



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

**UIT-T**

SECTEUR DE LA NORMALISATION  
DES TÉLÉCOMMUNICATIONS  
DE L'UIT

**X.746**

(02/2000)

SÉRIE X: RÉSEAUX DE DONNÉES ET  
COMMUNICATION ENTRE SYSTÈMES OUVERTS  
Gestion OSI – Fonctions de gestion et fonctions ODMA

---

**Technologies de l'information – Interconnexion  
des systèmes ouverts – Gestion-systèmes:  
Fonction de programmation**

Recommandation UIT-T X.746

---

RECOMMANDATIONS UIT-T DE LA SÉRIE X  
**RÉSEAUX DE DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS**

<b>RÉSEAUX PUBLICS DE DONNÉES</b>	
Services et fonctionnalités	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalisation et commutation	X.50–X.89
Aspects réseau	X.90–X.149
Maintenance	X.150–X.179
Dispositions administratives	X.180–X.199
<b>INTERCONNEXION DES SYSTÈMES OUVERTS</b>	
Modèle et notation	X.200–X.209
Définitions des services	X.210–X.219
Spécifications des protocoles en mode connexion	X.220–X.229
Spécifications des protocoles en mode sans connexion	X.230–X.239
Formulaires PICS	X.240–X.259
Identification des protocoles	X.260–X.269
Protocoles de sécurité	X.270–X.279
Objets gérés des couches	X.280–X.289
Tests de conformité	X.290–X.299
<b>INTERFONCTIONNEMENT DES RÉSEAUX</b>	
Généralités	X.300–X.349
Systèmes de transmission de données par satellite	X.350–X.369
Réseaux à protocole Internet	X.370–X.399
<b>SYSTÈMES DE MESSAGERIE</b>	<b>X.400–X.499</b>
<b>ANNUAIRE</b>	<b>X.500–X.599</b>
<b>RÉSEAUTAGE OSI ET ASPECTS SYSTÈMES</b>	
Réseautage	X.600–X.629
Efficacité	X.630–X.639
Qualité de service	X.640–X.649
Dénomination, adressage et enregistrement	X.650–X.679
Notation de syntaxe abstraite numéro un (ASN.1)	X.680–X.699
<b>GESTION OSI</b>	
Cadre général et architecture de la gestion-systèmes	X.700–X.709
Service et protocole de communication de gestion	X.710–X.719
Structure de l'information de gestion	X.720–X.729
<b>Fonctions de gestion et fonctions ODMA</b>	<b>X.730–X.799</b>
<b>SÉCURITÉ</b>	<b>X.800–X.849</b>
<b>APPLICATIONS OSI</b>	
Engagement, concomitance et rétablissement	X.850–X.859
Traitement transactionnel	X.860–X.879
Opérations distantes	X.880–X.899
<b>TRAITEMENT RÉPARTI OUVERT</b>	<b>X.900–X.999</b>

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

**TECHNOLOGIES DE L'INFORMATION – INTERCONNEXION DES SYSTÈMES  
OUVERTS – GESTION-SYSTÈMES: FONCTION DE PROGRAMMATION**

**Résumé**

La présente Recommandation | Norme internationale définit la fonction de programmation. Cette fonction est une fonction de gestion-systèmes qui peut être utilisée par le processus d'application dans un environnement de gestion centralisée ou répartie pour échanger l'information et des commande aux fins de gestion-systèmes.

**Source**

La Recommandation X.746 de l'UIT-T, élaborée par la Commission d'études 4 (1997-2000) de l'UIT-T, a été approuvée le 4 février 2000. Un texte identique est publié comme Norme Internationale ISO/CEI 10164-15.

## AVANT-PROPOS

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

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

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

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

## NOTE

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

## DROITS DE PROPRIÉTÉ INTELLECTUELLE

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

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

© UIT 2002

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

## TABLE DES MATIÈRES

		<i>Page</i>
1	Domaine d'application.....	2
2	Références normatives.....	2
	2.1 Recommandations   Normes internationales identiques.....	2
	2.2 Paires de Recommandations   Normes internationales équivalentes par leur contenu technique.....	2
	2.3 Autres références.....	2
3	Définitions.....	2
	3.1 Définitions du modèle de référence de base.....	2
	3.2 Définitions de la notation de syntaxe abstraite numéro un.....	2
	3.3 Définitions du cadre général de gestion.....	2
	3.4 Définitions du service commun d'information de gestion.....	2
	3.5 Définitions de la présentation générale de la gestion-systèmes.....	2
	3.6 Définitions du modèle d'information de gestion.....	2
	3.7 Définitions supplémentaires.....	2
4	Abréviations.....	2
5	Conventions.....	2
6	Prescriptions.....	2
7	Modèle.....	2
	7.1 Mécanisme de programmation interne.....	2
	7.2 Mécanisme de programmation externe.....	2
	7.3 Modes de programmation.....	2
	7.4 Relations entre objets SO et objets SMO.....	2
8	Définitions génériques.....	2
	8.1 Information de gestion requise pour la programmation interne.....	2
	8.2 Objets gérés.....	2
	8.3 Paquetages.....	2
	8.4 Propriétés des objets SMO.....	2
	8.5 Conformité.....	2
	8.6 Définitions génériques de la fonction de gestion d'objet.....	2
	8.7 Définitions génériques de la fonction de gestion d'état.....	2
	8.8 Définitions génériques de la fonction de gestion de compte rendu.....	2
	8.9 Définitions génériques de la fonction de gestion de test.....	2
	8.10 Définitions génériques de la fonction de récapitulation.....	2
9	Définition de service.....	2
10	Unités fonctionnelles.....	2
11	Protocole et syntaxe abstraite.....	2
	11.1 Objets gérés.....	2
	11.2 Attributs de gestion.....	2
	11.3 Actions de gestion.....	2
	11.4 Notifications de gestion.....	2
12	Relations avec d'autres fonctions.....	2
13	Conformité.....	2
	13.1 Conformité statique.....	2
	13.2 Conformité dynamique.....	2
	13.3 Prescriptions relatives à la déclaration de conformité d'implémentation de gestion.....	2
Annexe A – Définition des informations de gestion.....		2
	A.1 Définition des classes d'objets.....	2
	A.1.1 Définition de l'objet programmeur.....	2
	A.1.2 Définition de l'objet programmeur journalier.....	2
	A.1.3 Définition de l'objet programmeur hebdomadaire.....	2
	A.1.4 Définition de l'objet programmeur mensuel.....	2

A.1.5	Définition de l'objet programmeur périodique .....	2
A.1.6	Définition de l'objet programmeur d'opérations journalier.....	2
A.1.7	Définition de l'objet programmeur d'opérations hebdomadaire.....	2
A.1.8	Définition de l'objet programmeur d'opérations mensuel.....	2
A.1.9	Définition de l'objet programmeur d'opérations périodique.....	2
A.1.10	Définition de l'objet enregistrement de résultat d'opération.....	2
A.1.11	Définition de l'objet programmeur multiple.....	2
A.1.12	Contrôleur de type de jour.....	2
A.2	Liens de dénomination.....	2
A.2.1	Liens de dénomination de programmeurs.....	2
A.2.2	Liens de dénomination de contrôleur de type de jour.....	2
A.3	Paquetages.....	2
A.3.1	Paquetage programmation journalière multiple.....	2
A.3.2	Paquetage programmation mensuelle multiple.....	2
A.3.3	Paquetage programmation hebdomadaire multiple.....	2
A.3.4	Paquetage programmation périodique.....	2
A.3.5	Paquetage mode de resynchronisation.....	2
A.3.6	Paquetage objets gérés programmés.....	2
A.3.7	Paquetage objet programmeur.....	2
A.3.8	Paquetage programmation d'opérations.....	2
A.3.9	Paquetage notification d'opération.....	2
A.4	Attributs.....	2
A.4.1	Attribut nom de programmeur externe.....	2
A.4.2	Attribut "en service".....	2
A.4.3	Attribut mode de resynchronisation.....	2
A.4.4	Attribut objets gérés programmés.....	2
A.4.5	Attribut identificateur de programmeur.....	2
A.4.6	Attribut données de programmation.....	2
A.4.7	Attribut séquence journalière.....	2
A.4.8	Attribut séquence mensuelle.....	2
A.4.9	Attribut séquence hebdomadaire.....	2
A.4.10	Attribut période de temps.....	2
A.4.11	Attribut spécifications d'opération.....	2
A.4.12	Attribut résultat d'opération.....	2
A.4.13	Attribut instance de contrôleur de type de jour.....	2
A.4.14	Attribut identificateur de contrôleur de type de jour.....	2
A.4.15	Attribut liste de traduction du jour de la semaine.....	2
A.4.16	Valeur initiale du comportement pour le type de jour actuel.....	2
A.5	Notifications.....	2
A.5.1	Notification de résultat d'opération.....	2
A.6	Définitions ASN.1.....	2
Annexe B	– Exemples de valeurs de données de programmation.....	2
B.1	Exemple 1: utilisation de plusieurs valeurs d'index.....	2
B.1.1	Solution avec intervalles.....	2
B.1.2	Solution avec déclenchement.....	2
B.2	Exemple 2: utilisation du type de jour.....	2
B.2.1	Solution avec intervalles.....	2
B.2.2	Solution avec déclenchement.....	2
B.3	Exemple 3: recouvrement d'intervalles, répétition implicite (mensuelle) et divers types de programmeurs.....	2
Annexe C	– Formulaire MCS.....	2
Annexe D	– Formulaire PICS.....	2
Annexe E	– Formulaire MOCS.....	2
Annexe F	– Formulaire MIDS.....	2
Annexe G	– Formulaire MRCS.....	2

## Introduction

La Rec. UIT-T X.746 | ISO/CEI 10164-15 appartient à la famille des Recommandations | Normes internationales traitant de la gestion-systèmes.

