal-black (6),  
display-and-border-to-nominal-black (7) },  
domain BOOLEAN }

Reset-Par2 ::= SEQUENCE {  
drcs-reset BOOLEAN,  
macro-pdi-reset BOOLEAN,  
texture-reset BOOLEAN,  
unprotected-field-reset BOOLEAN,  
blink-pdi-reset BOOLEAN,  
text-pdi-reset BOOLEAN }

- Réinitialiser sélectivement les paramètres identifiés.
- *Remarque* – La syntaxe de données SD III comprend également la fonction de réinitialisation NSR qui est identifiée séparément plus haut.

### A.3.10 Chaîne géométrique

Toutes les syntaxes de données de terminal source offrent une possibilité géométrique, mais les possibilités disponibles dans les divers systèmes géométriques sont très différentes. La syntaxe de données d'interfonctionnement regroupe les commandes géométriques courantes. La Recommandation F.300 identifie les diverses catégories dans lesquelles les fonctions géométriques peuvent être rangées. Ces catégories sont utilisées ci-après. La SDI utilise des coordonnées normées pour toutes les commandes géométriques. Elle utilise aussi des spécifications de coordonnées relatives pour toutes les listes de coordonnées, sauf pour les commandes déterminer position et point de marquage, qui sont absolues. Cependant, dans certains cas, il est possible de choisir entre les coordonnées absolues et les coordonnées relatives. Toutes les autres formes de coordonnées utilisées dans l'une quelconque des syntaxes de données axées sur les terminaux, par exemple l'emploi général de coordonnées absolues ou incrémentielles, seront converties dans les formes indiquées plus haut.

Geometric-String ::= CHOICE { [1] Geometric-Drawing-Command,  
[2] Geometric-Control-Command }

#### A.3.10.1 Commande de dessin géométrique

Geometric-Drawing-Command ::= CHOICE { [1] Marker-Point,  
[2] Line,  
[3] Arc-Circle,  
[4] Rectangle,  
[5] Polygon,  
[6] Spline,  
[7] Pixel-Array }

Certaines des syntaxes de données de terminal source fournissent une méthode pour passer facultativement d'une primitive à une autre de manière relative. On dispose ainsi d'un niveau d'efficacité dans certaines situations, mais la multiplicité des formats équivalents rendrait plus complexe la syntaxe de données d'interfonctionnement. C'est pourquoi cette syntaxe exige la spécification de la position initiale d'une commande de dessin, comme élément de la chaîne de paramètres pour chaque commande. Dans certains cas de conversion à partir de syntaxes de données permettant l'association relative des commandes, il faudra que le processus de conversion calcule la position actuelle effective au début d'une commande et qu'il inclue cette donnée comme élément de la chaîne de paramètres. Il n'existe pas, dans la SDI, d'équivalent direct de la commande déterminer position des syntaxes de données I et III. Cette information est transmise comme paramètre initial de chacune des autres commandes de dessin.

##### A.3.10.1.1 Point de marquage

Les diverses syntaxes de données de terminal diffèrent quant à leur possibilité de présenter une forme de marqueur en un point. Les syntaxes SD I et SD DIII permettent seulement de dessiner un point, alors que SD II permet aussi de dessiner une forme de marqueur en un point spécifique. Le processus de conversion est capable de simuler facilement la fonctionnalité de point de marquage, dans la conversion à la syntaxe de données SD I ou SD III, par l'emploi de plusieurs fonctions de présentation, éventuellement incluses dans une commande MACRO pour des raisons d'efficacité. La commande marqueur point ou marqueur forme est identifiée par le repère de contexte dans l'instruction CHOIX. La forme du marqueur de forme (marqueur) est définie par une commande géométrique.

Marker-Point ::= CHOICE { [1] Dot-Point,  
[2] Shape-Point }

Dot-Point ::= SEQUENCE OF { Abs-Coord }

-- Cette commande transmet les éléments fonctionnels de la commande DÉTERMINATION POINT des syntaxes de données I et III, et de la commande POLYMARQUEUR de la syntaxe de données II, le marqueur donnant la forme d'un point.

Shape-Point ::= SEQUENCE OF { Abs-Coord }

-- Cette commande transmet les éléments fonctionnels de la commande POLYMARQUEUR de la syntaxe de données II avec une forme de marqueur générale.

#### A.3.10.1.2 *Ligne*

Toutes les syntaxes de données de terminal offrent la possibilité de dessiner une ligne unique ou une série de lignes. Il y a de petites différences quant à la manière dont sont traitées les conditions aux limites; toutefois, une conversion directe est généralement possible.

Line ::= SEQUENCE OF { Abs-Coord, SEQUENCE OF { Rel-Coord } }

-- Cette commande transmet les éléments fonctionnels de la commande LIGNE des syntaxes de données I et III, et de la commande LIGNE BRISÉE de la syntaxe de données II.

#### A.3.10.1.3 *Arc-cercle*

La possibilité de dessiner un arc ou un cercle diffère quelque peu entre les diverses syntaxes de données. Dans chacune de ces syntaxes, la fonction cercle-arc a été optimisée au point de permettre une communication efficace de l'information d'arc ou de cercle dans le contexte de la syntaxe de données. La syntaxe de données d'interfonctionnement a moins à voir avec l'efficacité qu'avec la communication d'une information suffisante pour permettre la conversion. C'est la raison pour laquelle la SDI n'est pas concernée par les différentes manières de communiquer les mêmes paramètres; toutefois, la SDI contiendra toutes les fonctions disponibles dans la possibilité cercle-arc dans les diverses syntaxes de données.

Arc-Circle ::= CHOICE { [1] Circle,  
[2] Arc-3-Point,  
[3] Arc-3-Point-Chord,  
[4] Arc-3-Point-Pie,  
[5] Ellipse,  
[6] Elliptic-Arc,  
[7] Elliptic-Arc-Chord,  
[8] Elliptic-Arc-Pie,  
[9] Arc-Centre-Cord,  
[10] Arc-Centre-Pie }

##### A.3.10.1.3.1 *Cercle*

Circle ::= SEQUENCE { Abs-Coord, Coord }

-- Cette commande transmet les éléments fonctionnels de la commande SD I, et III ARC (forme cercle) et de la commande SD II GDP (cercle).

-- La coordonnée absolue définit la position initiale du cercle. L'autre coordonnée définit le diamètre du cercle en spécifiant un point sur le côté opposé.

##### A.3.10.1.3.2 *Arc 3 points*

Arc-3-Point ::= SEQUENCE { Abs-Coord, Coord, Coord }

-- Cette commande transmet les éléments fonctionnels de la commande SD I, et III ARC (forme contour) et de la commande SD II GDP (arc de cercle 3 points).

-- La coordonnée absolue définit la position initiale de l'arc. Les deux autres coordonnées définissent respectivement un point sur l'arc et la position finale de l'arc.

##### A.3.10.1.3.3 *Segment circulaire 3 points*

Arc-3-Point-Chord ::= SEQUENCE { Abs-Coord, Coord, Coord }

-- Cette commande transmet les éléments fonctionnels de la commande SD 1, et III ARC (segment circulaire rempli) et de la commande SD II GDP (segment circulaire 3 points).

-- La coordonnée absolue définit la position initiale de l'arc. Les deux autres coordonnées définissent respectivement un point sur l'arc et la position finale de l'arc. Une corde est tracée de la position initiale à la position finale de l'arc.

#### A.3.10.1.3.4 *Secteur circulaire 3 points*

Arc-3-Point-Pie ::= SEQUENCE { Abs-Coord, Coord, Coord }

- Cette commande transmet les éléments fonctionnels de la commande SD II GDP (secteur circulaire 3 points).
- La coordonnée absolue définit la position initiale de l'arc. Les deux autres coordonnées définissent respectivement un point sur l'arc et la position finale de l'arc. Deux lignes sont tracées, de la position initiale au centre géométrique de l'arc puis à la position finale de l'arc, pour former un secteur. Un secteur rempli n'est pas directement disponible dans la syntaxe de données SD I ou SD III, mais le processus de conversion peut simuler la fonction en utilisant un arc et deux lignes.

#### A.3.10.1.3.5 *Ellipse*

Ellipse ::= SEQUENCE { Abs-Coord, Coord, Coord, Coord }

- Cette commande transmet les éléments fonctionnels de la commande SD II GDP (ellipse).
- La coordonnée absolue définit la position initiale de l'ellipse. Un deuxième paramètre de coordonnée définit un point du côté opposé de l'arc, qui établit le diamètre du grand axe. Les troisième et quatrième paramètres définissent le diamètre du petit axe. Une ellipse ou un arc elliptique ne sont pas directement disponibles dans les syntaxes de données SD I ou SD III, mais le processus de conversion peut simuler la fonction par tronçons successifs ou en ajustant une courbe de lissage.

#### A.3.10.1.3.6 *Arc elliptique*

Elliptic-Arc ::= SEQUENCE { Abs-Coord, Coord, Coord, Coord }

- Cette commande transmet les éléments fonctionnels de la commande SD II GDP (arc elliptique).
- La coordonnée absolue définit la position initiale de l'arc. Un deuxième paramètre de coordonnée définit un point du côté opposé de l'arc, qui établit le diamètre du grand axe. Un troisième paramètre définit le diamètre du petit axe. Un quatrième paramètre définit la position finale de l'arc.

#### A.3.10.1.3.7 *Segment elliptique*

Elliptic-Arc-Chord ::= SEQUENCE { Abs-Coord, Coord, Coord, Coord }

- Cette commande transmet les éléments fonctionnels de la commande SD II GDP (segment elliptique).
- La coordonnée absolue définit la position initiale de l'arc. Un deuxième paramètre de coordonnée définit un point du côté opposé de l'arc, qui établit le diamètre du grand axe. Un troisième paramètre définit le diamètre du petit axe. Un quatrième paramètre définit la position finale de l'arc. Une corde est tracée de la position initiale à la position finale de l'arc.

#### A.3.10.1.3.8 *Secteur elliptique*

Elliptic-Arc-Pie ::= SEQUENCE { Abs-Coord, Coord, Coord, Coord }

- Cette commande transmet la fonctionnalité de la commande SD II GDP (secteur elliptique).
- La coordonnée absolue définit la position initiale de l'arc. Un deuxième paramètre de coordonnée définit un point du côté opposé de l'arc, qui établit le diamètre du grand axe. Un troisième paramètre de coordonnée définit le diamètre du petit axe. Un quatrième paramètre définit la position finale de l'arc. Deux lignes sont tracées, du point initial au centre géométrique de l'arc puis à la position finale, pour former un secteur.

#### A.3.10.1.3.9 *Coord centre arc*

Arc-Centre-Chord ::= SEQUENCE { Abs-Coord, Coord, Coord }

- Cette commande transmet les éléments fonctionnels de la commande SD II GDP (coord centre arc).

- La coordonnée absolue définit le centre de l'arc. Les autres paramètres de coordonnée définissent les points de départ et d'arrivée de l'arc.

#### A.3.10.1.3.10 *Secteur circulaire central*

Arc-Centre-Pie ::= SEQUENCE { Abs-Coord, Coord, Coord }

- Cette commande transmet la fonctionnalité de la commande SD II GDP (secteur circulaire central).
- La coordonnée absolue définit le centre de l'arc. Les autres paramètres de coordonnée définissent les points de départ et d'arrivée de l'arc.

#### A.3.10.1.4 *Rectangle*

Rectangle ::= SEQUENCE { Abs-Coord, Rel-Coord }

- Cette commande transmet les éléments fonctionnels de la commande RECTANGLE de SD I et III, et de la commande GDP (rectangle) de SD II.
- La coordonnée absolue définit la position initiale du rectangle. Un paramètre de coordonnée relative définit un point sur le côté opposé du rectangle, qui établit la taille du rectangle.

#### A.3.10.1.5 *Polygone*

Polygon ::= SEQUENCE { Abs-Coord, SEQUENCE OF { Rel-Coord } }

- Cette commande transmet les éléments fonctionnels de la commande SD I et III POLYgone (rempli) et de la commande SD II ZONE REMPLIE. Les syntaxes de données I et III fournissent aussi une commande POLYgone (périmètre) qui peut être transmise à travers la SDI par une commande LIGNE avec une répétition du point initial comme point final.
- La coordonnée absolue définit la position initiale du polygone. La séquence de coordonnées relatives définit les sommets du polygone. Un polygone est toujours fermé et la position finale est la même que la position initiale.

#### A.3.10.1.6 *Courbe de lissage*

Spline ::= SEQUENCE { Abs-Coord, SEQUENCE OF { Rel-Coord } }

- Cette commande transmet les éléments fonctionnels de la commande SD I et III ARC (courbe de lissage) et de la commande SD II GDP (courbe de lissage).
- La coordonnée absolue définit la position de la polycourbe. La séquence de coordonnées relatives (supérieures à 3) définit la courbe.
- *Remarque* – Les diverses syntaxes de données de terminal n'utilisent pas exactement la même définition pour le type et/ou les paramètres de la fonction génératrice de la courbe de lissage; toutefois, toutes les syntaxes de données de terminal source ont tendance à utiliser une courbe de lissage d'un certain type. Cela est susceptible de provoquer des différences notables dans l'image obtenue après conversion, mais c'est le résultat le plus exact qu'on puisse obtenir d'une manière raisonnable.

#### A.3.10.1.7 *Matrice de pixels*

Pixel-Array ::= SEQUENCE {  
     first-point Abs-Coord,  
     second-point Abs-Coord,  
     third-point Rel-Coord,

- Ces trois points définissent l'aire du pixel qui pourrait être en général un parallélogramme. Les deux premiers points sont les points extrêmes d'une diagonale.

    cells-first-direction INTEGER,  
     cells-second-direction INTEGER,

- Ces valeurs divisent l'aire du pixel selon un quadrillage à dimensions égales, pour représenter la résolution (logique) visée. La première direction est considérée du premier point vers le troisième. La deuxième direction va du premier point vers le point non spécifié. Ces valeurs sont faciles à déterminer, par exemple à partir du pel logique dans le cas POINT INCRÉMENTIEL.

Pixel-Array-Data }

Pixel-Array-Data ::= CHOICE { [1] IMPLICIT SEQUENCE OF Basic-Colour-Selection,  
[2] IMPLICIT SEQUENCE OF Direct-Colour-Selection,  
[3] IMPLICIT SEQUENCE OF Indexed-Colour-Selection }

- La liste de couleurs est définie d'après la `chaîne de commande de couleur`. La sélection d'une couleur auxiliaire n'est pas significative pour cette définition. La première couleur est mise en correspondance avec la cellule associée au premier point. Les éléments de couleur sont mis en correspondance à l'intérieur des rangées allant du premier au troisième point, et avec des rangées dont l'ordre est croissant à partir du troisième point jusqu'au deuxième point.

Les diverses syntaxes de données vidéotex de terminal source contiennent des commandes pour un codage incrémentiel efficace des lignes et des polygones, afin d'accroître l'efficacité. La possibilité d'incrémentiation diffère considérablement entre les différentes syntaxes de données; il n'est pas possible de réaliser un format intermédiaire approprié pour tous les environnements différents. Etant donné que l'efficacité est d'importance secondaire, les lignes et polygones incrémentiels doivent être communiqués sur la base des fonctions ligne et polygone ci-dessus.

#### A.3.10.2 Commandes géométriques

De nombreuses commandes sont disponibles dans chacune des syntaxes de données de terminal pour commander les fonctions de dessin géométrique. Beaucoup des commandes géométriques définies dans chacune des syntaxes de données peuvent sembler identiques, mais elles diffèrent quant à leur effet secondaire. C'est la raison pour laquelle on mentionne ici toutes ces commandes géométriques. C'est seulement dans le cas où les commandes sont identiques, par exemple, pour un certain nombre de commandes des syntaxes de données SD I et SD III, que l'on donne ci-après une définition commune.

Geometric-Control-Command ::= CHOICE { [1] Geo-Control-Command-1,  
[2] Geo-Control-Command-2 }

- Deux types de commandes géométriques figurent dans la SDI, afin de tenir compte des deux approches différentes adoptées dans la syntaxe de données II et dans la syntaxe de données I, III. Ces commandes sont groupées séparément car elles ne seraient jamais reçues ensemble.

##### A.3.10.2.1 Commande géo 1

Geometric-Control-Command-1 ::= CHOICE { [1] Numeric-Precision,  
[2] Drawing-Point-Size,  
[3] Line-Style,  
[4] Highlight,  
[5] Fill,  
[6] Field,  
[7] Blink-Process,  
[8] Wait }

- Commandes géométriques analogues à celles des syntaxes de données I et III.

##### A.3.10.2.1.1 Précision numérique

Numeric-Precision ::= SEQUENCE { REAL, REAL }

- Cette commande transmet les éléments fonctionnels de la commande géométrique DOMAINE des syntaxes de données I et III.