- X.730 (1992) | ISO/CEI 10164-1:1993: Fonction de gestion des objets.
- X.731 (1992) | ISO/CEI 10164-2:1993: Fonction de gestion d'états.
- X.732 (1992) | ISO/CEI 10164-3:1993: Attributs relationnels.
- X.733 (1992) | ISO/CEI 10164-4:1992: Fonction de signalisation des alarmes.
- X.734 (1992) | ISO/CEI 10164-5:1993: Fonction de gestion des rapports d'événement.
- X.735 (1992) | ISO/CEI 10164-6:1993: Fonction de commande des registres de consignation.
- X.736 (1992) | ISO/CEI 10164-7:1992: Fonction de signalisation des alarmes de sécurité.
- X.737 (1995) | ISO/CEI 10164-14:1996: Catégories de tests de confiance et de diagnostic.
- X.738 (1993) | ISO/CEI 10164-13:1995: Fonction de récapitulation.
- X.739 (1993) | ISO/CEI 10164-11:1994: Objets et attributs métriques.
- X.740 (1992) | ISO/CEI 10164-8:1993: Fonction de piste de vérification de sécurité.
- X.741 (1995) | ISO/CEI 10164-9:1995: Objets et attributs de contrôle d'accès.
- X.742 (1995) | ISO/CEI 10164-10:1995: Fonction de comptage d'utilisation aux fins de comptabilité.
- X.743 (1998) | ISO/CEI 10164-20:1999: Fonction de gestion du temps.
- X.744 (1996) | ISO/CEI 10164-18:1997: Fonction de gestion de logiciel.
- X.745 (1993) | ISO/CEI 10164-12:1994: Fonction de gestion des tests.
- X.746 (2000) | ISO/CEI 10164-15:2001: Fonction de programmation.
- X.748 (1999) | ISO/CEI 10164-22:2000: Fonction de supervision du temps de réponse.
- X.749 (1997) | ISO/CEI 10164-19:1998: Fonction de gestion de domaine de gestion et de politique de gestion.
- X.750 (1996) | ISO/CEI 10164-16:1997: Fonction de gestion de la connaissance de gestion.
- X.751 (1995) | ISO/CEI 10164-17:1996: Fonction de basculement.
- X.753 (1997) | ISO/CEI 10164-21:1998: Séquenceur de commandes pour la gestion-systèmes.



**NORME INTERNATIONALE  
RECOMMANDATION UIT-T**

**TECHNOLOGIES DE L'INFORMATION – INTERCONNEXION DES SYSTÈMES  
OUVERTS – GESTION-SYSTÈMES: FONCTION DE PROGRAMMATION**

**1 Domaine d'application**

La présente Recommandation | Norme internationale définit la fonction de programmation, qui est une fonction de gestion-systèmes pouvant être utilisée par un processus d'application dans un environnement de gestion centralisée ou répartie pour échanger des informations et des commandes aux fins de la gestion-systèmes, selon la définition de la Rec. CCITT X.700 | ISO/CEI 7498-4. La présente Recommandation | Norme internationale, qui se positionne dans la couche Application de la Rec. UIT-T X.200 | ISO/CEI 7498-1, est définie selon le modèle fourni par l'ISO/CEI 9545. Le rôle des fonctions de gestion-systèmes est décrit dans la Rec. CCITT X.701 | ISO/CEI 10040.

La présente Recommandation | Norme internationale:

- identifie un ensemble de prescriptions satisfaites par la fonction;
- fournit un modèle de programmation;
- spécifie les prescriptions de gestion de la fonction et la manière de les mettre en œuvre en spécifiant des objets gérés et les comportements de ceux-ci;
- définit les prescriptions de conformité auxquelles doivent obéir les implémentations de la présente Recommandation | Norme internationale;
- définit les objets gérés.

La présente Recommandation | Norme internationale ne définit pas:

- la manière selon laquelle l'utilisateur de la fonction de programmation doit accomplir la gestion;
- la nature de toute implémentation prévue pour assurer la fonction de programmation;
- la nature de toute interaction qui résulterait de l'utilisation de la fonction de programmation;
- les interactions qui résulteraient de l'utilisation simultanée de plusieurs fonctions de programmation;
- les situations où il convient d'utiliser la fonction de programmation;
- les services nécessaires à l'établissement d'une association de gestion et à sa libération normale ou anormale.

**2 Références normatives**

Les Recommandations et Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation | Norme internationale. Au moment de la publication, les éditions indiquées étaient en vigueur. Toutes Recommandations et Normes sont sujettes à révision et les parties prenantes aux accords fondés sur la présente Recommandation | Norme internationale sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur. Le Bureau de la normalisation des télécommunications de l'UIT tient à jour une liste des Recommandations UIT-T en vigueur.

**2.1 Recommandations | Normes internationales identiques**

- Recommandation UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: le modèle de référence de base.*
- Recommandation UIT-T X.210 (1993) | ISO/CEI 10731:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: conventions pour la définition des services de l'interconnexion de systèmes ouverts.*
- Recommandation UIT-T X.680 (1997) | ISO/CEI 8824-1:1998, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- Recommandation UIT-T X.681 (1997) | ISO/CEI 8824-2:1998, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des objets informationnels.*

## ISO/CEI 10164-15:2002 (F)

- Recommandation UIT-T X.682 (1997) | ISO/CEI 8824-3:1998, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des contraintes.*
- Recommandation UIT-T X.690 (1997) | ISO/CEI 8825-1:1998, *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives.*
- Recommandation UIT-T X.691 (1997) | ISO/CEI 8825-2:1998, *Technologies de l'information – Règles de codage ASN.1: spécification des règles de codage compact.*
- Recommandation CCITT X.701 (1992) | ISO/CEI 10040:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Aperçu général de la gestion-systèmes.*
- Recommandation UIT-T X.710 (1997) | ISO/CEI 9595:1998, *Technologies de l'information – Interconnexion des systèmes ouverts – Service commun d'information de gestion.*
- Recommandation CCITT X.721 (1992) | ISO/CEI 10165-2:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: définition des informations de gestion.*
- Recommandation CCITT X.722 (1992) | ISO/CEI 10165-4:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: directives pour la définition des objets gérés.*
- Recommandation UIT-T X.724 (1993) | ISO/CEI 10165-6:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Structure de l'information de gestion: spécifications et directives pour l'établissement des formulaires de déclaration de conformité d'instance associés à la gestion OSI.*
- Recommandation CCITT X.730 (1992) | ISO/CEI 10164-1:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de gestion des objets.*
- Recommandation CCITT X.731 (1992) | ISO/CEI 10164-2:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de gestion d'états.*
- Recommandation CCITT X.734 (1992) | ISO/CEI 10164-5:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de gestion des rapports d'événement.*
- Recommandation UIT-T X.738 (1993) | ISO/CEI 10164-13:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de récapitulation.*
- Recommandation UIT-T X.739 (1993) | ISO/CEI 10164-11:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: objets et attributs métriques.*
- Recommandation UIT-T X.745 (1993) | ISO/CEI 10164-12:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion-systèmes: fonction de gestion des tests.*

### 2.2 Paires de Recommandations | Normes internationales équivalentes par leur contenu technique

- Recommandation CCITT X.291 (1992), *Cadre général et méthodologie des tests de conformité OSI pour les Recommandations sur les protocoles pour les applications de l'UIT-T – Spécification de suite de tests abstraite.*  
  
ISO/CEI 9646-2:1994, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Cadre général et méthodologie des tests de conformité – Partie 2: Spécification des suites de tests abstraites.*
- Recommandation CCITT X.700 (1992), *Cadre de gestion pour l'interconnexion de systèmes ouverts pour les applications du CCITT.*  
  
ISO/CEI 7498-4:1989, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Partie 4: Cadre général de gestion.*

### 2.3 Autres références

- Recommandation UIT-T M.3100 (1995), *Modèle générique d'information de réseau.*
- ISO/CEI 9545:1994, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Structure de la couche application.*

### 3 Définitions

Pour les besoins de la présente Recommandation | Norme internationale, les définitions suivantes s'appliquent.

#### 3.1 Définitions du modèle de référence de base

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. UIT-T X.200 | ISO/CEI 7498-1:

- a) système ouvert;
- b) gestion-systèmes.

#### 3.2 Définitions de la notation de syntaxe abstraite numéro un

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. UIT-T X.680 | ISO/CEI 8824-1:

- a) identificateur d'objet.

#### 3.3 Définitions du cadre général de gestion

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. CCITT X.700 | ISO/CEI 7498-4:

- a) objet géré.

#### 3.4 Définitions du service commun d'information de gestion

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. UIT-T X.710 | ISO/CEI 9595:

- a) attribut;
- b) service commun d'information de gestion.

#### 3.5 Définitions de la présentation générale de la gestion-systèmes

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. CCITT X.701 | ISO/CEI 10040:

- a) agent;
- b) classe d'objets gérés;
- c) gestionnaire;
- d) notification;
- e) opérations de gestion-systèmes.

#### 3.6 Définitions du modèle d'information de gestion

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. CCITT X.720 | ISO/CEI 10165-1:

- a) action;
- b) classe effective;
- c) comportement;
- d) caractéristique;
- e) paquetage conditionnel;
- f) héritage;
- g) instanciation;
- h) paquetage obligatoire;
- i) corrélation de nom;

## ISO/CEI 10164-15:2002 (F)

- j) paquetage (de propriétés);
- k) sous-classe;
- l) hyperclasse.

### 3.7 Définitions supplémentaires

**3.7.1 programmation apériodique:** mode de programmation qui pilote le déclenchement d'activités à des instants déterminés spécifiés au sein d'instances spécifiées d'objet géré.