- Définition de la précision numérique nominale appliquée par la syntaxe de données de source. Comme les règles de codage ASN.1 autorisent la communication de n'importe quelle précision de données, cette commande n'affecte pas la précision des données communiquées. Elle est utilisée pour informer le processus de conversion de la précision nominale appliquée par la syntaxe de données de source. Le premier paramètre transmet la précision, exprimée par un nombre de bits significatifs, pour des opérandes à valeur unique. De la même manière, le deuxième paramètre transmet le nombre de bits significatifs pour des opérandes (2<sup>e</sup> et 3<sup>e</sup>) à valeurs multiples.

#### A.3.10.2.1.2 *Taille du point de dessin*

Drawing-Point-Size ::= Rel-Coord

- Cette commande transmet les éléments fonctionnels de la commande géométrique DOMAINE (taille du pel logique) des syntaxes de données I et III.
- Cette fonction de commande géométrique fixe la taille du point de dessin logique (PEL LOGIQUE), qui est une fraction des dimensions de l'écran unitaire. Le cas particulier zéro est interprété comme représentant la plus petite taille possible sur un dispositif de présentation.

#### A.3.10.2.1.3 *Style de ligne*

Line-Style ::= INTEGER { solid (1),  
dotted (2),  
dashed (3),  
dot-dashed (4) }

- Etablissement du style pour présenter des lignes à partir d'un ensemble fixe de styles de ligne.
- Cette commande transmet les éléments fonctionnels de la commande géométrique TEXTURE (texture de ligne) des syntaxes de données I et III.

#### A.3.10.2.1.4 *Surluminance*

Highlight ::= BOOLEAN

- Etablir si les zones remplies sont dessinées en mode surluminance, dans lequel le périmètre est dessiné en NOIR ou dans une couleur contrastant avec le remplissage.
- Cette commande transmet les éléments fonctionnels de la commande géométrique TEXTURE (surluminance) des syntaxes de données I et III.

#### A.3.10.2.1.5 *Remplissage*

Fill ::= BOOLEAN

- Déterminer si les polygones, les arcs fermés, les ellipses ou les rectangles doivent être remplis. Pour des raisons d'efficacité, cette commande est codée dans l'opcode qui identifie la primitive de dessin dans certaines syntaxes de données de terminal source. Cette fonction a été séparée ici, afin de faciliter la conversion entre les syntaxes de données.
- Cette commande transmet les éléments fonctionnels de la commande géométrique TEXTURE (modèle de texture de remplissage) des syntaxes de données I et III.

#### A.3.10.2.1.6 *Champ*

Field ::= Rel-Coord

- Définition des dimensions de la zone active sur l'écran d'affichage. La commande champ établit les limites qui «contiennent» le texte, c'est-à-dire les limites des plages de défilement, et dans lesquelles fonctionnent les caractères de mise en page. La position initiale est définie par la position de dessin géométrique actuelle. Les paramètres de coordonnées relatives définissent un point sur le côté opposé du champ qui fixe la taille de l'aire rectangulaire de champ.

- Cette commande transmet les éléments fonctionnels de la commande géométrique CHAMP des syntaxes de données I et III.

#### A.3.10.2.1.7 *Clignotement*

Blink-Process ::= SEQUENCE { [1] INTEGER,  
[2] INTEGER OPTIONAL,  
[3] INTEGER OPTIONAL,  
[4] INTEGER OPTIONAL }

- Etablissement d'un processus de clignotement dans lequel la carte des couleurs est modifiée dynamiquement pour un intervalle et une phase spécifiés. Le premier nombre entier représente l'adresse de la carte des couleurs de couleur travail, puis l'intervalle travail, l'intervalle repos et le retard phase, respectivement en 1/10 de seconde. La possibilité de traitement des processus de clignotement dépend étroitement du modèle de terminal. En général, les processus clignotement peuvent être utilisés pour simuler toute autre possibilité de clignotement disponible dans toute syntaxe de données, dans les limites de la mémoire disponible assignée à ces opérations, telle qu'elle est spécifiée par les conditions de valeurs limites. Toutefois, il n'est pas facile de simuler les processus Clignotement dans des conditions d'affichage dotées de possibilités insuffisantes.
- Cette commande transmet les éléments fonctionnels de la commande géométrique CLIGNOTEMENT des syntaxes de données I et III.

#### A.3.10.2.1.8 *Attente*

Wait ::= INTEGER

- Fixer une temporisation pour le traitement des données de présentation, pendant une durée exprimée en 1/10 de seconde. Bien qu'elle soit très simple, la commande attente pose de très grands problèmes de conversion. La raison en est qu'il s'agit d'une commande dynamique. La dynamique de présentation ne peut pas être garantie dans la conversion parce qu'il peut être nécessaire de modifier l'ordre de présentation des commandes pour tenir compte des différences entre les modèles de terminal des deux syntaxes de données. La conversion de la commande attente ne doit être tentée que si les processus de présentation de la source et de la cible sont synchronisés, c'est-à-dire si les commandes de présentation n'ont pas besoin d'être triées dans la conversion, ou à la fin d'une unité (page) de données.

#### A.3.10.2.2 *Commande géo 2*

Geo-Control-Command-2 ::= CHOICE { [1] Display-Element-Attributes,  
[2] Control-Element-Attributes }

- Commandes géométriques analogues à celles de la syntaxe de données II.
- Les attributs d'élément d'affichage relèvent des primitives d'affichage de sortie. Certaines de ces primitives peuvent être similaires à celles du paragraphe Commande géo 1, mais les effets secondaires sont différents dans ces commandes.
- Les attributs d'élément de commande établissent les fonctions de commande de transformation d'affichage, de découpage et de poste de travail, qui sont spécifiques aux conditions d'affichage associées à la syntaxe de données II.
- L'emploi de facilités groupées est pour études ultérieure.

### A.3.10.2.2.1 Attributs d'élément d'affichage

```

Display-Element-Attributes ::= CHOICE {
    [1] IMPLICIT Line-Attributes,
    [2] IMPLICIT Marker-Attributes,
    [3] IMPLICIT Fill-Area-Attributes }

Line-Attributes ::= SET {
    [1] IMPLICIT Line-Type OPTIONAL,
    [2] IMPLICIT Line-Width-Scale-Factor OPTIONAL,
    [3] IMPLICIT Polyline-Colour-Index OPTIONAL }

Line-Type ::= INTEGER {
    solid (0),
    dashed (1),
    dotted (2),
    dashed-dotted (3),
    implementation dependent (4) }

Line-Width-Scale-Factor ::= REAL
Polyline-Colour-Index ::= Colour-Index
Marker-Attributes ::= SET {
    [1] IMPLICIT Marker-Type OPTIONAL,
    [2] IMPLICIT Marker-Size-Scale-Factor OPTIONAL,
    [3] IMPLICIT Polymarker-Colour-Index OPTIONAL }

Marker-Type ::= INTEGER {
    dot (0),
    plus (1),
    asterisk (2),
    circle (3),
    diagonal-cross (4) }

Marker-Size-Scale-Factor ::= REAL
Polymarker-Colour-Index ::= Colour-Index
Fill-Area-Attributes ::= SET {
    [1] IMPLICIT Fill-Area-Interior-Style OPTIONAL,
    [2] IMPLICIT Fill-Area-Colour-Style OPTIONAL,
    [3] IMPLICIT Fill-Area-Style-Index OPTIONAL,
    [4] IMPLICIT Pattern-Reference-Point OPTIONAL,
    [5] IMPLICIT Pattern-Vectors OPTIONAL }

Fill-Area-Interior-Style ::= INTEGER {
    hollow (0),
    solid (1),
    pattern (2),
    hatch (3) }

Fill-Area-Colour-Index ::= Colour-Index
Fill-Area-Style-Index ::= INTEGER {
-- Pour le modèle de style intérieur, l'index de style choisit un modèle défini par «chaîne de commande de
  modèle de remplissage».
-- Pour les hachures de style intérieures, les styles suivants sont choisis:
    lignes verticales (0),
    lignes horizontales (1),
    lignes à pente de 45° (2),
    lignes à pente de 45° (3),
    lignes croisées verticales et horizontales (4),
    lignes croisées 45 et 45° (5) }

Pattern-Reference-Point ::= Abs-Coord
Pattern-Vectors ::= SEQUENCE { Abs-Coord, Abs-Coord }
-- L'origine de l'espace NDC et le premier point définissent le vecteur de hauteur de modèle. L'origine de
  l'espace NDC et le deuxième point définissent le vecteur de largeur de modèle.
Colour-Index ::= CHOICE {
    [1] IMPLICIT Basic-Colour-Selection,
    [2] IMPLICIT Indexed-Colour-Selection }

```

### A.3.10.2.2 Attributs d'élément de commande

```

Control-Element-Attributes ::= CHOICE {
    [1] WS-Management-Primitives,
    [2] Transformation-Primitives }

WS-Management-Primitives ::= CHOICE {
open-workstation           [1] IMPLICIT INTEGER,
                           --Identificateur de WS
close-workstation          [2] IMPLICIT INTEGER,
                           --Identificateur de WS
activate-workstation        [3] IMPLICIT INTEGER,
                           --Identificateur de WS
deactivate-workstation      [4] IMPLICIT INTEGER,
                           --Identificateur de WS
clear-workstation          [5] IMPLICIT INTEGER,
                           --Identificateur de WS
set-defaults               [6] IMPLICIT NULL,
update-workstation         [7] IMPLICIT Update-WS,
deferral-state             [8] IMPLICIT Deferral-State }

Update-WS
workstation-identfier      ::= SEQUENCE {
regeneration-flag          INTEGER,
                           INTEGER { perform (0),
                                     postpone (1) } }

Deferral-State
workstation-identfier      ::= SEQUENCE {
deferral-mode              INTEGER,
                           INTEGER { asap (0),
                                     bnil (1),
                                     bnig (2),
                                     asti (3) } }

implicit-regeneration      INTEGER { suppressed (0),
                                     allowed (1) } }

Transformation-Primitives ::= SET {
    [1] IMPLICIT WS-Window OPTIONAL,
    [2] IMPLICIT WS-Viewport OPTIONAL,
    [3] IMPLICIT Clipping-Rectangle OPTIONAL }

WS-Window
workstation-Identfier      ::= SEQUENCE {
first-point                INTEGER,
Abs-Coord,
second-point               Abs-Coord }

WS-Viewport
workstation-identfier      ::= SEQUENCE {
xmin                       INTEGER,
xmax                       REAL,
ymin                       REAL,
ymax                       REAL }

Clipping-Rectangle
first-point                ::= SEQUENCE {
second-point               Abs-Coord,
Abs-Coord }

```

### A.3.10.3 Coordonnées géométriques

Les données de coordonnées pour les opérations géométriques sont mémorisées sous forme de coordonnées d'affichage normées dans les trois syntaxes de données de source. Toutefois, les détails exacts du format de numéro diffèrent notablement entre les principes adoptés pour les syntaxes de données SD I et SD III et ceux adoptés pour SD II. Puisque le but de la SDI est l'interfonctionnement, il convient d'éviter des différences en ce qui concerne le format de numéro. C'est la raison pour laquelle on utilise dans la SDI un plan de numérotation simple, basé sur le type de données RÉEL avec signe de ASN.1. Les numéros RÉEL de ASN.1 sont autodélimitants et de longueur arbitraire; il n'y a donc aucune difficulté de précision et il est inutile d'assigner des champs de bits spéciaux pour déterminer la longueur du numéro. Une coordonnée peut donc être représentée par un couple de nombres. La transposition d'un champ de données

réelles sur un champ de données numériques, dans l'une quelconque des syntaxes de données, dépend de la syntaxe de données considérée. Dans le cas des syntaxes de données SD I et SD III, l'aire d'affichage d'unité normée est transposée sur la partie fractionnaire (mantisse) du champ de numéro réel. Pour SD II, on utilise la mantisse et l'exposant du nombre réel.

Etant donné que des spécifications de coordonnées tridimensionnelles sont disponibles dans toutes les syntaxes de données, on trouvera ci-après un triplet entier à titre optionnel. Comme l'opération tridimensionnelle est optionnelle, la projection sur deux dimensions doit être définie de telle façon que l'information tridimensionnelle puisse être vue dans un environnement à deux dimensions par le truchement de l'interfonctionnement. On a recours à une projection planaire avec l'hypothèse  $Z = 0$ .

```
Coord      ::= IMPLICIT CHOICE { Abs-Coord, Rel-Coord }
Abs-Coord  ::= CHOICE { [1] X-Y,
                       [2] X-Y-Z }
X-Y        ::= SEQUENCE { REAL, REAL }
-- Coordonnées X, Y absolues
X-Y-Z      ::= SEQUENCE { REAL, REAL, REAL }
-- Coordonnées X, Y, Z absolues
Rel-Coord  ::= CHOICE {
               [3] DX-DY,
               [4] DX-DY-DZ }
DX-DY      ::= SEQUENCE { REAL, REAL }
-- Coordonnées relatives DX, DY
DX-DY-DZ   ::= SEQUENCE { REAL, REAL }
-- Coordonnées relatives DX, DY, DZ
```

### A.3.11 Chaîne de commande d'animation

La possibilité de réaliser des effets dynamiques ou d'animation sur le dispositif de présentation dépend en grande partie du modèle de terminal et des conditions d'affichage. Plusieurs syntaxes de données de terminal fournissent quelques facilités spécialisées pour l'obtention d'effets dynamiques. Par exemple, les syntaxes de données SD I et II ont une facilité de clignotement triphasique et SD III a une fonction de clignotement en phase de la carte des couleurs. Les effets dynamiques produits par ces fonctions spéciales ne seront généralement pas conservés dans la conversion. Cela est d'autant plus vrai que l'ordre d'affichage des entités de présentation est susceptible d'être modifié par le processus de conversion pour tenir compte des différences existant dans le modèle de terminal. A l'exception du clignotement rapide, il faut que le processus de conversion tienne compte des effets dynamiques, même s'il n'est pas capable de les convertir fidèlement, car ces effets sont de nature à modifier notablement l'image finale résultante.

La syntaxe de données SD I offre une possibilité sophistiquée d'animation dépendant du modèle de terminal. Cette possibilité fait appel à un modèle de terminal multiplan dans lequel l'ordre et la position relative des divers plans sont variables. Les effets pouvant ainsi être produits sont spécifiques de l'environnement dans lequel ils ont été définis. Toutefois, les commandes d'animation de la syntaxe de données SD I doivent être incluses dans la syntaxe de données d'interfonctionnement, car elles influencent le résultat final de l'affichage. Le processus de conversion doit produire l'image résultante finale correcte.

```
Animation-Control-String ::= CHOICE { mvi-start      [1] NULL,
                                       mvi-stop       [2] NULL,
                                       mvi-repeat-start [3] MVI-Repeat-Start,
                                       mvi-repeat-end   [4] NULL,
                                       mvi-move        [5] MVI-Move }
```

- Début MVI est une fonction prise de la Recommandation T.101 SD I (position dans le jeu de code MVI: 2/0)
- Fin MVI est une fonction prise de la Recommandation T.101 SD I (position dans le jeu de code MVI: 2/1)

#### A.3.11.1 Début de répétition MVI

MVI-Repeat-Start ::= SEQUENCE { GRAPHICSTRING, INTEGER }

- Caractère général (DÉBUT RÉPÉTITION) pris de la Recommandation T.101 SD I (position dans le jeu de code MVI: 3/12 ou 11/12), suivi d'un comptage du nombre de répétitions.
- Fin de répétition MVI est une fonction prise de la Recommandation T.101 SD I (position dans le jeu de code MVI: 3/13 ou 11/13).

#### A.3.11.2 Déplacement MVI

MVI-Move ::= SEQUENCE { Move-Origin, Move-Termination, Move-Time }

- Déplacement MVI est une fonction prise de la Recommandation T.101 SD I (position dans le jeu de code MVI: 3/10 ou 11/10).

Move-Origin ::= Abs-Coord

- Paramètres X, Y codés sous forme de fractions binaires remplies.

Move-Termination ::= OCTETSTRING

- Paramètres X, Y codés sous forme de fractions binaires remplies.

Move-Time ::= INTEGER

- Comptage numérique de la durée de l'opération de déplacement en 1/10 de seconde.

#### A.3.12 Chaîne de commande de segments

La syntaxe de données II offre une possibilité facultative de mémorisation et édition de segments. On retient une ou deux mémoires pour les segments d'affichage. Les commandes d'édition peuvent produire des effets dynamiques en modifiant le segment d'affichage mémorisé et en provoquant un nouvel affichage de l'image. Un segment d'affichage peut contenir n'importe quelles données de chaîne géométrique ainsi que les attributs de segment spéciaux décrits ci-après.

La commande des segments est similaire à la commande d'animation car elle fournit des fonctions qui commandent des possibilités spéciales dépendant des conditions d'affichage. Comme il n'existe pas de fonctions analogues dans la syntaxe de données I ou III, ces fonctions doivent être traitées dans le processus de conversion. Pour la conversion de la syntaxe de données II dans la syntaxe de données I ou III, on utilise un seul «poste de travail» (ou écran d'affichage).

Segment-Control-String ::= CHOICE { [1] Work-Station-Dependent,  
[2] Work-Station-Independent }

Work-Station-Dependent ::= CHOICE { [1] W-Create,  
[2] W-Close,  
[3] W-Rename,  
[4] W-Delete-1,  
[5] W-Delete-2,  
[6] W-Redraw,  
[7] W-Set-Highlight,  
[8] W-Set-Visibility,  
[9] W-Set-Seg-Transparent,  
[10] W-Set-Priority }

A.3.12.1.1 *Création W*

W-Create ::= INTEGER

-- Ouverture du segment identifié.

A.3.12.1.2 *Fermeture W*

W-Close ::= INTEGER

-- Fermeture du segment identifié.

A.3.12.1.3 *Réappellation W*

W-Rename ::= SEQUENCE {  
                                  old-segment-number [1] INTEGER,  
                                  new-segment-number [2] INTEGER }

-- Nouvelle appellation pour passer de l'ancien numéro de segment au nouveau numéro de segment.

A.3.12.1.4 *Oblitération W 1*

W-Delete-1 ::= SEQUENCE {  
                                  work-station-id [1] INTEGER,  
                                  segment-number [2] INTEGER }

-- Suppression du segment identifié sur le poste de travail.

A.3.12.1.5 *Oblitération W 2*

W-Delete-2 ::= INTEGER

-- Oblitération du segment identifié sur tous les postes de travail.

A.3.12.1.6 *Nouveau dessin W*

W-Redraw ::= INTEGER

-- Nouveau dessin sur le poste de travail identifié.

A.3.12.1.7 *Détermination surluminance W*

W-Set-Highlight ::= SEQUENCE {  
                                  highlight-segment-number [1] INTEGER,  
                                  highlight-attribute [2] INTEGER }

-- Détermination de l'attribut de surluminance du segment identifié.

A.3.12.1.8 *Détermination visibilité W*

W-Set-Visibility ::= SEQUENCE {  
                                  visibility-segment-number [1] INTEGER,  
                                  visibility-attribute [2] INTEGER }

-- Détermination de l'attribut de visibilité du segment identifié.

#### A.3.12.1.9 *Détermination seg transparent W*

W-Set-Seg-Transparent ::= SEQUENCE { transparent-segment-number [1] INTEGER,  
transform-matrix [2] MAT }

-- Détermination des attributs de la matrice de transformation pour le segment identifié.

MAT ::= SET { matrix-element-11 [11] REAL,  
matrix-element-12 [12] REAL,  
matrix-element-13 [13] REAL,  
matrix-element-21 [21] REAL,  
matrix-element-22 [22] REAL,  
matrix-element-23 [23] REAL }

-- Définition de la matrice de transformation.

#### A.3.12.1.10 *Détermination priorité W*

W-Set-Priority ::= SEQUENCE { priority-segment-number [1] INTEGER,  
priority-value [2] REAL }

-- Détermination de l'attribut de priorité de segment pour le segment identifié. Analogue à la priorité d'ordre d'affichage.

#### A.3.12.2 *Indépendant du poste de travail*

Work-Station-Independent ::= CHOICE { [1] W-Associated,  
[2] W-Copy,  
[3] W-Insert }

##### A.3.12.2.1 *Association W*

W-Associated ::= SEQUENCE { associated-w-station-id [1] INTEGER,  
associated-segment-number [2] INTEGER }

-- Association du segment identifié avec le poste de travail identifié.

##### A.3.12.2.2 *Copie W*

W-Copy ::= SEQUENCE { copy-w-station-id [1] INTEGER,  
copy-segment-number [2] INTEGER }

-- Copie des primitives du poste de travail identifié.

##### A.3.12.2.3 *Insertion W*

W-Insert ::= SEQUENCE { insert-segment-number [1] INTEGER,  
insert-transform-matrix-ref [2] MAT }

-- Transformation et affichage de segment.

#### A.3.13 *Chaîne de commande de couleurs*

Toutes les syntaxes de données de terminal source offrent la possibilité de définir la couleur et possèdent, au moins facultativement, une possibilité de carte de couleurs. Il existe cependant de grandes différences entre les modèles de couleurs utilisés par les diverses syntaxes. Afin d'établir une base neutre pour la couleur, on utilise ici le modèle de couleur élaboré pour la norme ISO 8613 (Texte and Offices Systems - Office Document Architecture).

Le modèle de couleur de base utilisé dans la norme ISO 8613 est un cube de couleurs construit sur les 3 vecteurs de base, rouge, vert et bleu (RVB). Dans ce modèle, une couleur est représentée par un triplet de composantes RVB. Du point de vue logique, ces couleurs sont normées de 0 (minimum) à 1 (maximum). En conséquence, noir est le

triplet <0,0,0> et blanc le triplet <1,1,1>. Toutes les syntaxes de données de terminal vidéotex ont des modèles de couleurs différents du modèle ci-dessus; la transposition des modèles de couleurs spécifiques vers le cube de couleurs RVB de base doit donc être comprise par le processus de conversion.

On dispose de deux modes d'indexage de couleurs: direct et indexé. Dans la sélection directe des couleurs, la couleur est définie par la spécification d'un triplet de valeurs discrètes pour les composantes RVB. Dans le mode de sélection des couleurs avec indexage, la couleur est définie par un index qui permet d'entrer dans un tableau de couleurs unique qui donne des valeurs de couleur discrètes. Le nombre des couleurs pouvant être défini dans ce tableau dépend du modèle de terminal. La limite admise par hypothèse dans la définition d'un ensemble de données particulier est spécifiée dans la section Définition des valeurs limites. Si un système de réception n'est pas capable de reproduire l'intervalle des valeurs de couleurs spécifié par une valeur de couleur directe ou par la valeur de couleur indexée par un index de couleur, on adopte un ajustement le plus proche, conformément aux principes énoncés dans la norme ISO 8613. Il existe une variante du mode couleur avec indexage, appelé mode couleur auxiliaire, qui est utilisé pour définir une couleur pour le fond d'une cellule de caractère de texte ou de mosaïque.

```
Colour-Control-String ::= CHOICE { [1] Basic-Colour-Selection,
                                   [2] Direct-Colour-Selection,
                                   [3] Indexed-Colour-Selection,
                                   [4] Auxiliary-Colour-Selection,
                                   [5] Colour-Index-Setup }
```

#### A.3.13.1 Sélection de couleur de base

```
Basic-Colour-Selection ::= INTEGER { black           (0),
                                     red             (1),
                                     green          (2),
                                     yellow         (3),
                                     blue           (4),
                                     magenta        (5),
                                     cyan           (6),
                                     white          (7),
                                     auxiliary-black (8),
                                     auxiliary-red   (9),
                                     auxiliary-green (10),
                                     auxiliary-yellow (11),
                                     auxiliary-blue  (12),
                                     auxiliary-magenta (13),
                                     auxiliary-cyan  (14),
                                     auxiliary-white  (15),
                                     auxiliary-foreground (16) }
```

- Plusieurs syntaxes de données de terminal permettent d'accéder simplement aux couleurs primaires de base, par utilisation des codes du jeu C1. Les divers ensembles de commandes C1 diffèrent fondamentalement en ce qui concerne les modalités de l'interaction entre la commande de couleurs et les autres attributs. Afin d'éviter cette difficulté dans l'interfonctionnement, on se bornera à spécifier ici les commandes de couleurs de base elles-mêmes. La gamme de spécification des couleurs dépend du modèle de terminal. On a échappé à cette tendance en définissant toutes les commandes de sélection de couleur sur la base du modèle de couleurs spécifié dans la norme ISO 8613. Les lois de dépendance de la gamme de couleurs (attributs de rangées série ou attributs de cellules parallèle) doivent être exprimées sur la base du modèle de couleurs abstraites par le processus de conversion. Cela signifie que toutes les règles inhérentes à la méthode des attributs série pour spécifier les couleurs de base, ou à la méthode des attributs parallèle, doivent être résolues par le processus de conversion qui crée les commandes de couleur de la SDI.
- Les commandes de couleurs auxiliaires spécifient la couleur du fond pour le texte et les mosaïques. La commande 'premier plan auxiliaire' spécifie que la couleur du fond doit être réglée sur la couleur du premier plan actuelle.

### A.3.13.2 *Sélection de couleur directe*

Direct-Colour-Selection ::= SEQUENCE { REAL, REAL, REAL }

-- La sélection de couleur directe permet de spécifier les couleurs sur la base des composantes rouge, vert, bleu, du modèle de couleurs. On utilise le type de données REAL (REEL) de ASN.1, car ce type de nombre est autodélimitant et de longueur arbitraire. Les paramètres de nombres réels sont rapportés à la valeur de couleur maximale pour chaque composante. Les paramètres sont respectivement rouge, vert et bleu.

### A.3.13.3 *Sélection de couleur avec indexage*

Indexed-Colour-Selection ::= INTEGER

-- La sélection de couleur avec indexage permet de spécifier les couleurs à l'aide d'un index pour entrer dans une carte de couleurs indirectes qui contient les spécifications de couleurs rouge, vert et bleu pour chaque couleur. La longueur de la carte de couleurs et le nombre des cartes de couleurs disponibles dépendent du modèle de terminal. Le paramètre INTEGER (ENTIER) est interprété en fonction de la taille actuelle de la carte de couleurs, spécifiée dans la définition des valeurs limites. Afin de respecter les règles énoncées pour tenir compte des différences entre les valeurs de couleurs (voir la norme ISO 8613), le paramètre INTEGER (ENTIER) est interprété comme une fraction normée de la longueur de carte spécifiée. Les cartes multiples sont logiquement équivalentes à une grande carte unique qui englobe plusieurs sous-cartes. Dans la SDI, l'utilisation de plusieurs cartes de couleurs est réalisée par partition arbitraire de la carte couleurs SDI unique.

### A.3.13.4 *Sélection de couleur auxiliaire*

Auxiliary-Colour-Selection ::= INTEGER

-- La sélection de couleur auxiliaire permet de spécifier des couleurs pour le fond des cellules de caractère de texte ou de mosaïques. L'application de cette commande est analogue à celle de la sélection de couleur avec indexage (voir ci-dessus), à ceci près qu'on fixe la couleur actuelle du fond.

### A.3.13.5 *Etablissement de l'index de couleurs*

Colour-Index-Setup ::= SEQUENCE { INTEGER, REAL, REAL, REAL }

-- La commande établissement de l'index de couleurs définit le contenu de la carte de couleurs. Le premier paramètre introduit les index dans la carte, de la même façon que pour la commande sélection de couleur avec indexage. Les trois autres paramètres définissent les valeurs des couleurs rouge, vert et bleu, de la même façon que pour la commande spécification de couleur directe.

### A.3.14 *Chaîne de commande de texte*

La manière dont le texte est présenté ainsi que les attributs spécialisés et contraintes relatifs à la présentation du texte, diffèrent selon la syntaxe de données de terminal considérée.

Text-Control-String ::= CHOICE { [1] General-Text-Control,  
[2] Word-Wrap-Control }

#### A.3.14.1 *Commande de texte générale*

General-Text-Control ::= SEQUENCE { [1] General-Text-Control-Code,  
[2] G-Text-Par1 OPTIONAL,  
[3] G-Text-Par2 OPTIONAL,  
[4] Rel-Coord OPTIONAL,  
[5] Abs-Coord OPTIONAL }

General-Text-Control-Code ::= GRAPHICSTRING

-- Fonction de commande générale prise de la Recommandation T.101, SD III [position dans le jeu G PDI: 2/2, (10/2)].

-- *Remarque* – Le jeu G PDI a comme caractère final 5/7 dans SD III.

```

G-Text-Par1 ::= SET { [1] Char-Rotation OPTIONAL,
                      [2] IMPLICIT Char-Path OPTIONAL,
                      [3] Char-Spacing OPTIONAL,
                      [4] IMPLICIT Text-Precision OPTIONAL,
                      [5] IMPLICIT Char-Expansion-Factor OPTIONAL,
                      [6] Text-Colour-Index OPTIONAL,
                      [7] IMPLICIT Text-Alignment OPTIONAL }

Char-Rotation ::= CHOICE { predefined [1] IMPLICIT INTEGER {
                                char-rotation-0      (0),
                                char-rotation-90     (1),
                                char-rotation-180    (2),
                                char-rotation-270    (3) }
                      continuous [2] IMPLICIT SEQUENCE {
                                height-vector Abs-Coord,
                                width-vector Abs-Coord } }

Char-Path ::= INTEGER { char-path-right (0),
                        char-path-left  (1),
                        char-path-up    (2),
                        char-path-down  (3) }

Char-Spacing ::= CHOICE { predefined [1] IMPLICIT INTEGER {
                                char-spacing-1      (0),
                                char-spacing-5/4   (1),
                                char-spacing-3/2   (2) }
                      continuous [2] IMPLICIT REAL }

Text-Precision ::= INTEGER { string (0),
                             char   (1),
                             stroke (2) }

Char-Expansion-Factor ::= REAL

Text-Control-Index ::= CHOICE { [1] IMPLICIT Basic-Colour-Selection,
                                [2] IMPLICIT Indexed-Colour-Selection }

Text-Alignment ::= SEQUENCE { Horizontal-Alignment,
                               Vertical-Alignment }

Horizontal-Alignment ::= INTEGER { normal (0),
                                  left   (1),
                                  centre (2),
                                  right  (3) }

Vertical-Alignment ::= INTEGER { normal (0),
                                 top    (1),
                                 cap    (2),
                                 half   (3),
                                 base   (4),
                                 bottom (5) }

G-Text-Par2 ::= SEQUENCE { INTEGER { cursor-style-underscore (0),
                                    cursor-style-block        (1),
                                    cursor-style-cross-hair    (2),
                                    cursor-style-custom         (3) }
                          INTEGER { cursor-&-geometric-drawing-position-together (0),
                                    cursor-leads-geometric-drawing-position      (1),
                                    geometric-drawing-position-leads-cursor      (1),
                                    cursor-&-geometric-drawing-position-separate  (3) }
                          INTEGER { char-interrow-spacing-1 (0),
                                    char-interrow-spacing-5/4 (1),
                                    char-interrow-spacing-3/2 (2),
                                    char-interrow-spacing-2   (3) }
                          Char-Block-Dimension }

```

-- Les coordonnées relatives définissent la taille du champ de caractère.

Char-Block-Dimensions ::= Rel-Coord

#### A.3.14.2 *Commande d'enroulement de mots*

Une des syntaxes de données de terminal offre la possibilité d'enrouler la présentation des caractères sur une limite de mot plutôt que sur une limite de caractère. Cette possibilité ne peut pas être convertie directement dans d'autres syntaxes de données; toutefois, l'effet peut être obtenu dans le convertisseur par l'émission de caractères appropriés de mise en page.

```
Word-Wrap-Control ::= INTEGER { Word-Wrap-On (1),  
                               Word-Wrap-Off (2) }
```

-- Avec enroulement de mots est une fonction prise de la Recommandation T.101 SD III [position dans le jeu C1: 5/5, (9/5)].

-- Sans enroulement de mots est une fonction prise de la Recommandation T.101, SD III [position dans le jeu C1: 5/6 (9/6)].

#### A.3.15 *Image synthétique de chaîne photographique*

Toutes les syntaxes de données de terminal fournissent une méthode pour traiter un ensemble de pixels. Certaines de ces syntaxes fournissent aussi des possibilités photographiques générales qui définissent des méthodes plus efficaces pour coder le même type de données. Cela signifie que l'interfonctionnement de toutes les syntaxes de données de terminal est possible pour les données photographiques, même s'il risque d'être inefficace dans certains cas.

Deux catégories d'images photographiques sont identifiées ci-après. Il s'agit de la forme synthétique et de la forme naturelle d'images photographiques. La forme synthétique correspond aux possibilités photographiques de la syntaxe de données I. Le codage naturel des images photographiques est encore à l'étude.

```
Photo-Graphic-String-Synthetic-Image ::= CHOICE { [1] Line-Dot-Pattern,  
                                                  [2] Line-Dot-Pattern-Comp,  
                                                  [3] Field-Dot-Pattern,  
                                                  [4] Colouring-Block,  
                                                  [5] Colouring-Block-Comp,  
                                                  [6] Field-Colouring-Block,  
                                                  [7] Field-Colouring-Block-Comp,  
                                                  [8] Free-Format-Colouring-Block }
```

-- Les fonctions d'image synthétique photographique correspondent à la Recommandation T.101, SD I. Ces fonctions se prêtent à l'affichage d'images synthétiques telles que les caractères kanji, les graphiques, etc.