**3.7.2 programmation d'intervalle:** mode de programmation qui pilote un nombre d'intervalles d'exécution d'activités au sein d'instances spécifiées d'objet géré.

**3.7.3 programmation périodique:** mode de programmation qui pilote le déclenchement répétitif d'activités au sein d'instances spécifiées d'objet géré.

**3.7.4 programmation:** méthode de pilotage des instants d'exécution d'une activité programmée au sein d'un objet géré ou représentée par un tel objet.

**3.7.5 objet géré programmé (SMO, *scheduled managed object*):** objet géré dont les activités doivent être programmées.

**3.7.6 objet programmeur (SO, *scheduler object*):** objet géré qui définit le mode et les valeurs de programmation à appliquer aux activités de l'objet géré programmé.

**3.7.7 programmation de déclenchement:** mode de programmation qui pilote le déclenchement d'activités au sein d'instances spécifiées d'objet géré.

## 4 Abréviations

Pour les besoins de la présente Recommandation | Norme internationale, les abréviations suivantes sont utilisées:

ASN.1	Notation de syntaxe abstraite numéro un ( <i>abstract syntax notation one</i> )
CMIS	Service commun d'informations de gestion ( <i>common management information service</i> )
ICS	Déclaration de conformité d'implémentation ( <i>implementation conformance statement</i> )
MAPDU	Unité de données protocolaire d'application de gestion ( <i>management application protocol data unit</i> )
MCS	Récapitulatif de conformité de gestion ( <i>management conformance statement</i> )
MOCS	Déclaration de conformité d'objet géré ( <i>managed object conformance statement</i> )
NE	Élément de réseau ( <i>network element</i> )
OC	Classe d'objet ( <i>object class</i> )
OS	Système d'exploitation ( <i>operation system</i> )
SMO	Objet géré programmé ( <i>scheduled managed object</i> )
SO	Objet programmeur ( <i>scheduler object</i> )

## 5 Conventions

Les formulaires ICS spécifiés dans la présente Recommandation | Norme internationale (voir les Annexes C à G) utilisent les notations communes définies dans la Rec. CCITT X.291 | ISO/CEI 9646-2 et dans la Rec. CCITT X.296 | ISO/CEI 9646-7.

## 6 Prescriptions

Les prescriptions à satisfaire en termes de fonctionnalité sont les suivantes:

- fournir une fonction capable de programmer conformément à un programme unique une multiplicité d'activités au sein de multiples objets gérés;
- pouvoir spécifier la durée pendant laquelle un programme est actif;
- définir en termes d'heure effective d'horloge de 24 heures les instants de départ et de fin des programmes pilotant l'intervalle d'opération d'une activité au sein d'un objet géré;
- fournir une fonction capable de programmer le déclenchement périodique ou apériodique d'une activité;
- assurer que les informations de programmation communiquées à l'objet programmé sont indépendantes de l'action effectuée par l'objet en question. Le programmeur peut n'avoir aucune connaissance au sujet de cette action. Il en résulte que les relations entre le programmeur, les informations de programmation et les actions correspondantes existent au niveau de l'objet programmé;
- permettre la coexistence de plusieurs programmeurs indépendants;
- fournir une programmation basée sur le type de jour (par exemple week-end, Noël ou vacances).

### La programmation d'intervalle doit:

- fournir une fonction qui pilote les activités programmées d'un ou de plusieurs objets gérés;
- fournir un programme configurable se répétant sur une période de temps spécifiée, la période de temps spécifiée pouvant être un jour, une semaine ou un mois;
- fournir un nombre d'intervalles définis par l'utilisateur ainsi que les instants de départ et de fin de chaque intervalle à l'intérieur de la période spécifiée;
- permettre le chevauchement des intervalles. Il en résulte que des règles de priorité sont nécessaires pour déterminer l'intervalle actif.

### La programmation de déclenchement doit:

- fournir une fonction qui pilote le déclenchement d'une activité d'un ou de plusieurs objets gérés;
- fournir une période configurable pour les répétitions du déclenchement;
- fournir une liste d'instant de déclenchement définie par l'utilisateur.

## 7 Modèle

La programmation peut être modélisée soit comme un objet géré distinct, soit comme une partie de l'objet géré dont l'opération ou l'activité est à programmer.

Des caractéristiques du pilotage d'un programme peuvent être importées dans une classe d'objets gérés ou peuvent être définies comme un objet géré distinct. Ces deux manières de définir la programmation d'un objet géré sont nommées respectivement programmation interne et programmation externe. La présente Recommandation | Norme internationale décrit des modèles pour les deux modes de programmation interne et externe.

La présente Recommandation | Norme internationale décrit également dans le paragraphe 7.3 ci-dessous quatre modes de programmation: la programmation d'intervalle, la programmation de déclenchement (périodique et apériodique) et la programmation par index (programmeur multiple). Chacun de ces modes de programmation peut être utilisé avec les mécanismes de programmation interne ou externe.

Les activités pouvant être pilotées par la programmation sont définies comme parties de la classe d'objets gérés programmés (SMO, *scheduled managed object*). Il est nécessaire que l'objet SMO contienne des caractéristiques relatives à ces activités programmées.

### 7.1 Mécanisme de programmation interne

Il convient de définir un mécanisme de programmation au sein d'une classe d'objets gérés si ce mécanisme ne risque pas d'être modifié ultérieurement et si l'objet géré doit être programmé d'une manière individuelle. Le mécanisme de programmation peut être défini au sein d'une classe d'objets gérés par l'inclusion des composants de programmation adéquats (attributs et comportements par exemple). Si plus d'un mode de programmation est défini au sein d'une classe d'objets gérés, les conditions d'instanciation de chaque mode de programmation doivent être définies dans la définition de la classe d'objets gérés.

Quand le mécanisme de programmation est défini au sein de l'objet dont l'activité est programmée, aucun objet supplémentaire n'est nécessaire et la programmation peut être manipulée par l'utilisation des opérations de gestion-systèmes. Toutefois, si de multiples activités doivent être programmées au sein d'un objet géré en utilisant ce mécanisme, des caractéristiques de programmation distinctes sont requises pour chaque activité.

Les caractéristiques de programmation de chaque activité peuvent inclure plus d'un mode de programmation (voir § 7.3) et les conditions d'instanciation de chaque mode doivent être définies dans la définition de la classe d'objets gérés.

### 7.2 Mécanisme de programmation externe

Il est avantageux de définir un mécanisme de programmation externe de manière à pouvoir déterminer les programmes indépendamment des objets SMO. De nombreux objets gérés peuvent être ainsi pilotés par un programme unique. Si le programme est fourni par un seul objet programmeur (SO, *scheduler object*), les composants de programmation dans les objets SMO peuvent devenir superflus. Ceci élimine la nécessité de dupliquer et de coordonner des programmes entre objets SMO.

La fonction de programmation est représentée par des objets SO distincts des objets SMO comme le montre la Figure 1. Un objet SO peut piloter les activités d'un nombre quelconque d'objets SMO. Des programmes externes multiples sont possibles pour la même activité. Le paragraphe 7.3 décrit la manière de définir plus d'un mode de programmation pour la même activité.

L'objet programmeur fournit un programme à un objet SMO. Les objets SMO doivent posséder des attributs identifiant l'objet SO qui fournit le programme. Chacun de ces attributs doit être associé avec un comportement qui décrit l'effet du programme sur l'objet SMO. Lorsqu'on utilise l'index SO, il n'est pas nécessaire d'utiliser divers objets SO pour obtenir cet effet.

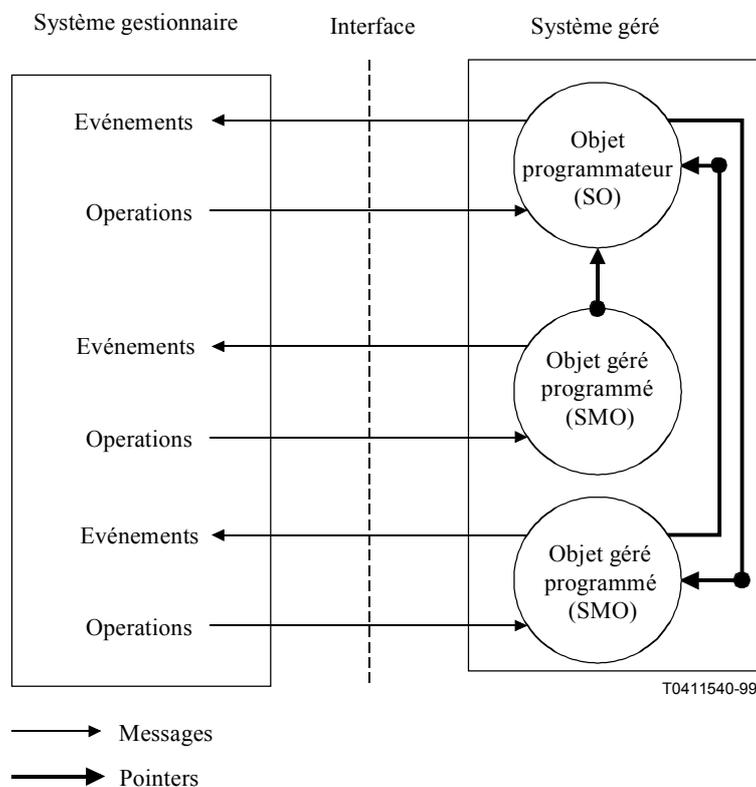


Figure 1 – Modèle d'objet programmeur

### 7.3 Modes de programmation

Il existe quatre modes spécifiques de programmation: la programmation d'intervalle, la programmation de déclenchement, la programmation d'opération et la programmation par index. La présente Recommandation | Norme internationale décrit quatre modes de programmation d'intervalle: la programmation d'intervalle journalière, hebdomadaire, mensuelle et par type de jour (voir § 7.3.1); deux modes de programmation de déclenchement: la programmation périodique et la programmation apériodique (voir § 7.3.2); la programmation d'opération et la programmation par index. Ces modes de programmation sont définis par des paquetages qui peuvent être inclus dans un objet programmeur pour une programmation externe ou dans les objets gérés pour une programmation interne (sauf en ce qui concerne la programmation d'opération).