##### A.3.15.1 *Modèle de ligne par points*

```
Line-Dot-Pattern ::= SEQUENCE { y-origin-point-coordinate-Idp Abs-Coord,  
                                dot-pattern-data-Idp BITSTRING }
```

-- Les fonctions de modèle par points indiquent une sélection de deux couleurs qui sont définies par un bloc de coloration, un bloc de coloration de champ, etc. Cette fonction fournit des données de modèle par points pour une seule ligne ou plusieurs lignes à la fois.

##### A.3.15.2 *Modèle comp de ligne par points*

```
Line-Dot-Pattern-Comp ::= SEQUENCE { y-origin-point-coordinate-Idpc Abs-Coord,  
                                     mh-run-length coded-data BITSTRING }
```

-- La fonction modèle comp de ligne par points est équivalente à la fonction modèle de ligne par points, à ceci près que les modèles par points sont codés avec compression, par utilisation du code de longueur de course M.H.

##### A.3.15.3 *Modèle de champ par points*

```
Field-Dot-Pattern ::= SEQUENCE { xy-origin-point-coordinate Abs-Coord,  
                                  dx-dy-field-dimensions Rel-Coord,  
                                  dot-pattern-data-fdp BITSTRING }
```

-- La fonction modèle de champ par points est équivalente à la fonction modèle de ligne par points, à ceci près que cette fonction définit le modèle par points dans une aire rectangulaire.

#### A.3.15.4 *Bloc de coloration*

Colouring-Block ::= SEQUENCE { fg-bg-da-existence-indicator INTEGER,  
y-origin-point-coordinate-cb Abs-Coord,  
SEQUENCE OF { SEQUENCE {  
fg-colour BITSTRING,  
bg-colour BITSTRING,  
display-attributes-cb BITSTRING } } }

-- La fonction bloc de coloration définit une image photographique en spécifiant la couleur du premier plan (FG), la couleur du fond (BG) et les attributs d'affichage de certains blocs dont l'aval est indiqué par le paramètre coordonnée du point d'origine y.

#### A.3.15.5 *Bloc de coloration comp*

Colouring-Block-Comp ::= SEQUENCE { colouring-block-comp-function-id INTEGER,  
fg-bg-da-existence-indicator-cbc INTEGER,  
y-origin-point-coordinate-cbs Abs-Coord,  
SEQUENCE OF { SEQUENCE {  
fg-comp-colour BITSTRING,  
fg-runlength BITSTRING,  
bg-comp-colour BITSTRING,  
bg-runlength BITSTRING,  
display-attributes-cbc BITSTRING,  
da-runlength BITSTRING } } }

-- La fonction bloc de coloration comp est équivalente à la fonction bloc de coloration, à ceci près que les données de couleur et d'attributs d'affichage sont codées avec compression comme un code de longueur de course.

#### A.3.15.6 *Bloc de coloration de champ*

Field-Colouring-Block ::= SEQUENCE { field-colouring-block-function-id INTEGER,  
fg-bg-da-existence-indicator-fcb INTEGER,  
xy-origin-point-coordinate-fcb Abs-Coord,  
dx-dy-field-dimensions-fcb Rel-Coord,  
SEQUENCE OF { SEQUENCE {  
fg-colour-fcb BITSTRING,  
bg-colour-fcb BITSTRING,  
display-attributes-fcb BITSTRING } } }

-- La fonction bloc de coloration de champ définit une image photographique en spécifiant la couleur du premier plan (FG), la couleur du fond (BG) et les attributs d'affichage de certains blocs qui sont contenus dans le champ assigné par les coordonnées des points d'origine xy et les dimensions de champ dx-dy.

### A.3.15.7 Bloc de coloration de champ comp

```
Field-Colouring-Block-Comp ::= SEQUENCE { field-colouring-block-comp-function-id INTEGER,  
fg-bg-da-existence-indicator-fcbc INTEGER,  
xy-origin-point-coordinate-fcbc Abs-Coord,  
dx-dy-field-dimensions-fcbc Rel-Coord,  
SEQUENCE OF { SEQUENCE {  
fg-colour-fcbc BITSTRING,  
fg-runlength-fcbc BITSTRING,  
bg-comp-colour-fcbc BITSTRING,  
bg-runlength-fcbc BITSTRING,  
display-attributes-fbc BITSTRING,  
da-runlength-fcbc BITSTRING }}}}
```

-- La fonction bloc de coloration de champ comp est équivalente à la fonction bloc de coloration de champ, à ceci près que les données de couleur et d'attributs d'affichage sont codées avec compression comme un code de longueur de course.

### A.3.15.8 Bloc de coloration de format libre

```
Free-Format-Colouring-Block ::= SEQUENCE { fg-bg-da-existence-indicator-ffcb INTEGER,  
fg-bg-da-code-length INTEGER,  
run-length-code-length-ffcb INTEGER,  
xy-origin-point-coordinate-ffcb Abs-Coord,  
dx-dy-field-dimensions-ffcb Rel-Coord,  
SEQUENCE OF { SEQUENCE {  
fg-colour-ffcb BITSTRING,  
runlength-ffcb BITSTRING,  
bg-comp-colour-ffcb BITSTRING,  
bg-runlength-ffcb BITSTRING,  
display-attributes-ffcb BITSTRING,  
da-runlength-ffcb BITSTRING }}}}
```

-- La fonction bloc de coloration de format libre est équivalente à la fonction bloc de coloration de champ comp, à ceci près qu'il est possible de fixer arbitrairement la longueur de code des attributs de premier plan, de fond et d'affichage ainsi que la longueur de course.

### A.3.16 Image naturelle de chaîne photographique

```

Photo-Graphic-String-Natural-Image ::= CHOICE { [0] IMPLICIT Header,
                                                [1] IMPLICIT Transfer,
                                                [2] IMPLICIT Table-Header,
                                                [3] IMPLICIT Table-Transfer }

Header ::= SET {
    [0] IMPLICIT Components OPTIONAL,
    CHOICE { [1] IMPLICIT Resolution OPTIONAL,
            [2] IMPLICIT PixelPair OPTIONAL }
            [3] IMPLICIT BitsPerDisplay OPTIONAL,
            [4] IMPLICIT SamplingStructure OPTIONAL,
    CHOICE { [5] IMPLICIT Adpcm OPTIONAL,
            [6] IMPLICIT Adct OPTIONAL } }

Components ::= INTEGER { colorYU*V* (0),
                        monochrome (1) }

Resolution ::= INTEGER { 4-2-2 (0),
                        2-1-1 (1) }

PixelPair ::= SEQUENCE { PixHor, PixVer }

PixHor ::= INTEGER
-- Nombre de pixels horizontaux.

PixVer ::= INTEGER
-- Nombre de pixels verticaux.

BitsPerDisplay ::= SEQUENCE OF INTEGER { 8 bits/pixel (0),
                                         1 bit/pixel (1),
                                         2 bits/pixel (2),
                                         ... 9 bits/pixel (9),... }

-- Une valeur par élément donne le nombre de gris ou de couleurs qu'un pixel peut avoir.

SamplingStructure ::= SEQUENCE {
    spatial { INTEGER { line and orthogonal (0),
                      line and orthogonal field quincunx (1),
                      line quincunx field orthogonal (2),
                      line orthogonal single field (3),
                      line quincunx single field (4) }
    temporal { INTEGER { coincident (0),
                      alternate samples (1),
                      sequential line (2) } }

Adpcm ::= SEQUENCE { INTEGER { Type dpcm (1) },
                    INTEGER { Subtype 1 dimension (0) } }

Adct ::= SEQUENCE { INTEGER { Type transform (2) },
                    INTEGER { Subtype Cosine (1) },
                    INTEGER { Subtype 2 dimension (0) } }

Transfer ::= SET { Origin, Area, Data }

Origin ::= [0] IMPLICIT PixelPair OPTIONAL
Area ::= [1] IMPLICIT PixelPair OPTIONAL
Data ::= CHOICE { [2] IMPLICIT OCTETSTRING OPTIONAL,
-- Valeur quelconque de 4/0 à 7/F.
                [3] IMPLICIT OCTETSTRING OPTIONAL }

-- Mode transparent octet/8 bits.

TableHeader ::= SET { TableSet, TableSize }

TableSet ::= [0] IMPLICIT SEQUENCE { type ::= INTEGER,
                                     number ::= INTEGER }

TableSize ::= [1] IMPLICIT SEQUENCE { depth ::= INTEGER,
                                     heigth ::= INTEGER,
                                     width ::= INTEGER OPTIONAL }

TableTransfer ::= SET { TableSet, Position, Data }

Position ::= TableSize

```

### A.3.17 Macro-instructions

Deux des trois syntaxes de données de terminal offrent une possibilité de macro-instructions. Cette possibilité permet de regrouper des chaînes de données de présentation, afin que l'exécution puisse se faire par référence à une commande unique. Les deux syntaxes SD I et SD III offrent essentiellement la même possibilité de macros; toutefois, en règle générale, une macro incluse dans une certaine syntaxe de données ne peut pas être convertie en une macro dans une autre syntaxe de données. La raison en est qu'une macro peut contenir une chaîne quelconque de données de présentation. En raison des différences entre les modèles de terminal des diverses syntaxes de données, il est souvent nécessaire de trier les commandes contenues dans le train de données pour obtenir l'effet de présentation recherché. Le groupage arbitraire de l'information dans des macros permet d'éviter le tri général. Le but des fonctions macros ordinaires est de fournir une efficacité de communications en éliminant la communication d'un code répétitif; il est donc possible de développer les macros dans le processus de conversion. De ce fait, la conversion d'une macro est la chaîne des données de présentation qu'elle représente.

Dans les syntaxes de données SD I et SD III, on a deux formes spéciales de macros: les macros déclenchées par touche et les macros de transmission. Les macros déclenchées par touche sont un lien entre l'exécution de la fonction macro et une touche locale du terminal. Comme cette opération dépend de l'interaction avec l'utilisateur, le contenu de la macro ne peut pas être développé par anticipation dans le convertisseur. Le convertisseur doit transmettre au terminal la totalité de la page d'information, avec le contenu de la macro déclenchée par touche, trié et mis en facteur dans la page. Ce problème doit être traité par l'architecture de présentation d'interfonctionnement. Une macro de transmission pose aussi un problème de conversion. Le contenu de cette macro doit être renvoyé à la source par interaction avec l'utilisateur. Dans l'interfonctionnement, cela pourrait signifier que des données de la syntaxe SD , contenues dans une macro de transmission, pourraient être présentes dans un terminal de la SD III après une conversion, le résultat étant que ces données pourraient être renvoyées à la source sans changement. Pour éviter la confusion, il est nécessaire de pouvoir identifier toutes les conditions de codage, ou identifier de façon univoque chaque tableau de code dans chaque syntaxe de données.

MACRO-String ::= CHOICE { [1] Define-Macro,  
[2] Define-and-Execute-Macro,  
[3] Define-Transmit-Macro,  
[4] Define-End-of-Macro-Definition,  
[5] Macro-Invocation }

-- Les macros déclenchées par touche sont des macros ayant des numéros de référence de 0 à 7 dans la syntaxe de données SD III.

#### A.3.17.1 Définition macro

Define-Macro ::= SEQUENCE { SID, INTEGER }

-- Caractère de commande général (DEF MACRO) pris de la Recommandation T.101, SD III [position dans le jeu C1: 4/0, (8/0)] et (P-DEF MACRO) de SD I [position dans le jeu C1: 5/5, (9/5) suivi du paramètre 4/0].

-- Nombre entier de 0 à 95 correspondant au numéro de référence de macro pour la macro que l'on définit.

#### A.3.17.2 Définition et exécution macro

Define-and-Execute-Macro ::= SEQUENCE { SID, INTEGER }

-- Caractère de commande général (DEFP MACRO) pris de la Recommandation T.101, SD III [position dans le jeu C1: 4/1, (8/1)] et (P-DEFP MACRO) de SD I [position dans le jeu C1: 5/5, (9/5) suivi du paramètre 4/1].

-- Nombre entier de 0 à 95 correspondant au numéro de référence macro pour la macro que l'on définit.

#### A.3.17.3 Définition macro de transmission

Define-Transmit-Macro ::= SEQUENCE { SID, INTEGER }

-- Caractère de commande général (DEFT MACRO) pris de la Recommandation T.101, SD III [position dans le jeu C1: 4/2, (8/2)] et (P-DEFT MACRO) de SD I [position dans le jeu C1: 5/5, (9/5) suivi du paramètre 4/2].

-- Nombre entier de 0 à 95 correspondant au numéro de référence macro pour la macro que l'on définit.

#### A.3.17.4 Définition de fin de définition de macro

Define-End-of-Macro-Definition ::= SID

- Caractère de commande général [END (Macro)] pris de la Recommandation T.101, SD III [position dans le jeu C1: 4/5, (8/5)] et (END MACRO) de SD I [position dans le jeu C1: 5/5, (9/5) suivi du paramètre 4/15].

#### A.3.17.5 Appel de macro

Macro-Invocation ::= INTEGER

- Nombre entier de 0 à 95 correspondant au numéro de référence macro pour la macro que l'on appelle.
- *Remarque* – Les macros peuvent appeler d'autres macros à tout moment et jusqu'à une profondeur quelconque.

#### A.3.18 Chaîne JCDR

La facilité appelée jeu de caractères dynamiquement redéfinissables (JCDR) permet de définir des caractères supplémentaires pour le texte ou les mosaïques, et de les utiliser comme des caractères ordinaires de texte alphanumérique ou de mosaïques. Les trois syntaxes de données de terminal offrent une possibilité de JCDR, mais le fonctionnement du JCDR varie beaucoup selon les conditions d'affichage. Il n'est généralement pas possible de faire une conversion exacte entre un type de JCDR et un autre, en raison des conditions aux limites imposées par chacune des syntaxes de données de terminal. Des limites différentes sont imposées au nombre de caractères JCDR pouvant être définis ou à la capacité de mémoire utilisable pour stocker ces caractères. La définition des caractères JCDR présente une difficulté particulière. Dans une des syntaxes de données de terminal source, on admet tous les modes de présentation pour cette définition: commandes de dessin géométriques, bits (photographique) et texte, et même d'autres caractères JCDR. Dans les deux autres syntaxes de données de source, les caractères JCDR sont définis selon une méthode basée sur les bits (photographique). Même les deux méthodes photographiques de définition ne sont pas équivalentes; elles comportent en effet des densités de pixels différentes et on risque d'avoir des erreurs de quantification importantes dans la transposition d'une série de pixels sur une autre série de taille différente. La syntaxe de données d'interfonctionnement comporte trois types de définition de JCDR, pour tenir compte des caractéristiques des trois syntaxes de données de source. Le processus de conversion dispose ainsi d'une quantité d'information suffisante pour effectuer la meilleure conversion possible.

DRCS-String ::= CHOICE { [1] Define-DRCS-Type-I-1byte,  
[2] Define-DRCS-Type-I-2byte,  
[3] Define-DRCS-Type-II,  
[4] Define-DRCS-Type-III,  
[5] End-of-DRCS-Definition-Type-III,  
[6] DRCS-Invocation,  
[7] DRCS-Invocation-2byte }

##### A.3.18.1 Définition JCDR type I, 1 multiplet

Define-DRCS-Type-I-1byte ::= SEQUENCE { DRCS-I-Char-Size,  
DRCS-I-Code,  
DRCS-I-Data }

DRCS-I-Char-Size ::= INTEGER { normal-size (1),  
medium-size (2),  
small-size (3) }

DRCS-I-Code ::= INTEGER

- Nombre entier de 0 à 95 correspondant au numéro de référence JCDR pour le JCDR à 1 multiplet que l'on appelle.

DRCS-I-Data ::= BITSTRING

### A.3.18.2 Définition JCDR type I, 2 multipléts

```
Define-DRCS-Type-I-2byte ::= SEQUENCE { DRCS-I-Char-Size,
                                         DRCS-I-Code,
                                         DRCS-I-Data }
```

-- Cette structure est la même que "Define-DRCS-Type-I-1byte", à ceci près que "DRCS-I-Code" est un nombre entier de 0 à 8835 correspondant au numéro de référence JCDR pour le JCDR à multipléts que l'on appelle.

### A.3.18.3 Définition JCDR type II

```
Define-DRCS-Type-II ::= SEQUENCE { [1] IMPLICIT DRCS-Header OPTIONAL,
```

-- Description des propriétés générales du JCDR à charger. Appliqué pour toutes les unités de transfert subséquentes de modèle JCDR.

```
                                [2] IMPLICIT DRCS-Pattern OPTIONAL }
```

-- Données du modèle effectif.

```
DRCS-Header ::= SEQUENCE { Identification-of-Char-Set,
                           Select-Dot-Composition }
```

```
Identification-of-Char-Set ::= SEQUENCE { repertory-info SET {
                                         repertory-# INTEGER { first repertory      (1),
                                                                second repertory   (2) },
                                         delete-existing-drcs BOOLEAN
                                         registration-info CHOICE {
                                         iso-registration      [1] IMPLICIT GRAPHICSTRING,
                                         private-drcs-#        [2] IMPLICIT INTEGER } }
```

```
Select-Dot-Composition ::= SEQUENCE { Character-Cell-Structure,
                                       Blocking-Factor,
                                       Pixel-Characteristics }
```

```
Character-Cell-Structure ::= CHOICE { matrix-dimensions [1] IMPLICIT SEQUENCE {
                                       horizontal          INTEGER,
                                       vertical            INTEGER },
```

-- Selon SDC type 1.

```
predefined-matrices [2] IMPLICIT INTEGER {
                                         n16*24          (0),
                                         n16*20          (1),
                                         n16*12          (2),
                                         n16*10          (3),
                                         n12*24          (4),
                                         n12*20          (5),
                                         n12*12          (6),
                                         n12*10          (7),
                                         n8*12           (8),
                                         n8*10           (9),
                                         n6*12           (10),
                                         n6*10           (11),
                                         n6*5            (12),
                                         n4*10           (13),
                                         n4*5            (14),
                                         n6*6            (15) }
```

-- Selon SDC type 2.

```
Blocking-Factor ::= SEQUENCE { horizontal INTEGER,
                               vertical   INTEGER }
```

-- Selon SDC type 2.

Blocking-Factor ::= SEQUENCE { horizontal INTEGER,  
vertical INTEGER }

-- Groupement de cellules de caractère, considérées comme formant une seule cellule de caractère pendant la description du caractère.

Pixel-Characteristics ::= CHOICE { number of bits [1] IMPLICIT INTEGER,  
predefined-numbers [2] IMPLICIT INTEGER }  
basic-DRCS (1),  
four-colour-DRCS (4),  
eight-colour-DRCS (8),  
sixteen-colour-DRCS (16) }

-- 1 bit/point.

-- JCDR à 4 couleurs (4), noir, rouge, vert, jaune, de 'sélection couleur de base'.

-- eight-colour-DRCS (8), 8 premières couleurs de 'sélection couleur de base'.

-- sixteen-colour-DRCS (16), 16 couleurs redéfinissables.

-- Ce type de données décrit les modèles pour les caractères du JCDR téléchargé, sur la base de l'unité d'en-tête transmise en dernier. Elle ne comporte pas de compression pour les données de modèle. Il n'existe pas de codages similaires dans les syntaxes de données I et III, et cette méthode peut être utilisée pour une transposition adéquate. Tous les codages différents de la méthode directe ainsi que les codes servant à améliorer l'efficacité (multiplets S) doivent être transformés pour la description suivante.

DRCS-Pattern ::= SEQUENCE { first character GRAPHICSTRING,

-- Code du premier caractère du bloc de caractères.

pattern-units SEQUENCE OF {  
pattern-block-# SEQUENCE OF INTEGER,  
pattern-block BIT STRING } }

-- Chaque bloc de modèle contient un bit de chacun des points, en commençant par le coin supérieur gauche et en passant de rangée en rangée de la gauche vers la droite. Les numéros de bloc de modèle sont rangés à partir du bit de moindre poids. Si le bloc de modèle est précédé de deux numéros de bloc ou davantage, le bloc de modèle est appliqué à tous. Les numéros de bloc vont de 0 à `caractéristiques de pixel` 1. La longueur du bloc de modèle est numériquement égale au nombre des pixels contenus dans le bloc.

#### A.3.18.4 Définition JCDR type III

Define-DRCS-Type-III ::= INTEGER

-- Fonction prise de la Recommandation T.101, SD III [position dans le jeu C1:4/3, (8/3)].

-- Nombre entier de 0 à 95 correspondant au numéro de référence JCDR pour le caractère JCDR que l'on définit et qui sera suivi d'une chaîne de données.

#### A.3.18.5 Fin de définition de JCDR type III

End-of-DRCS-Definition-Type-III ::= GRAPHICSTRING

-- Caractère de commande général [END (DRCS)] pris de la Recommandation T.101, SD III [position dans le jeu C1: 4/5, (8/5)].

#### A.3.18.6 Appel JCDR

DRCS-Invocation ::= INTEGER

-- Nombre entier de 0 à 95 correspondant au numéro de référence JCDR pour le JCDR que l'on appelle.

### A.3.18.7 Appel JCDR à 2 multiplets

DRCS-Invocation-2byte ::= INTEGER

- Nombre entier de 0 à 8835 correspondant au numéro de référence JCDR pour le JCDR à 2 multiplets que l'on appelle.

### A.3.19 Chaîne de commande de motif de remplissage

Deux des syntaxes de données vidéotex de source offrent la possibilité de remplir une zone définie géométriquement avec un motif de remplissage arbitraire: style intérieur, hachures ou texture. Une des syntaxes de données vidéotex de terminal, SD I, n'offre pas cette possibilité; il faut donc en tenir compte dans le processus de conversion en assignant des couleurs distinctives, ou d'autres moyens, pour indiquer la différence entre les zones remplies par des motifs. La méthode de mise en oeuvre de cette possibilité dans les deux autres syntaxes de données de source est très différente. La syntaxe de données SD III fournit quatre motifs de texture prédéfinis, parmi lesquels le remplissage plein, et quatre masques de texture redéfinissables. Ces masques sont rectilignes et rapportés à l'origine de la zone d'affichage normée. Cela signifie qu'il y aura un alignement parfait entre des zones contiguës qui sont remplies par le même motif. Dans les motifs du type style intérieur définis dans SD II, le motif peut être défini sur une zone en forme de parallélogramme et il est rapporté à l'origine de la zone. La syntaxe de données SD II fournit aussi huit motifs de remplissage prédéfinis (hachures). En général, toute texture d'un motif du type style intérieur peut être simulée dans le processus de conversion, mais on ne peut pas garantir des effets secondaires tels que l'alignement exact des motifs. Dans SD III, les motifs de texture sont définis par l'insertion d'une chaîne de données de présentation dans la définition du motif; par contre, dans SD II les styles intérieurs sont définis sur la base d'un carrelage. Le processus de conversion doit déterminer le motif avant d'effectuer la conversion. Les limites à imposer aux variables globales, par exemple la capacité disponible de mémoire de texture, sont définies par les indicateurs de conditions de valeurs limites dans le vecteur d'état.

Fill-Pattern-Control-String ::= CHOICE { [1] Define-Texture,  
[2] End-of-Texture-Definition,  
[3] Texture-Mask-Size,  
[4] Set-Pattern-Representation,  
[5] Pattern-Selection }

#### A.3.19.1 Définition texture

Define-Texture ::= INTEGER

- Fonction prise de la Recommandation T.101, SD III [position dans le jeu: 4/4, (8/4)].
- Nombre entier de 4 à 7 correspondant au masque de texture redéfinissable devant être défini. *Remarque* - Les masques de texture 0 à 3 sont prédéfinis; ils ne peuvent pas être redéfinis pour être suivis d'une chaîne de données.

#### A.3.19.2 Fin de définition texture

End-of-Texture-Definition ::= GRAPHICSTRING

- Caractère de commande général [END (TEXTURE)] pris de la Recommandation T.101, SD III [position dans le jeu C1: 4/5, (8/5)].

#### A.3.19.3 Taille masque de texture

Texture-Mask-Size ::= Rel-Coord

- Fixation de la taille du masque de texture, jusqu'aux limites définies par les conditions aux limites.
- Fonction prise de la Recommandation T.101, SD III [position dans le jeu PDI: 2/3, (10/3)].

#### A.3.19.4 Détermination de la représentation de motif

Set-Pattern-Representation ::= SEQUENCE { pattern-index INTEGER,

-- Ce nombre correspond à la définition du motif actuel. On peut y faire référence au moyen des indices de style subséquents de zone remplie.

delta-x INTEGER,

delta-y INTEGER,

-- Spécification d'une grille delta-x\* delta-y\*. La couleur de chaque cellule est indiquée par les

pattern-cell-data Pixel-Array-Data }

-- On a l'association suivante entre la palette de couleurs et les cellules: l'élément (1, delta-y) est associé à la cellule dont le point de référence de motif se trouve dans un angle. Les éléments, dans l'ordre croissant de leur première dimension, sont associés à des cellules successives dans la direction du vecteur de largeur de motif; les éléments, dans l'ordre décroissant de leur deuxième dimension, sont associés à des cellules successives dans la direction du vecteur de hauteur de motif.

-- Ces définitions de motifs sont prises de SD II; elles sont applicables en relation avec les attributs de zone remplie définis par le type de données "attributs d'éléments d'affichage".

#### A.3.19.5 Sélection motif

Pattern-Selection ::= INTEGER

-- Nombre entier de 0 à 7 correspondant au masque de texture que l'on sélectionne.

#### A.3.20 Chaîne musicale

La facilité musicale est une option spécifique à une seule des syntaxes de données de terminal. Elle exige des possibilités spéciales et ne peut pas être convertie d'une manière raisonnable. L'information de musique est incluse dans la syntaxe de données d'interfonctionnement en vue d'une compatibilité future, de telle sorte qu'on puisse réaliser l'interfonctionnement entre l'information fournie par SD I et toutes versions futures de SD II ou III qui pourraient comporter une facilité musicale.

Music-String ::= CHOICE { [1] Music-Code-Sequence,  
[2] Music-Control-Sequence }

##### A.3.20.1 Séquence code musique

Music-Code-Sequence ::= GRAPHICSTRING

-- Caractères pris de la Recommandation T.101, SD I [Ensemble de tonalité musical (hauteur/durée)].  
*Remarque* – L'ensemble de tonalité musical est un ensemble à deux multiplats qui peut être décrit comme la combinaison de deux ensembles à un multiplat, l'un pour la durée et l'autre pour la hauteur. Il est fait référence à T.101 car ce tableau de code n'a pas encore été enregistré.

##### A.3.20.2 Séquence commande musique

Music-Control-Sequence ::= GRAPHICSTRING

-- Caractères de commande pris de la Recommandation T.101, SD I [ensemble C1 commande musicale]. L'ensemble commande musicale comprend les fonctions suivantes: début séquence musicale, fin séquence musicale, début partie mélodie, début partie rythme, fin partie rythme, étiquette musicale, saut à partie, répétition musicale, branche musicale, niveau son, changement de timbre, pause ou tonalité de longue durée. Il est fait référence à T.101 car ce tableau de code n'a pas encore été enregistré.

#### A.3.21 Chaîne de télélogiciel

Telesoftware-String ::= Complément d'étude

Audio-Data-String ::= Complément d'étude

## APPENDICE I

(à la Recommandation T.101)

**Répertoires de caractères de texte et de caractères mosaïques**

Dans la syntaxe de données d'interfonctionnement, tous les caractères de texte et les caractères mosaïques ont un nom de code afin de pouvoir être identifiés individuellement. On trouve ci-dessous un répertoire exhaustif de tous les caractères de texte et caractères mosaïques utilisés dans les syntaxes de données spécifiées dans la Recommandation T.101. Cela simplifie nombre de références aux jeux de caractères graphiques utilisés dans la SDI, car il suffit d'utiliser les noms de code dans le corps de la description de l'ASN.1 de la syntaxe de données d'interfonctionnement. Aucune des syntaxes de données vidéotex n'utilise tous les caractères de texte et les caractères mosaïques recensés ci-dessous. Dans certains domaines, les différentes syntaxes de données se chevauchent largement. Afin de faciliter le transcodage et les conversions, on a identifié plusieurs catégories. Des répertoires distincts ont été établis pour chacune de ces catégories.

I.1 *Répertoire 1* – Caractères de texte alphanumériques communs

Le répertoire 1 est le répertoire commun des caractères de texte alphanumériques de base. Ces caractères sont tirés des jeux de caractères primaires et supplémentaires de la Recommandation T.51, les caractères finals répertoriés étant 4/0 et 6/2 respectivement. En outre, le répertoire comprend le caractère ESPACE (SP01) ainsi que le caractère OBLITÉRATION (SM34). Les noms descriptifs donnés aux caractères diffèrent selon les diverses syntaxes de données de terminal définies dans la Recommandation T.101 et selon le répertoire de caractères alphanumériques définis dans la norme ISO 6937. On trouvera ici des noms composites, ce qui permet d'englober toute la gamme de significations spécifiées dans les différentes syntaxes de données de terminal recensées dans la Recommandation T.101 tout en maintenant le plus de cohérence possible avec la norme ISO 6937.

I.1.1 *Caractères alphabétiques latins*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
LA01	a minuscule	LC15	c minuscule avec accent circonflexe
LA02	A majuscule	LC16	C majuscule avec accent circonflexe
LA11	a minuscule avec accent aigu	LC21	c minuscule avec caron
LA12	A majuscule avec accent aigu	LC22	C majuscule avec caron
LA13	a minuscule avec accent grave	LC29	c minuscule avec point
LA14	A majuscule avec accent grave	LC30	C majuscule avec point
LA15	a minuscule avec accent circonflexe	LC41	c minuscule avec cédille
LA16	A majuscule avec accent circonflexe	LC42	C majuscule avec cédille
LA17	a minuscule avec tréma ou signe umlaut	LD01	d minuscule
LA18	A majuscule avec tréma ou signe umlaut	LD02	D majuscule
LA19	a minuscule avec tilde	LD21	d minuscule avec caron
LA20	A majuscule avec tilde	LD22	D majuscule avec caron
LA23	a minuscule avec brève	LD61	d minuscule avec barre
LA24	A majuscule avec brève	LD62	D majuscule avec barre (eth islandais)
LA27	a minuscule avec rond	LD63	eth minuscule islandais
LA28	A majuscule avec rond	LE01	e minuscule

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
LA31	a minuscule avec macron	LE02	E majuscule
LA32	A majuscule avec macron	LE11	e minuscule avec accent aigu
LA43	a minuscule avec ogonek	LE12	E majuscule avec accent aigu
LA44	A majuscule avec ogonek	LE13	e minuscule avec accent grave
LA51	diphthongue æ minuscule	LE14	E majuscule avec accent grave
LA52	diphthongue Æ majuscule	LE15	e minuscule avec accent circonflexe
LB01	b minuscule	LE16	E majuscule avec accent circonflexe
LB02	B majuscule	LE17	e minuscule avec tréma ou signe umlaut
LC01	c minuscule	LE18	E majuscule avec tréma ou signe umlaut
LC02	C majuscule	LE21	e minuscule avec caron
LC11	c minuscule avec accent aigu	LE22	E majuscule avec caron
LC12	C majuscule avec accent aigu	LE29	e minuscule avec point
LE30	E majuscule avec point	LL12	L majuscule avec accent aigu
LE31	e minuscule avec macron	LL21	l minuscule avec caron
LE32	E majuscule avec macron	LL22	L majuscule avec caron
LE43	e minuscule avec ogonek	LL41	l minuscule avec cédille
LE44	E majuscule avec ogonek	LL42	L majuscule avec cédille
LF01	f minuscule	LL61	l minuscule avec barre
LF02	F majuscule	LL62	L majuscule avec barre
LG01	g minuscule	LL63	l minuscule avec point au milieu
LG02	G majuscule	LL64	L majuscule avec point au milieu
LG11	g minuscule avec accent aigu	LM01	m minuscule
LG15	g minuscule avec accent circonflexe	LM02	M majuscule
LG16	G majuscule avec accent circonflexe	LN01	n minuscule
LG23	g minuscule avec brève	LN02	N majuscule
LG24	G majuscule avec brève	LN11	n minuscule avec accent aigu
LG29	g minuscule avec point	LN12	N majuscule avec accent aigu
LG30	G majuscule avec point	LN19	n minuscule avec tilde
LG42	G majuscule avec cédille	LN20	N majuscule avec tilde
LH01	h minuscule	LN21	n minuscule avec caron
LH02	H majuscule	LN22	N majuscule avec caron
LH15	h minuscule avec accent circonflexe	LN41	n minuscule avec cédille
LH16	H majuscule avec accent circonflexe	LN42	N majuscule avec cédille
LH61	h minuscule avec barre	LN61	eng minuscule, lapon
LH62	H majuscule avec barre	LN62	ENG majuscule, lapon
LI01	i minuscule	LN63	n minuscule avec apostrophe
LI02	I majuscule	LO01	o minuscule
LI11	i minuscule avec accent aigu	LO02	O majuscule
LI12	I majuscule avec accent aigu	LO11	o minuscule avec accent aigu
LI13	i minuscule accent grave	LO12	O majuscule avec accent aigu
LI14	I majuscule avec accent grave	LO13	o minuscule avec accent grave
LI15	i minuscule avec accent circonflexe	LO14	O majuscule avec accent grave
LI16	I majuscule avec accent circonflexe	LO15	o minuscule avec accent circonflexe
LI17	i minuscule avec tréma ou signe umlaut	LO16	O majuscule avec accent circonflexe
LI18	I majuscule avec tréma ou signe umlaut	LO17	o minuscule avec tréma ou signe umlaut
LI19	i minuscule avec tilde	LO18	O majuscule avec tréma ou signe umlaut
LI20	I majuscule avec tilde	LO19	o minuscule avec tilde
LI30	I majuscule avec point	LO20	O majuscule avec tilde
LI31	i minuscule avec macron	LO25	o minuscule avec double accent aigu
LI32	I majuscule avec macron	LO26	O majuscule avec double accent aigu
LI43	i minuscule avec ogonek	LO31	o minuscule avec macron