NOTE – D'autres paquetages de programmation sont définis dans la Rec. CCITT X.734 | ISO/CEI 10164-5.

Si une activité nécessite une combinaison de programmations d'intervalle et de déclenchement, le déclenchement n'est effectif qu'à l'intérieur des intervalles définis par le programme d'intervalle.

#### 7.3.1 Programmation d'intervalle

La programmation d'intervalle est utilisée pour définir un programme pilotant une séquence de transitions de l'activité d'un objet SMO entre états actif et inactif. Le programme peut se répéter de l'une des manières suivantes: à intervalles quotidiens spécifiés pendant un nombre de jours donné, à intervalles quotidiens spécifiés à des jours spécifiés de la semaine pendant un nombre donné de semaines, ou à intervalles quotidiens spécifiés aux jours de mois spécifiés pendant un nombre donné de mois. Chacun de ces modes de programmation d'intervalle journalière, hebdomadaire ou mensuelle est spécifié par la sélection, à l'intérieur de la classe d'objets "programmeur" appropriée, des intervalles du paramètre jour pour les attributs de masque jour, semaine ou mois.

La durée pendant laquelle la programmation d'intervalle affecte l'opération de l'objet SMO peut être pilotée par la spécification des instants de départ et de fin (date et heure).

Les intervalles d'opération sont spécifiés par un ensemble d'instant de début et de fin d'intervalle.

Le fonctionnement des programmeurs d'intervalle peut être suspendue en plaçant l'attribut d'état administratif en position verrouillée et reprise en positionnant l'attribut d'état administratif en position déverrouillée.

#### 7.3.2 Programmation de déclenchement

##### 7.3.2.1 Programmation périodique

La programmation périodique est utilisée pour définir un programme qui déclenche d'une manière répétitive à des intervalles de temps réguliers des activités spécifiées au sein d'instances spécifiées d'objet géré. La durée pendant laquelle les activités spécifiées dans l'objet SMO sont susceptibles d'être déclenchées peut être pilotée par les instants spécifiés de début de durée et de fin de durée (date et heure). Quand un programmeur périodique est créé, soit il se déclenche à l'instant spécifié de début de durée (qui peut être l'instant de création de l'objet), ou il synchronise son premier point de déclenchement sur un instant de déclenchement spécifié. La période est ensuite synchronisée sur le premier point de déclenchement.

Le fonctionnement d'un programmeur peut être suspendu et repris en positionnant son attribut d'état administratif. Deux méthodes de synchronisation des points de déclenchement peuvent être utilisées quand le fonctionnement d'un programmeur est repris: la synchronisation sur l'instant de la période ou le mode de resynchronisation. Si on spécifie la synchronisation sur l'instant de la période, le déclenchement sera toujours synchronisé sur cet instant. Si un mode de resynchronisation a été spécifié dans l'objet SO, selon le mode de resynchronisation sélectionné, le déclenchement peut être synchronisé sur l'instant spécifié de début de la durée, ou il peut être synchronisé sur l'instant de redémarrage de l'objet SO. En cas d'absence de l'instant de synchronisation de la période et du mode de resynchronisation, la période sera toujours synchronisée sur l'instant spécifié de début de durée.

##### 7.3.2.2 Programmation apériodique

Une activité dans un objet géré peut être déclenchée à des instants programmés. Ceci est réalisé en spécifiant un ensemble d'instant de déclenchement plutôt qu'en spécifiant un intervalle de fonctionnement pour cette activité. Ce mécanisme permet de déclencher les activités d'un objet géré à des instants absolus plutôt qu'à intervalles réguliers à partir d'un instant de départ, comme c'est le cas en programmation périodique (voir § 7.3.2.1).

Un programme de déclenchement apériodique peut se répéter de l'une des manières suivantes: à des instants de déclenchement quotidiens spécifiés pendant un nombre de jours donné, à des instants de déclenchement spécifiés à des jours spécifiés de la semaine pendant un nombre donné de semaines, ou à des instants de déclenchement spécifiés à des jours de mois spécifiés pendant un nombre donné de mois. Chacun de ces modes de programmation apériodique journalière, hebdomadaire ou mensuelle est spécifié par la sélection, à l'intérieur de la classe d'objets "programmeur" appropriée, des paramètres "instants de déclenchement" pour les attributs de masque jour, semaine ou mois.

### 7.3.3 Programmation d'opération

Un objet programmeur utilisant la programmation d'opération détermine les opérations exécutées sur les objets SMO conformément à son programme.

Dans ce cas, l'objet SO peut disposer de notifications pour le compte rendu de succès ou d'échec d'exécution des opérations. Un objet programmeur utilisant la programmation d'opération possède des attributs identifiant un programme, les objets SMO qui doivent être programmés ainsi que les opérations et paramètres qui vont être exigés conformément au programme. Quand la notification de résultat est émise, le ou les résultats d'opération doivent contenir les paramètres de la classe d'objets gérés et de l'instance de l'objet.

NOTE – L'envoi de messages entre objets gérés dans un même système, qu'il soit explicitement ou implicitement prévu dans ce modèle, n'implique aucun besoin de test de conformité pour les interactions entre objets.

### 7.3.4 Programmation par index

La programmation par index constitue une extension des modes de programmation antérieurs.

La programmation multiple étend les fonctionnalités de la programmation d'intervalle et de la programmation apériodique:

- pour la programmation d'intervalle, le programmeur multiple permet la transition d'une activité entre plusieurs états (deux ou plus). Chacun de ces états correspond à une valeur de l'index. Le programmeur multiple autorise également des intervalles avec chevauchement;
- pour la programmation apériodique, le programmeur multiple permet le déclenchement d'activités (éventuellement différentes) en fonction de la valeur de l'index. La programmation apériodique du programmeur multiple peut également être utilisée pour la programmation des opérations.

La programmation par index permet de programmer dans les deux cas des activités qui dépendent du type de jour, par exemple de vacances ou de week-end. Un contrôleur de type de jour est utilisé à cet effet pour regrouper les jours en catégories selon leur type, par exemple, le premier janvier est classifié comme "jour spécial 1".

## 7.4 Relations entre objets SO et objets SMO

Un objet SMO peut être programmé par plus d'un objet SO. Pour pouvoir être programmé par un programmeur d'intervalle ou de déclenchement externe, un objet SMO doit posséder un attribut pointant sur l'objet SO (l'attribut nom de programmeur externe). D'une manière optionnelle, l'objet SO peut avoir un attribut pointant sur l'objet SMO (attribut objets gérés programmés). Des objets SMO possédant de multiples activités à programmer doivent avoir un attribut pointant sur l'objet SO approprié associé à chaque activité. Un objet SO unique peut fournir un programme pour de nombreux objets SMO. Voir Figure 1.

Si un objet SMO est supprimé, l'entrée correspondant à cet objet dans l'attribut objets gérés programmés du ou des objets SO correspondants sera supprimée. L'objet SO continuera à exister même s'il ne reste plus d'entrées dans l'attribut objets gérés programmés. Si l'objet SO est supprimé, les activités de l'objet SMO doivent être dans l'état défini par le comportement de l'objet SMO.

Les changements de l'état administratif et opérationnel de l'objet SMO n'affectent pas l'objet SO. Si l'état administratif de l'objet SO passe en verrouillé ou si l'état opérationnel passe en hors service, l'état de l'activité dans l'objet SMO devient inactif. Cet état peut être représenté par un attribut de l'objet SMO associé à cette activité. Si l'état administratif de l'objet SO passe en déverrouillé ou si l'état opérationnel devient "en service", l'objet SMO est mis dans l'état prévu par le programme défini pour l'objet SO.

La relation entre l'objet SO et l'objet SMO est établie au moment de la création de l'objet SMO ou lorsque l'identificateur de l'objet SO est ajouté à l'attribut nom de programmeur externe d'un objet SMO existant. Quand l'objet SMO est créé avec l'identificateur de l'objet SO inclus dans l'attribut nom de programmeur externe, l'identificateur de l'instance de l'objet SMO est ajouté à l'attribut objets gérés programmés de l'objet SO (si l'instance de l'objet SO le supporte). La relation se termine lorsque l'un des deux objets est détruit comme indiqué ci-dessus, ou lorsque l'identificateur de l'objet SO est retiré de l'attribut objets gérés programmés de l'objet SMO.

## 8 Définitions génériques

### 8.1 Information de gestion requise pour la programmation interne

Il est possible de définir une classe d'objets gérés incluant un mécanisme de programmation en important le paquetage de programmation approprié (programmation périodique, programmation journalière, programmation hebdomadaire, programmation journalière multiple, programmation hebdomadaire multiple, ou programmation mensuelle multiple) dans la définition de la classe d'objets gérés et en liant ce paquetage avec les activités appropriées dans la clause de comportement. (Les paquetages de programmation journalière et hebdomadaire sont définis dans la Rec. CCITT X.734 | ISO/CEI 10164-5.)