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
LI44	I majuscule avec ogonek	LO32	O majuscule avec macron
LI51	ligature ij minuscule	LO51	ligature œ minuscule
LI52	ligature IJ majuscule	LO52	ligature Œ majuscule
LI61	i minuscule sans point	LO61	o minuscule avec barre oblique
LJ01	j minuscule	LO62	O majuscule avec barre oblique
LJ02	J majuscule	LP01	p minuscule
LJ15	j minuscule avec accent circonflexe	LP02	P majuscule
LJ16	J majuscule avec accent circonflexe	LQ01	q minuscule
LK01	k minuscule	LQ02	Q majuscule
LK02	K majuscule	LR01	r minuscule
LK41	k minuscule avec cédille	LR02	R majuscule
LK42	K majuscule avec cédille	LR11	r minuscule avec accent aigu
LK61	k minuscule groenlandais	LR12	R majuscule avec accent aigu
LL01	l minuscule	LR21	r minuscule avec caron
LL02	L majuscule	LR22	R majuscule avec caron
LL11	l minuscule avec accent aigu	LR41	r minuscule avec cédille
LR42	R majuscule avec cédille	LU23	u minuscule avec brève
LS01	s minuscule	LU24	U majuscule avec brève
LS02	S majuscule	LU25	u minuscule avec double accent aigu
LS11	s minuscule avec accent aigu	LU26	U majuscule avec double accent aigu
LS12	S majuscule avec accent aigu	LU27	u minuscule avec rond
LS15	s minuscule avec accent circonflexe	LU28	U majuscule avec rond
LS16	S majuscule avec accent circonflexe	LU31	u minuscule avec macron
LS21	s minuscule avec caron	LU32	U majuscule avec macron
LS22	S majuscule avec caron	LU43	u minuscule avec ogonek
LS41	s minuscule avec cédille	LU44	U majuscule avec ogonek
LS42	S majuscule avec cédille	LV01	v minuscule
LS61	double s minuscule, allemand	LV02	V majuscule
LT01	t minuscule	LW01	w minuscule
LT02	T majuscule	LW02	W majuscule
LT21	t minuscule avec caron	LW15	w minuscule avec accent circonflexe
LT22	T majuscule avec caron	LW16	W majuscule avec accent circonflexe
LT41	t minuscule avec cédille	LX01	x minuscule
LT42	T majuscule avec cédille	LX02	X majuscule
LT61	t minuscule avec barre	LY01	y minuscule
LT62	T majuscule avec barre	LY02	Y majuscule
LT63	thorn minuscule, islandais	LY11	y minuscule avec accent aigu
LT64	Thorn majuscule, islandais	LY12	Y majuscule avec accent aigu
LU01	u minuscule	LY15	y minuscule avec accent circonflexe
LU02	U majuscule	LY16	Y majuscule avec accent circonflexe
LU11	u minuscule avec accent aigu	LY17	y minuscule avec tréma ou signe umlaut
LU12	U majuscule avec accent aigu	LY18	Y majuscule avec tréma ou signe umlaut
LU13	u minuscule avec accent grave	LZ01	z minuscule
LU14	U majuscule avec accent grave	LZ02	Z majuscule
LU15	u minuscule avec accent circonflexe	LZ11	z minuscule avec accent aigu
LU16	U majuscule avec accent circonflexe	LZ12	Z majuscule avec accent aigu
LU17	u minuscule avec tréma ou signe umlaut	LZ21	z minuscule avec caron
LU18	U majuscule avec tréma ou signe umlaut	LZ22	Z majuscule avec caron
LU19	u minuscule avec tilde	LZ29	z minuscule avec point
LU20	U majuscule avec tilde	LZ30	Z majuscule avec point

### I.1.2 *Caractères non alphabétiques*

#### I.1.2.1 *Chiffres décimaux*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
ND01	chiffre 1	ND06	chiffre 6
ND02	chiffre 2	ND07	chiffre 7
ND03	chiffre 3	ND08	chiffre 8
ND04	chiffre 4	ND09	chiffre 9
ND05	chiffre 5	ND10	chiffre 0

#### I.1.2.2 *Signes monétaires*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
SC01	symbole monétaire universel	SC04	symbole centime
SC02	symbole livre	SC05	symbole yen
SC03	symbole dollar		

#### I.1.2.3 *Signes de ponctuation*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
SP01	ESPACEMENT	SP13	deux points
SP02	point d'exclamation	SP14	point virgule
SP03	point d'exclamation inversé	SP15	point d'interrogation
SP04	guillemets	SP16	point d'interrogation inversé
SP05	apostrophe	SP17	crochet aigu gauche
SP06	parenthèse gauche	SP18	crochet aigu droit
SP07	parenthèse droite	SP19	guillemet simple gauche
SP08	virgule	SP20	guillemet simple droit
SP10	tiret (ou signe moins)	SP21	guillemet double gauche
SP11	point (ou point décimal)	SP22	guillemet double droit
SP12	barre oblique		

#### I.1.2.4 *Signes arithmétiques*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
SA01	signe plus	SA05	signe supérieur à
SA02	signe plus/moins	SA06	signe diviser
SA03	signe inférieur à	SA07	signe multiplier
SA04	signe égal		

#### I.1.2.5 *Exposants*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
NS01	indice supérieur 1	NS03	indice supérieur 3
NS02	indice supérieur 2		

### I.1.2.6 Fractions

Code de nom	Nom descriptif	Code de nom	Nom descriptif
NF01	fraction 1/2	SM40	fraction 3/8 (équivalent à NF19)
NF04	fraction 1/4	SM41	fraction 5/8 (équivalent à NF20)
NF05	fraction 3/4	SM42	fraction 7/8 (équivalent à NF21)
SM39	fraction 1/8 (équivalent à NF18)		

### I.1.2.7 Symboles divers

Code de nom	Nom descriptif	Code de nom	Nom descriptif
SM01	symbole de numéro	SM17	symbole micro
SM02	symbole pour cent	SM18	symbole ohm
SM03	perluète ou «et» commercial	SM19	symbole degré
SM04	étoile, astérisque	SM20	indicateur ordinal, masculin
SM05	à commercial	SM21	indicateur ordinal, féminin
SM06	crochet rectangulaire (gauche)	SM24	symbole section
SM07	barre oblique inversée	SM25	symbole paragraphe
SM08	crochet rectangulaire (droit)	SM26	point central
SM11	parenthèse gauche, accolade gauche	SM30	flèche vers la gauche
SM12	barre horizontale jointive	SM31	flèche vers la droite
SM13	barre verticale jointive	SM32	flèche vers le haut
SM14	parenthèse droite, accolade droite	SM33	flèche vers le bas

*Remarque* – Les caractères SM43, SM44, SM47, SM48 ont des noms multiples du fait que les noms descriptifs de ces caractères diffèrent sensiblement selon les syntaxes de données de terminal définies dans la Recommandation T.101. A l'origine, ces caractères devaient représenter des caractères d'accent selon la table de codes IRV de la norme ISO 646. Leur signification a changé depuis l'introduction de la méthode composite de codage des caractères accentués définie dans la Recommandation T.51 du CCITT, dans la norme ISO 6937 et dans plusieurs Recommandations du CCITT. C'est pourquoi des caractères ne doivent pas servir à créer des caractères accentués. Pour que la compatibilité avec les normes précédentes soit maintenue, on trouvera ici des noms multiples.

### I.1.2.8 Signes diacritiques (tels qu'ils s'affichent lorsqu'ils sont utilisés en conjonction avec le caractère ESPACE)

Code de nom	Nom descriptif	Code de nom	Nom descriptif
SD11	accent aigu	SD25	double accent aigu
SD13	accent grave	SD27	rond
SD15	accent circonflexe	SD29	point
SD17	tréma ou signe umlaut	SD31	macron
SD19	tilde	SD41	cédille
SD21	caron	SD43	ogonek
SD23	brève		

## I.2 Répertoire 2 – Caractères alphanumériques spéciaux

Ces caractères n'appartiennent qu'à une ou deux syntaxes de données de terminal et leur présentation doit être convertie afin que leur effet de présentation puisse être obtenu dans une autre syntaxe de données de terminal.

Code de nom	Nom descriptif	Code de nom	Nom descriptif
SM50	barre vectorielle autre que d'espacement	SM45	barre verticale jointive à gauche
SM51	barre oblique autre que d'espacement	SM46	barre verticale jointive à droite

*Remarque* – Les codes de nom SM50 et SM51 sont introduits ici parce que des codes de nom de ces caractères ne figurent ni dans le registre de l'ISO (table de codes, répertoriée dans le registre n° 99) ni dans la Recommandation T.101 du CCITT.

### I.3 *Répertoire 3* – Caractères kana

Tout ce jeu de caractères n'appartient qu'à une seule des syntaxes de données de terminal; en conséquence, la présentation de ces caractères doit être convertie afin que leur effet de présentation puisse être obtenu dans une autre syntaxe de données de terminal.

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
JA01	point katakana	JA07	a minuscule katakana
JA02	parenthèse gauche katakana	JA08	i minuscule katakana
JA03	parenthèse droite katakana	JA09	u minuscule katakana
JA04	virgule katakana	JA10	e minuscule katakana
JA05	symbole de conjonction katakana	JA11	o minuscule katakana
JA06	WO katakana	JA12	ya katakana
JA13	yu minuscule katakana	JA39	NU katakana
JA14	yo katakana	JA40	NE katakana
JA15	tsu katakana	JA41	NO katakana
JA16	symbole sonore prolongé	JA42	HA katakana
JA17	A katakana	JA43	HI katakana
JA18	I katakana	JA44	FU katakana
JA19	U katakana	JA45	HE katakana
JA20	E katakana	JA46	HO katakana
JA21	O katakana	JA47	MA katakana
JA22	KA katakana	JA48	MI katakana
JA23	KI katakana	JA49	MU katakana
JA24	KU katakana	JA50	ME katakana
JA25	KE katakana	JA51	MOI katakana
JA26	KO katakana	JA52	YA katakana
JA27	SA katakana	JA53	YU katakana
JA28	SHI katakana	JA54	YO katakana
JA29	SU katakana	JA55	RA katakana
JA30	SE katakana	JA56	RI katakana
JA31	SO katakana	JA57	RU katakana
JA32	TA katakana	JA58	RE katakana
JA33	CHI katakana	JA59	RO katakana
JA34	TSU katakana	JA60	WA katakana
JA35	TE katakana	JA61	N ou M katakana
JA36	TO katakana	JA62	symbole sonore vocalisé
JA37	NA katakana	JA63	symbole sonore semi-vocalisé
JA38	NI katakana		

*Remarque* – Du fait que les codes de nom correspondant à ces caractères ne figurent ni dans le registre de l'ISO (registre numéro 13) ni dans la Recommandation T.101 du CCITT, on trouvera ici des codes qui commencent par JA.

### I.4 *Répertoire 4* – Caractères kanji

Ces caractères n'appartiennent qu'à une des syntaxes de données de terminal, bien que la table de codes des caractères kanji répertoriée soit quelquefois utilisée avec les jeux de caractères d'une autre syntaxe de données de terminal. A moins que les caractères kanji soient disponibles aux deux extrémités d'un échange, la présentation de ces caractères doit être convertie afin que leur effet de présentation puisse être obtenu dans l'autre syntaxe de données de terminal. Le jeu de caractère kanji est répertorié sous forme de jeu à deux multiplats. La Recommandation T.101 du CCITT utilise un sous-répertoire de ce jeu qui comprend quelque 3639 caractères, dont 2980 sont des caractères de symbole kanji. Ces caractères n'appartiennent qu'au présent répertoire. Les autres caractères, y compris un certain

nombre de caractères spéciaux ainsi que certains caractères alphabétiques pour d'autres langues, appartiennent également à d'autres jeux de caractères répertoriés. Par exemple, ce répertoire comprend un alphabet cyrillique, grec, katakana et hiragana, ainsi que tous les caractères de l'alphabet latin, à l'exclusion de 11, du répertoire 1. Les caractères identiques dans cette table de codes ont les mêmes *codes de nom* que les caractères équivalents dans le répertoire 1 et peuvent donc être convertis directement entre les syntaxes de données. Pour les autres caractères, il est nécessaire de procéder à un traitement particulier. En outre, ce répertoire compte 32 caractères de dessin semblables à ceux du répertoire 8. Ces caractères portent également les mêmes *codes de nom* que ceux d'autres jeux de caractères et font donc partie du grand répertoire de caractères de dessin (répertoire 8).

Le répertoire d'idéogrammes kanji et de caractères particuliers uniques est volumineux. Puisqu'il est nécessaire de procéder à un traitement spécial pour établir des conversions à partir de ce répertoire de caractères, il n'est pas nécessaire d'en dresser la liste ici. Il convient de se référer à la Recommandation T.101 du CCITT, syntaxe de données I, qui constitue un sous-répertoire du registre numéro 87 de l'ISO. Les codes de nom kanji JK01 à JK2890, HK01 à HK83 et JS01 à JS366 ont été introduits pour identifier les caractères d'idéogrammes kanji, les caractères hiragana ainsi que les caractères spéciaux kanji.

### I.5 Répertoire 5 – Caractères de l'alphabet grec

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
GA01	lettre grecque alpha minuscule	GN01	lettre grecque nu minuscule
GA02	lettre grecque alpha majuscule	GN02	lettre grecque nu majuscule
GA11	alpha minuscule avec accent	GO01	lettre grecque omicron minuscule
GA12	lettre grecque alpha majuscule avec accent	GO02	lettre grecque omicron majuscule
GB01	lettre grecque bêta minuscule	GO11	lettre grecque omicron minuscule avec accent
GB02	lettre grecque bêta majuscule	GO12	lettre grecque omicron majuscule avec accent
GD01	lettre grecque delta minuscule	GO61	lettre grecque oméga minuscule
GD02	lettre grecque delta majuscule	GO62	lettre grecque oméga majuscule
GE01	lettre grecque epsilon minuscule	GO63	lettre grecque oméga minuscule avec accent
GE02	lettre grecque epsilon majuscule	GO64	lettre grecque oméga majuscule avec accent
GE11	lettre grecque epsilon minuscule avec accent	GP01	lettre grecque pi minuscule
GE12	lettre grecque epsilon majuscule avec accent	GP02	lettre grecque pi majuscule
GE61	lettre grecque eta minuscule	GP61	lettre grecque psi minuscule
GE62	lettre grecque eta majuscule	GP62	lettre grecque psi majuscule
GE63	lettre grecque eta minuscule avec accent	GR01	lettre grecque rho minuscule
GE64	lettre grecque eta majuscule avec accent	GR02	lettre grecque rho majuscule
GF01	lettre grecque phi minuscule	GS01	lettre grecque sigma minuscule
GF02	lettre grecque phi majuscule	GS02	lettre grecque sigma majuscule
GG01	lettre grecque gamma minuscule	GS61	lettre grecque sigma finale minuscule
GG02	lettre grecque gamma majuscule	GT01	lettre grecque tau minuscule
GH01	lettre grecque khi minuscule	GT02	lettre grecque tau majuscule
GH02	lettre grecque khi majuscule	GT61	lettre grecque theta minuscule
GI01	lettre grecque iota minuscule	GT62	lettre grecque theta majuscule
GI02	lettre grecque iota majuscule	GU01	lettre grecque upsilon minuscule
GI11	lettre grecque iota minuscule avec accent	GU02	lettre grecque upsilon majuscule
GI12	lettre grecque iota majuscule avec accent	GU11	lettre grecque upsilon minuscule avec accent
GI17	lettre grecque iota minuscule avec tréma	GU12	lettre grecque upsilon majuscule avec accent
GI18	lettre grecque iota majuscule avec tréma	GU17	upsilon minuscule avec tréma
GI33	lettre grecque iota minuscule avec accent et tréma	GU18	upsilon majuscule avec tréma
GK01	lettre grecque kappa minuscule	GU33	upsilon minuscule avec accent et tréma
GK02	lettre grecque kappa majuscule	GX01	lettre grecque xi minuscule
GL01	lettre grecque lambda minuscule	GX02	lettre grecque xi majuscule
GL02	lettre grecque lambda majuscule	GZ01	lettre grecque zeta minuscule
GM01	lettre grecque mu minuscule	GZ02	lettre grecque zeta majuscule
GM02	lettre grecque mu majuscule	SD33	tréma et accent avec espacement