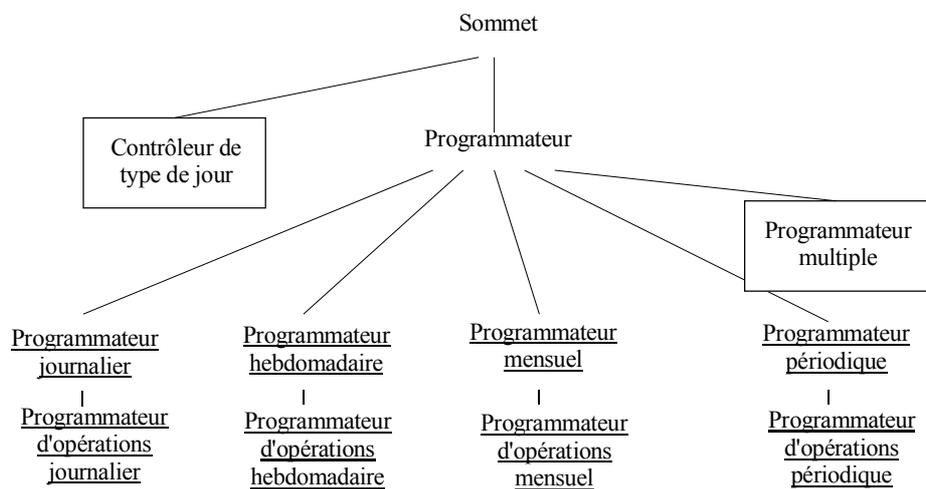
### 8.2 Objets gérés

La présente Recommandation | Norme internationale définit un ensemble de classes d'objets gérés de programmation. La Figure 2 montre la structure d'héritage de ces classes d'objets gérés.

Les classes d'objets gérés présentées dans le Tableau 1 sont définies pour satisfaire aux exigences des modes de programmation décrits au § 7.3.

**Tableau 1 – Modes de programmation et classes objet géré**

Mode de programmation	Classe(s) d'objets gérés
Programmation d'intervalle	Programmeur journalier, programmeur hebdomadaire, programmeur mensuel
Programmation périodique (déclencheur)	Programmeur périodique
Programmation aperiodique (déclencheur)	Programmeur journalier, programmeur hebdomadaire, programmeur mensuel
Programmation d'opération	Programmeur d'opération journalier, programmeur d'opération hebdomadaire, programmeur d'opération mensuel, programmeur d'opération périodique
Programmation par index et par type de jour	Contrôleur de type de jour, programmeur multiple



T0411550-99

NOTE – Les objets susceptibles d'être instanciés sont soulignés.

**Figure 2 – Structure d'héritage de la classe programmeur**

#### 8.2.1 Programmeur

##### 8.2.1.1 Aperçu général

La classe d'objets "programmeur" est une hyperclasse à partir de laquelle sont dérivées les autres classes d'objets programmeur.

### **8.2.1.2 Paquetages du programmeur**

La classe d'objets gérés programmeur contient les paquetages obligatoires suivants:

- paquetage objet programmeur;
- paquetage "durée", défini dans la Rec. CCITT X.734 | ISO/CEI 10164-5.

La classe d'objets gérés programmeur contient le paquetage conditionnel suivant:

- paquetage objets gérés programmés.

## **8.2.2 Programmeur journalier**

### **8.2.2.1 Aperçu général**

La classe d'objets "programmeur journalier" est une sous-classe de la classe d'objets "programmeur". Elle est utilisée pour programmer les intervalles d'activité ou le déclenchement aperiodique d'une activité d'un objet SMO sur une base journalière.

### **8.2.2.2 Paquetages du programmeur journalier**

La classe d'objets gérés programmeur journalier contient le paquetage obligatoire suivant:

- paquetage programmation journalière multiple.

## **8.2.3 Programmeur hebdomadaire**

### **8.2.3.1 Aperçu général**

La classe d'objets "programmeur hebdomadaire" est une sous-classe de la classe d'objets "programmeur". Elle est utilisée pour programmer les intervalles d'activité ou le déclenchement aperiodique d'une activité d'un objet SMO sur une base hebdomadaire.

### **8.2.3.2 Paquetages du programmeur hebdomadaire**

La classe d'objets gérés programmeur hebdomadaire contient le paquetage obligatoire suivant:

- paquetage "programmeur hebdomadaire".

## **8.2.4 Programmeur mensuel**

### **8.2.4.1 Aperçu général**

La classe d'objets "programmeur mensuel" est une sous-classe de la classe d'objets "programmeur". Elle est utilisée pour programmer des intervalles d'activité ou le déclenchement aperiodique d'une activité d'un objet SMO sur une base mensuelle.

### **8.2.4.2 Paquetages du programmeur mensuel**

La classe d'objets gérés programmeur mensuel contient le paquetage obligatoire suivant:

- paquetage programmation mensuelle multiple.

## **8.2.5 Programmeur périodique**

### **8.2.5.1 Aperçu général**

La classe d'objets "programmeur périodique" est une sous-classe de la classe d'objets "programmeur". Elle est utilisée pour programmer sur une base périodique régulière le déclenchement d'une activité d'un objet SMO.

### **8.2.5.2 Paquetages du programmeur périodique**

La classe d'objets gérés "programmeur périodique" contient le paquetage obligatoire suivant:

- paquetage programmation périodique.

La classe d'objets gérés "programmeur périodique" contient les paquetages conditionnels suivants qui ne peuvent être présents tous deux à la fois dans une instance d'objet "programmeur périodique":

- paquetage mode de resynchronisation;
- paquetage "synchronisation de période", défini dans la Rec. UIT-T X.738 | ISO/CEI 10164-13.

## 8.2.6 Programmeur d'opération journalier

### 8.2.6.1 Aperçu général

La classe d'objets "programmeur d'opération journalier" est une sous-classe de la classe d'objets "programmeur journalier". Elle est utilisée pour programmer, sur un objet SMO et sur une base journalière, des opérations get, set et action.

### 8.2.6.2 Paquetages du programmeur d'opération journalier

La classe d'objets gérés "programmeur d'opération journalier" contient le paquetage obligatoire suivant:

- paquetage programmation d'opérations.

La classe d'objets gérés "programmeur d'opération journalier" contient le paquetage conditionnel suivant:

- paquetage résultat d'opération.

Le paquetage objets gérés programmés hérité de la classe d'objets "programmeur" ne doit pas être instancié pour cette classe d'objets. La valeur des instants de déclenchement pour l'attribut "séquence de jours" doit être spécifiée.

## 8.2.7 Programmeur d'opération hebdomadaire

### 8.2.7.1 Aperçu général

La classe d'objets "programmeur d'opération hebdomadaire" est une sous-classe de la classe d'objets "programmeur hebdomadaire". Elle est utilisée pour programmer, sur un objet SMO et sur une base hebdomadaire, des opérations get, set et action.

### 8.2.7.2 Paquetages du programmeur d'opération hebdomadaire

La classe d'objets gérés "programmeur d'opération hebdomadaire" contient le paquetage obligatoire suivant:

- paquetage programmation d'opérations.

La classe d'objets gérés "programmeur d'opération hebdomadaire" contient le paquetage conditionnel suivant:

- paquetage résultat d'opération.

Le paquetage objets gérés programmés hérité de la classe d'objets "programmeur" ne doit pas être instancié pour cette classe d'objets. La valeur des instants de déclenchement pour l'attribut séquence hebdomadaire doit être spécifiée.

## 8.2.8 Programmeur d'opération mensuel

### 8.2.8.1 Aperçu général

La classe d'objets "programmeur d'opération mensuel" est une sous-classe de la classe d'objets "programmeur mensuel". Elle est utilisée pour programmer, sur un objet SMO et sur une base mensuelle, des opérations get, set et action.

### 8.2.8.2 Paquetages du programmeur d'opération mensuel

La classe d'objets gérés "programmeur d'opération mensuel" contient le paquetage obligatoire suivant:

- paquetage programmation d'opérations.

La classe d'objets gérés "programmeur d'opération mensuel" contient le paquetage conditionnel suivant:

- paquetage résultat d'opération.

Le paquetage objets gérés programmés hérité de la classe d'objets "programmeur" ne doit pas être instancié pour cette classe d'objets. La valeur des instants de déclenchement pour l'attribut séquence mensuelle doit être spécifiée.

## 8.2.9 Programmeur d'opération périodique

### 8.2.9.1 Aperçu général

La classe d'objets "programmeur d'opération périodique" est une sous-classe de la classe d'objets "programmeur périodique". Elle est utilisée pour programmer sur une base périodique régulière des opérations sur un objet SMO.

### 8.2.9.2 Paquetages du programmeur d'opération périodique

La classe d'objets gérés programmeur d'opération périodique contient le paquetage obligatoire suivant:

- paquetage programmation d'opérations.

## ISO/CEI 10164-15:2002 (F)

La classe d'objets gérés programmeur d'opération périodique contient le paquetage conditionnel suivant:

- paquetage résultat d'opération.

Le paquetage objets gérés programmés hérité de la classe d'objets "programmeur" ne doit pas être instancié pour cette classe d'objets.

### 8.2.10 Programmeur multiple

#### 8.2.10.1 Aperçu général

Le programmeur de programmation multiple fournit la capacité de piloter des activités qui nécessitent une information plus complète que la simple mise en service ou hors service. Il permet la définition de programmes multiples, associés chacun à une activité. Ces activités correspondent à des valeurs d'index ou sont déclenchées par des opérations. L'association entre une valeur d'index donnée et une activité est définie au sein de l'objet géré dans le cas d'utilisation d'une valeur d'index.