I.6 *Répertoire 6 – Caractères de l'alphabet cyrillique*

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
KA01	a cyrillique minuscule	KD02	de cyrillique majuscule
KA02	a cyrillique majuscule	KE01	je cyrillique minuscule
KA61	ja cyrillique minuscule	KE02	je cyrillique majuscule
KA62	ja cyrillique majuscule	KE17	jo cyrillique minuscule
KB01	be cyrillique minuscule	KE18	jo cyrillique majuscule
KB02	be cyrillique majuscule	KE61	e cyrillique minuscule
KC01	tse cyrillique minuscule	KE62	e cyrillique majuscule
KC02	tse cyrillique majuscule	KF01	eff cyrillique minuscule
KC21	tche cyrillique minuscule	KF02	eff cyrillique majuscule
KC22	tche cyrillique majuscule	KG01	gue cyrillique minuscule
KC61	chtcha cyrillique minuscule	KG02	gue cyrillique majuscule
KC62	chtcha cyrillique majuscule	KH01	kha cyrillique minuscule
KD01	de cyrillique minuscule	KH02	kha cyrillique majuscule
KI23	i bref cyrillique minuscule	KI01	i cyrillique minuscule
KI24	i bref cyrillique majuscule	KI02	i cyrillique majuscule
KJ01	zhe cyrillique minuscule	KS24	sha cyrillique majuscule
KJ02	zhe cyrillique majuscule	KT01	te cyrillique minuscule
KL01	el cyrillique minuscule	KT02	te cyrillique majuscule
KL02	el cyrillique majuscule	KU01	v cyrillique minuscule
KM01	em cyrillique minuscule	KU02	v cyrillique majuscule
KM02	em cyrillique majuscule	KV61	ju cyrillique minuscule
KN01	en cyrillique minuscule	KV62	ju cyrillique majuscule
KN02	en cyrillique majuscule	KV01	ve cyrillique minuscule
KO01	o cyrillique minuscule	KV02	ve cyrillique majuscule
KO02	o cyrillique majuscule	KY01	ieri cyrillique minuscule
KP01	pe cyrillique minuscule	KY02	ieri cyrillique majuscule
KP02	pe cyrillique majuscule	KY61	ier cyrillique minuscule (signe mou)
KR01	er cyrillique minuscule	KY62	ier cyrillique majuscule (signe mou)
KR02	er cyrillique majuscule	KY63	ier' cyrillique minuscule (signe dur)
KS01	es cyrillique minuscule	KY64	ier' cyrillique majuscule (signe dur)
KS02	es cyrillique majuscule	KZ01	ze cyrillique minuscule
KS23	sha cyrillique minuscule	KZ02	ze cyrillique majuscule
		KK01	ka cyrillique minuscule
		KK02	ka cyrillique majuscule

I.7 *Répertoire 7 – Caractères mosaïques de bloc*

Les caractères mosaïques de bloc sont un sous-répertoire des caractères mosaïques utilisés dans chaque syntaxe de données de terminal et sont donc directement transposables entre syntaxes de données. Même le codage des caractères mosaïques de bloc est identique, sauf pour les caractères mosaïques pleins. Un caractère mosaïque plein est inclus dans le répertoire 5. L'un ou l'autre des deux codages équivalents définis dans la Recommandation T.101 (position dans la table de codage: 5/15 ou 7/15) pour les caractères mosaïques pleins est acceptable dans une chaîne de caractères du répertoire 5.

*Remarque* – On utilise la convention suivante pour décrire la forme d'un caractère mosaïque de 2 sous-cellules de largeur et 3 sous-cellules de hauteur. Les cellules sont numérotées ainsi: la cellule supérieure gauche porte le numéro 1, la cellule supérieure droite le numéro 2, la cellule centrale gauche le numéro 3, la cellule centrale droite le numéro 4, la cellule inférieure gauche le numéro 5 et la cellule inférieure droite le numéro 6. Ce code est utilisé pour identifier chaque caractère du répertoire.

1	2
3	4
5	6

T0803710-49

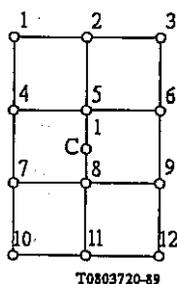
### Structure en sous-cellules des caractères mosaïques

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
MG00	caractère mosaïque vide	MG09	mosaïque de bloc 1,4
MG01	mosaïque de bloc 1	MG10	mosaïque de bloc 2,4
MG02	mosaïque de bloc 2	MG11	mosaïque de bloc 1,2,4
MG03	mosaïque de bloc 1,2	MG12	mosaïque de bloc 3,4
MG04	mosaïque de bloc 3	MG13	mosaïque de bloc 1,3,4
MG05	mosaïque de bloc 1,3	MG14	mosaïque de bloc 2,3,4
MG06	mosaïque de bloc 2,3	MG15	mosaïque de bloc 1,2,3,4
MG07	mosaïque de bloc 1,2,3	MG16	mosaïque de bloc 5
MG08	mosaïque de bloc 4	MG17	mosaïque de bloc 1,5
		MG18	mosaïque de bloc 2,5
MG19	mosaïque de bloc 1,2,5	MG43	mosaïque de bloc 1,2,4,6
MG20	mosaïque de bloc 3,5	MG44	mosaïque de bloc 3,4,6
MG21	mosaïque de bloc 1,3,5	MG45	mosaïque de bloc 1,3,4,6
MG22	mosaïque de bloc 2,3,5	MG46	mosaïque de bloc 2,3,4,6
MG23	mosaïque de bloc 1,2,3,5	MG47	mosaïque de bloc 1,2,3,4,6
MG24	mosaïque de bloc 4,5	MG48	mosaïque de bloc 5,6
MG25	mosaïque de bloc 1,4,5	MG49	mosaïque de bloc 1,5,6
MG26	mosaïque de bloc 2,4,5	MG50	mosaïque de bloc 2,5,6
MG27	mosaïque de bloc 1,2,4,5	MG51	mosaïque de bloc 1,2,5,6
MG28	mosaïque de bloc 3,4,5	MG52	mosaïque de bloc 3,5,6
MG29	mosaïque de bloc 1,3,4,5	MG53	mosaïque de bloc 1,3,5,6
MG30	mosaïque de bloc 2,3,4,5	MG54	mosaïque de bloc 2,3,5,6
MG31	mosaïque de bloc 1,2,3,4,5	MG55	mosaïque de bloc 1,2,3,5,6
MG32	mosaïque de bloc 6	MG56	mosaïque de bloc 4,5,6
MG33	mosaïque de bloc 1,6	MG57	mosaïque de bloc 1,4,5,6
MG34	mosaïque de bloc 2,6	MG58	mosaïque de bloc 2,4,5,6
MG35	mosaïque de bloc 1,2,6	MG59	mosaïque de bloc 1,2,4,5,6
MG36	mosaïque de bloc 3,6	MG60	mosaïque de bloc 3,4,5,6
MG37	mosaïque de bloc 1,3,6	MG61	mosaïque de bloc 1,3,4,5,6
MG38	mosaïque de bloc 2,3,6	MG62	mosaïque de bloc 2,3,4,5,6
MG39	mosaïque de bloc 1,2,3,6	MG63	mosaïque de bloc 1,2,3,4,5,6 (également équivalent à MG64-T.101, syntaxe de données III)
MG40	mosaïque de bloc 4,6		
MG41	mosaïque de bloc 1,4,6		
MG42	mosaïque de bloc 2,4,6		

#### I.8 Répertoire 8 – Caractères mosaïques lissés alignés en sous-cellules

L'ensemble général de caractères mosaïques lissés alignés en sous-cellules font partie du répertoire de caractères mosaïques utilisé dans deux des syntaxes de données de terminal. Ils ont le même codage. Ces caractères sont identifiés séparément comme un répertoire de la syntaxe de données d'interfonctionnement étant donné qu'ils sont directement transposables entre deux des syntaxes de données. La présentation de ces caractères doit être transformée pour obtenir l'effet de présentation dans un terminal utilisant une syntaxe de données n'acceptant pas ces caractères.

*Remarque* – On utilise la convention suivante pour décrire la forme d'un caractère mosaïque de 2 sous-cellules de largeur et 3 sous-cellules de hauteur. Les sommets des sous-cellules sont numérotés à partir du coin supérieur gauche de la sous-cellule supérieure, comme illustré ci-après. Le caractère mosaïque lissé est rempli en dessous (B), en-dessus (A), à droite (R), à gauche (L), d'une ligne ou de plusieurs lignes par les sommets indiqués.



### Sommets des sous-cellules de mosaïques

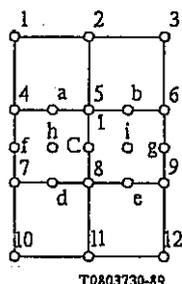
<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
SG01	caractère mosaïque lissé aligné en sous-cellules B-7,11	SG29	caractère mosaïque lissé aligné en sous-cellules A-7,11
SG02	caractère mosaïque lissé aligné en sous-cellules B-7,12	SG30	caractère mosaïque lissé aligné en sous-cellules A-7,12
SG03	caractère mosaïque lissé aligné en sous-cellules B-4,11	SG31	caractère mosaïque lissé aligné en sous-cellules A-4,11
SG04	caractère mosaïque lissé aligné en sous-cellules B-4,12	SG32	caractère mosaïque lissé aligné en sous-cellules A-4,12
SG05	caractère mosaïque lissé aligné en sous-cellules B-1,11	SG33	caractère mosaïque lissé aligné en sous-cellules A-1,11
SG06	caractère mosaïque lissé aligné en sous-cellules B-1,12	SG34	caractère mosaïque lissé aligné en sous-cellules A-1,12
SG07	caractère mosaïque lissé aligné en sous-cellules B-4,2	SG35	caractère mosaïque lissé aligné en sous-cellules A-4,2
SG08	caractère mosaïque lissé aligné en sous-cellules B-4,3	SG36	caractère mosaïque lissé aligné en sous-cellules A-4,3
SG09	caractère mosaïque lissé aligné en sous-cellules B-7,2	SG37	caractère mosaïque lissé aligné en sous-cellules A-7,2
SG10	caractère mosaïque lissé aligné en sous-cellules B-7,3	SG38	caractère mosaïque lissé aligné en sous-cellules A-7,3
SG11	caractère mosaïque lissé aligné en sous-cellules B-10,2	SG39	caractère mosaïque lissé aligné en sous-cellules A-10,2
SG12	caractère mosaïque lissé aligné en sous-cellules B-7,6	SG40	caractère mosaïque lissé aligné en sous-cellules A-7,6
SG13	caractère mosaïque lissé aligné en sous-cellules R-1,c,10	SG41	caractère mosaïque lissé aligné en sous-cellules L-1,c,10
SG14	caractère mosaïque lissé aligné en sous-cellules B-1,c,3	SG42	caractère mosaïque lissé aligné en sous-cellules A-1,c,3
SG15	caractère mosaïque lissé aligné en sous-cellules A-10,c,12	SG43	caractère mosaïque lissé aligné en sous-cellules A-11,9
SG16	caractère mosaïque lissé aligné en sous-cellules L-3,c,12	SG44	caractère mosaïque lissé aligné en sous-cellules A-10,9
SG17	caractère mosaïque lissé aligné en sous-cellules B-4,9	SG45	caractère mosaïque lissé aligné en sous-cellules A-11,6
SG18	caractère mosaïque lissé aligné en sous-cellules B-2,12	SG46	caractère mosaïque lissé aligné en sous-cellules A-10,6
SG19	caractère mosaïque lissé aligné en sous-cellules B-1,9	SG47	caractère mosaïque lissé aligné en sous-cellules A-11,3
SG20	caractère mosaïque lissé aligné en sous-cellules B-2,9	SG48	caractère mosaïque lissé aligné en sous-cellules A-10,3

Code de nom	Nom descriptif	Code de nom	Nom descriptif
SG21	caractère mosaïque lissé aligné en sous-cellules B-1,6	SG49	caractère mosaïque lissé aligné en sous-cellules A-2,6
SG22	caractère mosaïque lissé aligné en sous-cellules B-2,6	SG50	caractère mosaïque lissé aligné en sous-cellules A-1,6
SG23	caractère mosaïque lissé aligné en sous-cellules B-10,3	SG51	caractère mosaïque lissé aligné en sous-cellules A-2,9
SG24	caractère mosaïque lissé aligné en sous-cellules B-11,3	SG52	caractère mosaïque lissé aligné en sous-cellules A-1,9
SG25	caractère mosaïque lissé aligné en sous-cellules B-10,6	SG53	caractère mosaïque lissé aligné en sous-cellules A-2,12
SG26	caractère mosaïque lissé aligné en sous-cellules B-11,6	SG54	caractère mosaïque lissé aligné en sous-cellules A-4,9
SG27	caractère mosaïque lissé aligné en sous-cellules B-10,9	SG55	caractère mosaïque lissé aligné en sous-cellules R-3,c,12
SG28	caractère mosaïque lissé aligné en sous-cellules B-11,9	SG56	caractère mosaïque lissé aligné en sous-cellules B-10,c,12

### I.9 Répertoire 9 – Caractères mosaïques lissés généraux

On dispose d'une autre forme de caractères mosaïques lissés dans une seule des syntaxes de données de terminal. Ces caractères mosaïques lissés correspondent aux contours des demi-cellules. Etant donné que ces caractères n'appartiennent qu'à une seule syntaxe de données de terminal, leur présentation doit être transformée pour obtenir l'effet de présentation dans une autre syntaxe de données de terminal.

*Remarque* – La notation utilisée pour décrire ces caractères est identique à celle utilisée dans le répertoire 6 à l'exception des points intermédiaires qui correspondent aux demi-sous-cellules. Dans le cas où 4 nombres sont donnés, par exemple pour les caractères SG58 et SG71, la zone est délimitée de tous les côtés par les positions des quatre sous-cellules. Dans deux cas particuliers, pour les caractères SG68 et SG81, deux zones de la cellule mosaïque sont remplies.



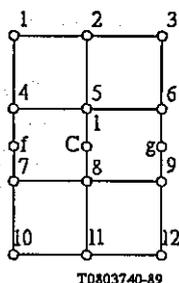
**Sommets des sous-cellules de mosaïques et points intermédiaires**

Code de nom	Nom descriptif	Code de nom	Nom descriptif
MS01	caractère mosaïque lissé général B-f,c	MS18	caractère mosaïque lissé général A-e,d & B-a,d
MS03	caractère mosaïque lissé général B-h,i	MS19	caractère mosaïque lissé général A-h,i
MS04	caractère mosaïque lissé général B-F,C,2	MS20	caractère mosaïque lissé général L-2,c,g
MS05	caractère mosaïque lissé général A-f,11	MS21	caractère mosaïque lissé général A-11,g
MS06	caractère mosaïque lissé général A-1,g	MS22	caractère mosaïque lissé général A-f,3
MS07	caractère mosaïque lissé général L-1,g,10	MS23	caractère mosaïque lissé général R-3,f,12
MS08	caractère mosaïque lissé général B-f,12	MS24	caractère mosaïque lissé général B-10,g
MS09	caractère mosaïque lissé général A-f,11,g	MS25	caractère mosaïque lissé général B-f,2,g
MS10	caractère mosaïque lissé général L-3,f,12	MS26	caractère mosaïque lissé général R-1,g,10
MS11	caractère mosaïque lissé général A-f,2,g	MS27	caractère mosaïque lissé général B-f,11,g
MS12	caractère mosaïque lissé général L-1,c,10 & R-3,c,12	MS28	caractère mosaïque lissé général A-1,c,3 & B-10,c,12
MS16	caractère mosaïque lissé général B-f,g		
MS17	caractère mosaïque lissé général B-c,g		

I.10 *Répertoire 10 – Caractères de tracé*

Un certain nombre de caractères de tracé, essentiellement des caractères de tracé de lignes, sont disponibles dans les diverses syntaxes de données de terminal. Il s'agit des caractères de tracé DG01 à DG04, DG13 à 24 et DG32, qui sont communs aux syntaxes de données I et II définies dans la Recommandation T.101 du CCITT, des caractères de tracé DG35 à DG50 qui n'appartiennent qu'à la syntaxe de données I, des caractères de tracé DG05 à DG12 et DG25 à DG31 qui n'appartiennent qu'à la syntaxe de données II et des caractères de tracé DG33 et DG34 qui n'appartiennent qu'à la syntaxe de données III.