Classe d'objets: programmeur multiple			
Attributs	M/C	Valeurs	Opération
données de programmation	M	ensemble	GET-REPLACE ADD-REMOVE
index par défaut	C	unique	GET-REPLACE REPLACE-WITH-DEFAULT
instance de contrôleur de type de jour	C	ensemble	GET-REPLACE
Notifications			
"ITU-T Recommendation X.746 [4]": operationNotificationPackage	C		

La colonne M/C indique si les informations présentes dans les attributs, actions ou notifications sont obligatoires (M, *mandatory*) ou conditionnelles (C).

La colonne "Valeurs" indique si la valeur de l'attribut est unique ou se constitue d'un ensemble.

La colonne "Opération" indique les opérations possibles sur l'attribut.

Le diagramme de relations entre entités (Figure 3) et l'arborescence de dénomination qui suivent correspondent aux objets gérés programmeur multiple et contrôleur de type de jour.

#### 8.2.10.2 Paquetages du programmeur de programmation multiple

La classe d'objets gérés programmeur de programmation multiple contient le paquetage obligatoire suivant:

- paquetage programmeur multiple.

La classe d'objets gérés programmeur de programmation multiple contient les paquetages conditionnels suivants:

- paquetage programmation d'intervalle;
- paquetage programmation de déclenchement;
- paquetage instance de contrôleur de type de jour.

La classe d'objets gérés programmation multiple contient la notification suivante:

- paquetage notification d'opération.

Le paquetage objets gérés programmés hérité de la classe d'objets "programmeur" ne doit pas être instancié pour cette classe d'objets.

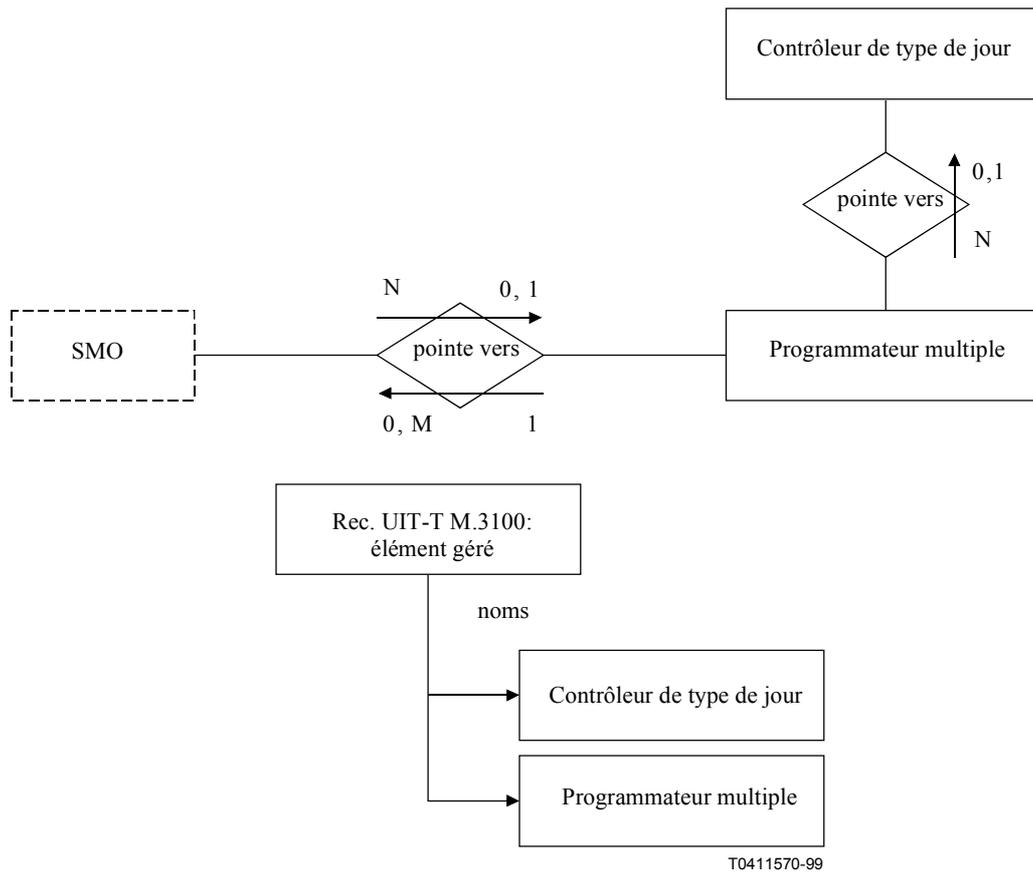


Figure 3 – Diagramme partiel de relations entre entités de programmation et relations de dénomination

### 8.2.11 Contrôleur de type de jour

#### 8.2.11.1 Aperçu général

Le contrôleur de type de jour fournit les informations de gestion nécessaires pour mettre en correspondance une date ou un jour de la semaine avec un type de jour et le type de jour actuellement valide.

Classe d'objets: contrôleur de type de jour			
Attributs	M/C	Valeurs	Opération
identificateur de contrôleur de type de jour	M	unique	GET
type de jour actuel	M	unique	GET
liste de traduction de date	M	ensemble	GET-REPLACE REPLACE-WITH-DEFAULT ADD-REMOVE
liste de traduction du jour de la semaine	M	ensemble	GET-REPLACE REPLACE-WITH-DEFAULT
<b>Notifications</b>			
"ITU-T Recommendation M.3100: (1995)": objectManagementNotificationsPackage	M		

#### 8.2.11.2 Paquetages du contrôleur de type de jour

La classe d'objets gérés contrôleur de type de jour contient le paquetage obligatoire suivant:

- paquetage contrôleur de type de jour.

## ISO/CEI 10164-15:2002 (F)

La classe d'objets gérés contrôleur de type de jour contient la notification obligatoire suivante:

- notification de gestion d'objet.

### 8.3 Paquetages

Les paquetages obligatoires doivent être présents dans toutes les instances d'objet géré appartenant à une classe d'objets gérés, alors que la présence des paquetages conditionnels est déterminée au moment de la création de l'objet géré.

#### 8.3.1 Paquetage objet programmeur

##### 8.3.1.1 Aperçu général

Le paquetage objet programmeur comprend les caractéristiques obligatoires de l'objet programmeur.

##### 8.3.1.2 Attributs du paquetage objet programmeur

Le paquetage objet programmeur possède les attributs suivants:

- a) identificateur du programmeur: cet attribut contient un nom qui identifie une instance de la classe d'objets gérés programmeur (utilisé comme dénomination);
- b) état administratif: cet attribut est défini dans la Rec. CCITT X.731 | ISO/CEI 10164-2;
- c) état opérationnel: cet attribut est défini dans la Rec. CCITT X.731 | ISO/CEI 10164-2.

##### 8.3.1.3 Notifications du paquetage objet programmeur

Le paquetage objet programmeur contient les notifications suivantes:

- changement de valeur d'attribut, défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- changement d'état, défini dans la Rec. CCITT X.731 | ISO/CEI 10164-2;
- création d'objet, définie dans la Rec. CCITT X.731 | ISO/CEI 10164-2;
- suppression d'objet, définie dans la Rec. CCITT X.731 | ISO/CEI 10164-2.

##### 8.3.1.4 Comportement du paquetage objet programmeur

Le paquetage objet programmeur fournit l'attribut de désignation de l'objet SO sous la forme de l'attribut identificateur du programmeur. Il fournit la faculté de suspension et de reprise du fonctionnement de l'objet SO par la modification de l'état administratif. L'attribut état administratif dispose des valeurs "verrouillé", "déverrouillé" et "fermeture en cours". L'attribut état opérationnel dispose des états "en service" et "hors service". L'objet programmeur est actif si son état administratif est déverrouillé et si son état opérationnel est "en service".

Quand l'état administratif passe sur verrouillé durant un intervalle programmé, l'intervalle se termine immédiatement et l'état administratif est mis sur verrouillé. Si l'état administratif passe en fermeture en cours pendant un intervalle programmé, l'intervalle se poursuit jusqu'à son instant normal de fin, instant auquel le programme se termine et l'état administratif est mis sur verrouillé.

Des notifications de changement de valeur d'attribut sont générées quand les attributs instant de début et instant de fin sont modifiés.

Des notifications de changement d'état sont générées quand les attributs d'état administratif et d'état opérationnel sont modifiés.

La notification de création d'objet est générée lors de l'instanciation d'un objet appartenant à une classe d'objets gérés qui contient le paquetage objet programmeur.

La notification de suppression d'objet est générée lors de la suppression d'une instance d'une classe d'objets gérés contenant le paquetage objet programmeur.

#### 8.3.2 Paquetage objets gérés programmés

##### 8.3.2.1 Aperçu général

Le paquetage objets gérés programmés contient une liste des objets SMO qui sont programmés par l'objet SO à l'instant courant.

### 8.3.2.2 Attributs du paquetage objets gérés programmés

Le paquetage objets gérés programmés possède l'attribut suivant:

- objets gérés programmés: cet attribut identifie les objets SMO qui utilisent cet objet SO pour programmer leurs activités et, en option, l'identificateur de l'attribut décrivant l'activité à programmer par l'objet SO. Des notifications de changement de valeur d'attribut sont générées quand l'attribut objets gérés programmés est modifié.

### 8.3.3 Comportement commun des programmeurs d'intervalles

Un programme d'intervalle est constitué par une collection (construite comme un groupe séquence ou séquence-de) de programmes pour un jour donné. Chaque programme journalier donné comporte un ensemble d'intervalles disjoints (sans recouvrement). Chacun de ces intervalles est spécifié comme une séquence d'instant de début et de fin dont les valeurs représentent un temps d'horloge de 24 heures coordonné avec la base de temps spécifiée comme instant de début dans le paquetage "durée". L'instant de fin ne doit pas être antérieur à l'instant de début. Un intervalle peut se poursuivre le jour suivant en spécifiant un instant de fin à 24:00 heures et en spécifiant un intervalle débutant à l'instant 0:00 le jour suivant.

Si une activité dans un objet géré doit être programmée en utilisant une base de zone horaire différente de la zone locale, les valeurs de l'instant de début et de l'instant de fin doivent être spécifiées en utilisant le format d'heure UTC (temps universel coordonné) pour le temps généralisé et les valeurs des intervalles de temps doivent être synchronisées sur le temps spécifié par ces attributs.

### 8.3.4 Paquetage programmation journalière multiple

#### 8.3.4.1 Aperçu général

Le paquetage programmation journalière multiple contient les caractéristiques obligatoires de l'objet programmeur d'intervalle journalier.

#### 8.3.4.2 Attributs du paquetage programmation journalière multiple

Le paquetage programmation journalière multiple possède l'attribut:

- séquence journalière: cet attribut définit pour un jour une séquence d'intervalles de temps en spécifiant les instants de début et de fin d'intervalle. Une valeur (heure = 0, minute = 0) pour l'instant de début signifie le début du jour; la valeur (heure = 0, minute = 0) pour l'instant de fin signifie la fin du jour (24 heures, 0 minute). Si la valeur de cet attribut n'est pas spécifiée dans la requête de création, la valeur attribuée par défaut correspond à un intervalle unique englobant les 24 heures du jour. Un instant de fin d'intervalle heure = 0, minute = 0 permet de prolonger l'intervalle sur le jour suivant. L'intervalle se poursuit si l'instant de départ du premier intervalle du jour suivant est heure = 0, minute = 0, sinon il se termine à la fin du jour.

#### 8.3.4.3 Comportement du paquetage programmation journalière multiple

Le paquetage programmation journalière multiple fournit la faculté de piloter automatiquement une activité au sein d'un objet géré. Il fournit la possibilité de programmer l'opération d'une activité avec une périodicité de 24 heures. Il est possible de définir une séquence continue et répétitive de programmes journaliers. Des intervalles de temps peuvent être spécifiés pour des jours spécifiques d'une séquence.

Le programme identifié par le premier élément de la séquence doit avoir été créé quand l'objet est activé. Chaque programme successif doit être créé tour à tour jusqu'à l'épuisement de la séquence qui est alors répétée.

Les intervalles d'un composant journalier au sein d'une séquence journalière définissent la liste d'intervalles de temps (instants journaliers de début et de fin d'intervalle) pendant lesquels l'activité programmée peut être exécutée. L'activité programmée est inactive pendant les intervalles exclus.

Si la valeur de l'attribut séquence journalière n'est pas spécifiée dans la requête de création, sa valeur sera prise conformément à la spécification par défaut. Ceci implique que l'activité dans l'objet SMO sera activée en permanence.

Des notifications de modification de valeur d'attribut sont générées quand l'attribut séquence journalière est modifié.

### 8.3.5 Paquetage programmation hebdomadaire multiple

#### 8.3.5.1 Aperçu général

Le paquetage programmation hebdomadaire multiple contient les caractéristiques obligatoires de l'objet programmeur d'intervalle hebdomadaire.

### 8.3.5.2 Attributs du paquetage "programmation hebdomadaire multiple"

Le paquetage "programmation hebdomadaire multiple" possède l'attribut suivant:

- séquence hebdomadaire: cet attribut définit une séquence d'intervalles de temps pour chaque jour de la semaine sous la forme d'une séquence de masques de semaine. Chaque masque de semaine est un ensemble de composants de masque dont chacun spécifie un ensemble d'intervalles de temps d'une horloge journalière de 24 heures correspondant à des jours sélectionnés de la semaine.

Le composant jours de la semaine à l'intérieur du type d'attribut définit les jours de la semaine sur lesquels le mécanisme de programmation opère. Si ce composant n'est pas présent dans une requête de création, sa valeur par défaut sera les sept jours de la semaine.

Le composant intervalles de jour à l'intérieur du type d'attribut définit une liste d'intervalles de temps (instants journaliers de début et de fin d'intervalle). Une valeur (heure = 0, minute = 0) pour l'instant de début signifie le début du jour, et la valeur (heure = 0, minute = 0) pour l'instant de fin signifie la fin du jour (24 heures, 0 minute). Si la valeur de cet attribut n'est pas spécifiée dans la requête de création, la valeur attribuée par défaut correspond à un intervalle unique couvrant les 24 heures du jour. Un instant de fin d'intervalle heure = 0, minute = 0 implique que l'intervalle peut se prolonger le jour suivant. L'intervalle se poursuit si l'instant de départ du premier intervalle du jour suivant est heure = 0, minute = 0, sinon il se termine à la fin du jour.

### 8.3.5.3 Comportement du paquetage "programmation hebdomadaire multiple"

Le paquetage "programmation hebdomadaire multiple" fournit la faculté de piloter automatiquement une activité au sein d'un objet géré. Il fournit la possibilité de programmer l'opération d'une activité avec une périodicité d'une semaine. Des intervalles de temps peuvent être spécifiés pour des jours spécifiés de chaque semaine. Il est possible de définir une séquence continue et répétitive de programmes hebdomadaires.

Le programme identifié par le premier élément de la séquence sera implémenté lorsque l'objet est activé. Chacun des programmes suivants sera implémenté tour à tour jusqu'à épuisement de la séquence qui est alors répétée. Un programme pour une semaine unique est constitué d'un ensemble de duplets contenant un élément identifiant le jour de la semaine et un élément identifiant un programme pour un jour unique. Pris dans sa totalité, cet ensemble identifie une collection d'intervalles disjoints répartis sur une semaine complète, le début de la semaine étant fixé le dimanche à 12 heures relativement à la base de temps spécifiée comme instant de départ du paquetage "durée".

Le composant intervalles de jour à l'intérieur de l'attribut définit une liste d'intervalles de temps (instant journalier de début et instant de fin d'intervalle) pour lesquels l'activité programmée peut être exécutée. L'activité programmée est inactive pendant les intervalles exclus.

Si la valeur de l'attribut n'est pas spécifiée dans la requête de création, sa valeur sera prise conformément à la spécification par défaut. Ceci implique que l'activité dans l'objet SMO sera activée en permanence.

Des notifications de modification de valeur d'attribut seront générées quand l'attribut est modifié.

## 8.3.6 Paquetage programmation mensuelle multiple

### 8.3.6.1 Aperçu général

Le paquetage programmation mensuelle multiple contient les caractéristiques obligatoires de l'objet programmeur d'intervalle mensuel.

### 8.3.6.2 Attributs du paquetage programmation mensuelle multiple

Le paquetage programmation mensuelle multiple possède l'attribut:

- séquence mensuelle: cet attribut définit une séquence d'intervalles de temps pour chaque jour du mois sous la forme d'une séquence de masques de mois. Chaque masque de mois est un ensemble de composants de masque dont chacun spécifie un ensemble d'intervalles de temps d'une horloge journalière de 24 heures concernant des jours sélectionnés du mois.

Le composant jours du mois à l'intérieur du type d'attribut séquence mensuelle définit les jours du mois sur lesquels le mécanisme de programmation opère. Cet attribut permet la sélection de jours du mois en avant à partir du premier jour du mois ou en arrière à partir du dernier jour du mois. Le composant est constitué de deux chaînes binaires. Les jours de la première chaîne sélectionnent les jours du mois en partant du premier jour du mois (le premier bit représente le premier jour, etc.). Les jours de la seconde chaîne sélectionnent les jours en partant du dernier jour du mois et en remontant en sens inverse à partir de la fin du mois (le premier bit représente le 30 d'un mois de 30 jours, le second bit représente le 29 de ce mois, etc.). Un jour du mois est sélectionné si le bit qui lui correspond dans l'une ou l'autre chaîne est positionné à 1.

Si ce composant n'est pas présent dans une requête de création, la valeur prise par défaut sera l'ensemble des jours du mois.

Le composant intervalles de jour à l'intérieur du type d'attribut séquence mensuelle définit une liste d'intervalles de temps (instants journaliers de début et de fin d'intervalle). Une valeur (heure = 0, minute = 0) pour l'instant de début signifie le début du jour, et la valeur (heure = 0, minute = 0) pour l'instant de fin signifie la fin du jour (24 heures, 0 minute). Si la valeur de cet attribut n'est pas spécifiée dans la requête de création, la valeur attribuée par défaut correspond à un intervalle unique englobant une période complète de 24 heures pour un jour. Un instant de fin d'intervalle heure = 0, minute = 0 implique que l'intervalle peut se prolonger le jour suivant. L'intervalle se poursuit si l'instant de départ du premier intervalle du jour suivant est heure = 0, minute = 0, sinon il se termine à la fin du jour.

### 8.3.6.3 Comportement du paquetage programmation mensuelle multiple

Le paquetage programmation mensuelle multiple fournit la faculté de piloter automatiquement une activité au sein d'un objet géré. Il fournit la possibilité de programmer l'opération d'une activité avec une périodicité d'un mois. Des intervalles de temps peuvent être spécifiés pour des jours spécifiés de chaque mois. Il est possible de définir une séquence continue et répétitive de programmes hebdomadaires.

Le programme identifié par le premier élément de la séquence doit avoir été créé quand l'objet est activé. Chaque programme successif doit être créé tour à tour jusqu'à l'épuisement de la séquence qui est alors répétée. Un programme pour un mois unique est constitué d'un ensemble de duplets contenant un élément identifiant le jour du mois et un élément identifiant un programme pour un jour unique. Pris dans sa totalité, cet ensemble identifie une collection d'intervalles disjoints répartis sur un mois entier, le début du mois étant fixé au premier du mois à 12 heures relativement à la base de temps spécifiée comme instant de départ du paquetage "durée". Les jours incorrects sont ignorés.