*Remarque* – On utilise la notation suivante pour décrire ces caractères: la méthode d'identification des sommets des sous-cellules du caractère et des points intermédiaires est identique à celle utilisée pour les caractères mosaïques du répertoire 7; toutefois, on utilise uniquement des lignes pour relier le sommet et les points intermédiaires. Certains des caractères de tracé comportent plus d'un élément de ligne. Cette caractéristique est signalée par la description de 2 éléments de ligne séparés par «&». En outre, l'épaisseur (la largeur) du trait peut varier selon les caractères de tracé. Les traits épais (plus larges) sont indiqués par (W). Des caractères graphiques de tracé particuliers sont décrits avec des mots.



**Sommets des sous-cellules du caractère et points intermédiaires**

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
DG01	caractère de tracé W-f,g & c,2	DG26	caractère de tracé f,g (avec flèche dirigée vers la gauche)
DG02	caractère de tracé W-f,g & c,11	DG27	caractère de tracé 2,11 (avec flèche dirigée vers le haut)
DG03	caractère de tracé 2,11 & W-c,g	DG28	caractère de tracé 2,11 (avec flèche dirigée vers le bas)
DG04	caractère de tracé 2,11 & W-f,c	DG29	caractère de tracé (petit point central)
DG05	caractère de tracé 2,f,11	DG30	caractère de tracé (gros point central)
DG06	caractère de tracé 2,g,11	DG31	caractère de tracé (gros point central vide)
DG07	caractère de tracé f,11,g	DG32	caractère de tracé (caractère moucheté)
DG08	caractère de tracé f,2,g	DG33	caractère de tracé 3,10
DG09	caractère de tracé f,2		
DG10	caractère de tracé g,2		
DG11	caractère de tracé f,11	DG34	caractère de tracé 1,12
DG12	caractère de tracé g,11	DG35	caractère de tracé W-2,11
DG13	caractère de tracé W-f,g & 2,11	DG36	caractère de tracé W-f,g
DG14	caractère de tracé 2,11	DG37	caractère de tracé W-11,c,g
DG15	caractère de tracé f,g	DG38	caractère de tracé W-f,c,11
DG16	caractère de tracé 11,c,g	DG39	caractère de tracé W-2,c,g
DG17	caractère de tracé f,c,11	DG40	caractère de tracé W-f,c,2
DG18	caractère de tracé 2,c,g	DG41	caractère de tracé W-2,11 & W-c,g
DG19	caractère de tracé f,c,2	DG42	caractère de tracé W-2,11 & W-f,c
DG20	caractère de tracé 2,11 & c,g	DG43	caractère de tracé W-f,g & W-e,11
DG21	caractère de tracé 2,11 & f,c	DG44	caractère de tracé W-f,g & W-c,2
		DG45	caractère de tracé W-f,g & W-2,11

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
DG22	caractère de tracé f,g & c, 11	DG46	caractère de tracé W-2,11 & c,g
DG23	caractère de tracé f,g & c,2	DG47	caractère de tracé W-2,11 & f,c
DG24	caractère de tracé f,g & 2,11	DG48	caractère de tracé f,g & W-c,11
DG25	caractère de tracé f,g (avec flèche dirigée vers la droite)	DG49	caractère de tracé f,g & W-c,2
		DG50	caractère de tracé W-2, 11 & f,g

*Remarque* – Les codes de nom DG33 à DG50 sont indiqués ici parce que des codes applicables à ces caractères ne sont répertoriés ni dans le registre de l'ISO, ni dans la Recommandation T.101 du CCITT.

#### I.11 *Répertoire 11* – Mosaïques spéciales

Un certain nombre de formes de mosaïques spéciales existe dans les différentes syntaxes de données. Ces formes sont généralement uniques et ne figurent que dans une seule des syntaxes de données de terminal. Il est nécessaire de convertir la présentation de ces caractères pour obtenir leur même effet dans une autre syntaxe de données de terminal.

<i>Code de nom</i>	<i>Nom descriptif</i>	<i>Code de nom</i>	<i>Nom descriptif</i>
MS02	petit carré central	MS29	demi-ovale à droite remplie
MS13	demi-ovale ouverte à gauche	MS30	demi-ovale à gauche inverse
MS14	demi-ovale ouverte à droite	MS31	demi-ovale à droite inverse
MS15	demi-ovale à gauche remplie		

## APPENDICE II

(à la Recommandation T.101)

### **Etats par défaut de la syntaxe de données d'interfonctionnement**

La syntaxe de données d'interfonctionnement permet de choisir les états par défaut à établir. Ces valeurs par défaut représentent les états par défaut des syntaxes de données interactives correspondant à la source des données. La consultation des tableaux II-1/T.101 à II-3/T.101 est un moyen commode pour établir les états nécessaires pour assurer l'interfonctionnement avec une source de données particulière. Seules trois tables des valeurs par défaut sont décrites. Si un profil spécifique d'une syntaxe de données ne correspond pas exactement à la table des valeurs par défaut indiquée ci-après, on peut utiliser des commandes vecteur d'état explicite pour modifier les états qui diffèrent.

Les tables des valeurs par défaut comprennent les valeurs par défaut de la fonction et des paramètres de la syntaxe de données ainsi que celles des conditions aux limites. On suppose toujours que l'indicateur priorité de modèle de terminal prend au départ la valeur 1 et que toutes les variables globales sont modifiées bien entendu au point initial ou lors d'une réinitialisation.

*Remarque* – Dans le tableau II-2/T.101 indiqué ci-après, la présence d'un point d'interrogation «?» signifie que l'information pour cette entrée dans la table doit être confirmée. Un point d'interrogation seul indique que l'information doit être fournie dans une version ultérieure du document.

TABLEAU II-1/T.101

## Jeu des valeurs par défaut I: Valeurs par défaut correspondant à la syntaxe de données I

<i>Valeurs des conditions aux limites par défaut</i>	
– Dimensions de l'écran	– 0,969; 0,797
– Limite de la palette	– 16
– Sous-zone de présentation	– plein écran
– Contraintes du mode caractère	– 15,5; 8
– Limite des coordonnées-polygone	– 256 (forme pleine)
– Limite des coordonnées-courbe de lissage	– non applicable
– Résolution de présentation	– 248, 204 (rangée de base)
– Limite de mémoire des macro-segments	– 3072 multiplats (min)
– Limite de mémoire du JCDR	– 416 multiplats (min)
<i>Valeurs par défaut des paramètres de présentation</i>	
position de texte courante	– coin gauche supérieur
couleur de forme courante	– blanche
couleur auxiliaire courante	– bleu d'intensité réduite
état de soulignement	– arrêt
état de clignotement	– arrêt
état de taille des caractères de base	– caractère = normale 16/256, 24/256
état de groupage des couleurs	– 4/256, 4/256
état de masquage	– arrêt
état de case avec inversion de caractères	– arrêt
état de marquage des caractères	– non applicable
état de protection de l'écran	– protégé
état de commande d'affichage	– défilement = arrêt
état de commande d'appareil	– non applicable
état de commande de curseur	– non applicable
position géométrique courante	– coin gauche inférieur
état de la commande 1 de présentation géométrique	– structure du trait = plein, motif de structure = plein mise en valeur = arrêt élément d'image logique = 0,0
état de la commande 2 de présentation géométrique	– non applicable
état d'attente	– aucun
état général du texte	– non applicable
état du sens d'écriture	– angle de rotation des caractères = 0° sens d'écriture = droite
approche des caractères = 1	– curseur = souligner interrangée = espace simple
état de texte de présentation géométrique	– non applicable
état de définition de JCDR	– aucun
état de définition macro	– aucun
état de motif de structure	– non applicable
état de mémoire de la partie musicale	– tonalité musicale = fixe commande musicale = fixe mémoire = aucune
état de configuration d'animation	– pas de trame d'animation ordre de trame = normal
état de configuration du poste de travail	– non applicable

TABLEAU II-1/T.101 (suite)

<i>Palette par défaut – Syntaxe de données I</i>					
	Adresse de palette		Valeur de couleur		
			R	V	B
noir	0	0000	0001	0001	0001
rouge	1	0001	1111	0000	0000
vert	2	0010	0000	1111	0000
jaune	3	0011	1111	1111	0000
bleu	4	0100	0000	0000	1111
lilas	5	0101	1111	0000	1111
turquoise	6	0110	0000	1111	1111
blanc	7	0111	1111	1111	1111
transparent	8	1000	0000	0000	0000
rouge IR	9	1001	0111	0000	0000
vert IR	10	1010	0000	0111	0000
jaune IR	11	1011	0111	0111	0000
bleu IR	12	1100	0000	0000	0111
lilas IR	13	1101	0111	0000	0111
turquoise IR	14	1110	0000	0111	0111
gris	15	1111	0111	0111	0111

TABLEAU II-2/T.101

Jeu des valeurs par défaut II: Valeurs par défaut correspondant à la syntaxe de données II

<i>Valeurs par défaut des limites</i>	
– Dimensions de l'écran	– 1,1
– Limite de la palette	– 32
– Sous-zone de présentation	– non applicable
– Contraintes du mode caractère	– 40, 24
– Limite des coordonnées-polygone	– 128
– Limite des coordonnées-polycourbé	– non applicable
– Résolution de présentation	– (?)
– Limite de mémoire des macro-segments	– (?)
– Limite de mémoire du JCDR	– 2048 multiplats
– Codage direct des couleurs	– 8 couleurs
<i>Valeurs par défaut des paramètres de présentation</i>	
position de texte courante	– coin gauche supérieur
couleur de forme courante	– blanche
couleur auxiliaire courante	– transparente
état de soulignement	– arrêt
état de clignotement	– arrêt
état de taille des caractères de base	– 1/40, 5/128, 40 caractères/24 rangées
état de masquage	– arrêt
état de case avec inversion de caractères	– arrêt
état de marquage des caractères	– arrêt
état de protection de l'écran	– non protégé
état de commande d'affichage	– défilement implicite activé. Pas de zone de défilement définie
état de commande d'appareil	– appareil d'affichage = marche appareil auxiliaire = arrêt appareil d'enregistrement = arrêt appareil d'impression sur papier = arrêt
état de commande de curseur	– arrêt
état de la commande 1 de présentation géométrique	– non applicable
état de la commande 2 de présentation géométrique	
– Rectangle de découpage	0,0; 0,0 et 1,0; 1,0
– mode d'ajournement	dès que possible
– Index de couleur de polygone	1
– Style de remplissage	vide
– Index de style de remplissage	1
– Détermination de mise en valeur	arrêt
– régénération implicite	permise
– Type de trait	plein
– Facteur d'échelle de largeur de trait	1,0
– Facteur d'échelle de taille de marque	1,0
– Type de marque	astérisque
– Vecteurs d'orientation de motif	0,0; 1,0 et 1,0; 0,0
– Index de couleur de ligne brisée	1
– Index de couleur de polymarque	1
– indicateur de régénération	ajournement
– Poste de travail-clôture	le plus grand carré qui tient dans la zone d'affichage
– Poste de travail-fenêtre	0,0; 0,0 et 1,0; 1,0
– identificateur de poste de travail	0
état de texte de présentation géométrique	
– Facteur d'agrandissement des caractères	1,0
– Approche (continue) des caractères	0,0
– Rotation (continue) des caractères	0,0; 7,0/320,0 et 7,0/320,0; 0,0
– Alignement des caractères	normal, normal
– Index de couleur de texte	1

TABLEAU II-2/T.101 (suite)

– Rendu des caractères	chaîne
– Sens d'écriture	droite
Commande de motif plein	
– Représentation de motif	
delta-x	1
delta-y	1
données relatives aux carreaux de motif	1
index de motif	1
état de définition de JCDR	– non défini
état de définition macro	– non applicable
état de motif de structure	– ?
état de mémoire de la partie musicale	– non applicable
état de configuration d'animation	– non applicable
état de commande de segment	
– Priorité de segment	0,0
– Matrice de transformation de segment	1,0; 0,0; 0,0
– visibilité de segment	visible

*Palette par défaut – Syntaxe de données II*

	Adresse de palette		Valeur de couleur		
			R	V	B
noir	0	00000	000000	000000	000000
rouge	1	00001	111111	000000	000000
vert	2	00010	000000	111111	000000
jaune	3	00011	111111	111111	000000
bleu	4	00100	000000	000000	111111
lilas	5	00101	111111	000000	111111
turquoise	6	00110	000000	111111	111111
blanc	7	00111	111111	111111	111111
transparent	8	01000	---	---	---
rouge IR	9	01001	011111	000000	000000
vert IR	10	01010	000000	011111	000000
jaune IR	11	01011	011111	011111	000000
bleu IR	12	01100	000000	000000	011111
lilas IR	13	01101	011111	000000	011111
turquoise IR	14	01110	000000	011111	011111
gris	15	01111	011111	011111	011111
noir	16	10000	000000	000000	000000
rouge	17	10001	111111	000000	000000
vert	18	10010	000000	111111	000000
jaune	19	10011	111111	111111	000000
bleu	20	10100	000000	000000	111111
lilas	21	10101	111111	000000	111111
turquoise	22	10110	000000	111111	111111
blanc	23	10111	111111	111111	111111
noir	24	11000	000000	000000	000000
rouge	25	11001	111111	000000	000000
vert	26	11010	000000	111111	000000
jaune	27	11011	111111	111111	000000
bleu	28	11100	000000	000000	111111
lilas	29	11101	111111	000000	111111
turquoise	30	11110	000000	111111	111111
blanc	31	11111	111111	111111	111111

TABLEAU II-3/T.101

## Jeu des valeurs par défaut III: Valeurs par défaut correspondant à la syntaxe de données III

<i>Valeurs par défaut des limites</i>	
– Dimensions de l'écran	– 0,9999; 0,78125
– Limite de la palette	– 16
– Sous-zone de présentation	– plein écran
– Contraintes du mode caractère	– 40, 20
– Limite des coordonnées-polygone	– 256
– Limite des coordonnées-polycourbe	– 256
– Résolution de présentation	– 256, 200 (nominale)
– Limite de mémoire des macro-segments	– 3072 multipliets
– Limite de mémoire du JCDR	– 3072 multipliets (en partage avec macro)
<i>Valeurs par défaut des paramètres de présentation</i>	
position de texte courante	– coin gauche inférieur
couleur de forme courante	– couleur = blanche mode = direct
couleur auxiliaire courante	– aucune
état de soulignement	– arrêt
état de clignotement	– arrêt
état de taille des caractères de base	– dx = 1/40, dy = 1/128
état de masquage	– non applicable
état de case avec inversion de caractères	– caractères = taille normale inversion = arrêt case (transparente) = arrêt
état de marquage des caractères	– non applicable
état de protection de l'écran	– protégé
état de commande d'affichage	– défilement = arrêt
état de commande d'appareil	– non applicable
état de commande de curseur	– arrêt (invisible)
état de la commande 1 de présentation géométrique	– structure du trait = plein motif de structure = plein gabarit de structure = 1/40, 5/128 mise en valeur = arrêt élément d'image logique = 0,0
état de la commande 2 de présentation géométrique	– non applicable
état d'attente	– aucun
état général du texte	– angle de rotation des caractères = 0° sens d'écriture = droite
approche des caractères = 1	– curseur = souligner interrangée = espace simple
état du sens d'écriture	– non applicable
état de texte de présentation géométrique	– non applicable
état de définition de JCDR	– non défini
état de définition macro	– non défini
état de motif de structure	– non défini
état de mémoire de la partie musicale	– non applicable
état de configuration d'animation	– non applicable
état de configuration du poste de travail	– non applicable

TABLEAU H-3/T.101 (suite)

<i>Palette par défaut – Syntaxe de données III</i>					
	Adresse de palette		Valeur de couleur		
			R	V	B
noir nominal	0	0000	000	000	000
	1	0001	001	001	001
gris	2	0010	010	010	010
	3	0011	011	011	011
	4	0100	100	100	100
blanc nominal	5	0101	101	101	101
	6	0110	110	110	110
	7	0111	111	111	111
teintes	8	1000	000	000	111
	9	1001	000	101	111
	10	1010	000	111	100
	11	1011	010	111	000
	12	1100	111	111	000
	13	1101	111	010	000
	14	1110	111	000	100
	15	1111	101	000	111



## 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	Transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	RGT et maintenance des réseaux: systèmes de transmission, de télégraphie, de télécopie, circuits téléphoniques et circuits loués internationaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
<b>Série T</b>	<b>Terminaux des services télématiques</b>
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	Réseaux de données et communication entre systèmes ouverts
Série Y	Infrastructure mondiale de l'information et protocole Internet
Série Z	Langages et aspects informatiques généraux des systèmes de télécommunication