Le composant intervalles de jour à l'intérieur de l'attribut séquence mensuelle définit une liste d'intervalles de temps (instant journalier de début et instant de fin d'intervalle) pour lesquels l'activité programmée peut être exécutée. L'activité programmée est inactive pendant les intervalles exclus.

Si la valeur de l'attribut séquence mensuelle n'est pas spécifiée dans la requête de création, sa valeur sera prise conformément à la spécification par défaut. Ceci implique que l'activité dans l'objet SMO sera activée en permanence.

Des notifications de modification de valeur d'attribut seront générées quand l'attribut séquence mensuelle est modifié.

## 8.3.7 Paquetage programmation périodique

### 8.3.7.1 Aperçu général

Le paquetage programmation périodique contient les caractéristiques obligatoires de l'objet programmeur périodique.

### 8.3.7.2 Attributs du paquetage programmation périodique

Le paquetage programmation périodique possède l'attribut suivant:

- période de temps: cet attribut définit la durée de la période de temps pour le déclenchement périodique d'une activité dans un objet SMO.

### 8.3.7.3 Comportement du paquetage programmation périodique

Le paquetage programmation périodique fournit la faculté de programmer des activités dans un objet SMO sur la base d'un programme défini. Une activité dans un objet SMO sera déclenchée par l'objet programmeur périodique.

Si la valeur de la période de temps n'est pas spécifiée par la requête de création, sa valeur sera prise par défaut à zéro seconde. Ceci signifie que le déclenchement n'aura pas lieu.

L'activité d'un programmeur peut être suspendue et reprise en positionnant son état administratif. Deux méthodes de synchronisation des points de déclenchement peuvent être utilisées quand l'activité est reprise: la synchronisation de période et le mode de resynchronisation. Si un instant de synchronisation de période est spécifié dans l'attribut de synchronisation de période, le déclenchement sera toujours synchronisé sur cet instant. Si l'attribut mode de resynchronisation est présent dans l'objet SO, le déclenchement sera synchronisé sur l'instant de début de la période si le mode de resynchronisation est faux, ou sera synchronisé sur l'instant de reprise si le mode de resynchronisation est vrai.

Si le paquetage "synchronisation de période" et le paquetage "mode de resynchronisation" ne sont pas présents dans l'instance d'objet, la période de temps est synchronisée sur l'instant de début dans le paquetage "durée". Si l'instant de début dans le paquetage "durée" n'est pas spécifié, la période de temps est synchronisée sur l'instant de création de l'objet.

Des notifications de changement de valeur d'attribut sont générées si les attributs période de temps, synchronisation de période ou mode de resynchronisation sont modifiés.

### **8.3.8 Paquetage mode de resynchronisation**

#### **8.3.8.1 Aperçu général**

Le paquetage mode de resynchronisation est utilisé pour spécifier de quelle manière un objet programmeur périodique doit resynchroniser le déclenchement du programme périodique lors de la réactivation d'un objet SO.

#### **8.3.8.2 Attributs du paquetage mode de resynchronisation**

Le paquetage mode de resynchronisation possède l'attribut suivant:

- mode resynchronisation: cet attribut définit de quelle manière la période de temps est définie ou redéfinie lors de l'activation du programmeur périodique (c'est-à-dire quand l'état administratif est mis sur déverrouillé alors que l'état opérationnel est "en service", ou quand l'état opérationnel est mis sur "en service" alors que l'état administratif est sur déverrouillé).

#### **8.3.8.3 Comportement du paquetage mode de resynchronisation**

L'attribut mode de resynchronisation fournit la possibilité de piloter le mode de synchronisation des périodes de déclenchement d'un programmeur périodique lors de la suspension et de l'activation ou réactivation de l'objet géré programmeur. Si la valeur est faux, ceci implique que la période de déclenchement sera synchronisée sur le point de déclenchement initial ou sur les points de déclenchement avant la suspension, selon que l'objet géré programmeur a respectivement été activé ou réactivé. Si la valeur est vrai, ceci implique que, lorsque l'objet géré programmeur a été créé dans l'état suspendu ou placé dans l'état suspendu après sa création (c'est-à-dire verrouillé), il se déclenche à la réactivation et synchronise la période de temps sur l'instant de réactivation.

### **8.3.9 Paquetage synchronisation de période**

Le paquetage synchronisation de période, défini dans la Rec. UIT-T X.738 | ISO/CEI 10164-13, spécifie l'instant de synchronisation pour les périodes. Le début de chaque période est à un instant qui se trouve à un nombre entier de périodes avant ou après l'instant de synchronisation de la période.

### **8.3.10 Paquetage programmation d'opérations**

#### **8.3.10.1 Aperçu général**

Le paquetage programmation d'opérations identifie les opérations spécifiques devant être programmées dans l'objet géré programmé. Un objet SO contenant ce paquetage déterminera l'exécution des opérations spécifiées sur les objets SMO spécifiés en accord avec le programme pris en charge par l'objet SO.

#### **8.3.10.2 Attributs du paquetage programmation d'opérations**

Le paquetage programmation d'opérations possède l'attribut suivant:

- spécifications d'opération: l'attribut spécifications d'opération identifie les opérations spécifiques à programmer dans l'objet géré programmé. Cet attribut du type lecture/écriture et opérationnel (ajouter/enlever) identifie les objets SMO et les opérations à exécuter sur les objets SMO conformément au programme.

### 8.3.11 Paquetage notification d'opération

#### 8.3.11.1 Aperçu général

Le paquetage notification d'opération contient la notification de résultat d'opération qui véhicule les résultats des opérations exécutées sur l'objet SMO.

#### 8.3.11.2 Notifications du paquetage notification d'opération

Le paquetage notification d'opération contient la notification suivante:

- résultat d'opération: la notification de résultat d'opération identifie les instances spécifiques de l'objet SMO et les résultats des opérations qui viennent d'être exécutées sur l'objet géré programmé. Cette information est contenue dans le paramètre résultat d'opération de la notification.

### 8.3.12 Paquetage programmeur multiple

#### 8.3.12.1 Aperçu général

Le paquetage programmeur multiple fournit la faculté de piloter des activités qui nécessitent des informations qui vont au delà de la simple programmation de mise en service et hors service. Il permet la définition de programmes indépendants multiples, dont chacun est associé à une activité.

#### 8.3.12.2 Attributs du paquetage programmeur multiple

Le paquetage programmeur multiple contient les attributs suivants:

- données de programmation: ensemble de programmes et de données de commande d'activité correspondante. Les valeurs de cet attribut sont des ensembles et les opérations permises sont GET, REPLACE, ADD, REMOVE.

### 8.3.13 Paquetage programmation d'intervalle

#### 8.3.13.1 Aperçu général

Le paquetage programmation d'intervalle fournit la faculté de programmer des activités dépendant d'intervalles et de la relation de ces intervalles avec des index. Lorsque les instants de création ou de reprise se trouvent au sein d'une durée définie par le programmeur, les activités au sein de l'objet SMO seront fixées conformément à l'index défini par le programme du programmeur d'intervalle. L'activité au sein de l'objet SMO sera fixée conformément à l'index par défaut lorsque le programmeur d'intervalle est supprimé ou suspendu, ou lorsque l'instant se trouve en-dehors de la durée définie par le programmeur.

#### 8.3.13.2 Attributs du paquetage programmation d'intervalle

Le paquetage programmation d'intervalle contient l'attribut suivant:

- index par défaut: la valeur indiquée par l'index par défaut s'applique lorsque aucun des intervalles n'est valide. Cette éventualité peut se présenter parce que les programmes définis dans les attributs données de programmation sont indépendants.

### 8.3.14 Paquetage programmation de déclenchement

#### 8.3.14.1 Aperçu général

Le paquetage programmation de déclenchement fournit la faculté de programmer des activités sur la base d'un mécanisme de déclenchement. Il est utilisé pour définir des contraintes sur l'attribut données de programmation. Le premier déclenchement d'une activité au sein de l'objet SMO se fera conformément au programme lorsque le déclencheur est créé ou réactivé à un instant qui se situe au sein de la durée définie par le programmeur. L'activité au sein de l'objet SMO ne sera pas déclenchée lorsque le programmeur de déclenchement est supprimé ou suspendu, ou lorsque l'instant se trouve en-dehors de la durée définie par le programmeur.

Les contraintes sur l'attribut données de programmation sont les suivantes:

- la structure TimesOfDayWps contiendra le champ "instants de déclenchement";
- le champ optionnel "priorité" ne sera pas présent.

### **8.3.14.2 Attributs du paquetage programmation de déclenchement**

Le paquetage programmation de déclenchement ne contient pas d'attribut.

### **8.3.15 Paquetage instance de contrôleur de type de jour**

#### **8.3.15.1 Aperçu général**

Le paquetage instance de contrôleur de type de jour fournit la faculté d'identifier l'instance d'un contrôleur de type de jour qui est valide pour l'instance de la classe d'objets programmeur multiple qui la contient.

#### **8.3.15.2 Attributs du paquetage instance de contrôleur de type de jour**

Le paquetage "instance de contrôleur de type de jour" contient l'attribut suivant:

- instance de contrôleur de type de jour: identifie l'instance de la classe d'objets contrôleur de type de jour correspondant au programmeur multiple actuel.

### **8.3.16 Paquetage contrôleur de type de jour**

#### **8.3.16.1 Aperçu général**

Le paquetage contrôleur de type de jour fournit la faculté de définir les informations nécessaires pour mettre en correspondance une date ou un jour de la semaine donné avec un type de jour et contient le type de jour actuellement valable.

#### **8.3.16.2 Attributs du paquetage contrôleur de type de jour**

Le paquetage contrôleur de type de jour contient les attributs suivants:

- identificateur de contrôleur de type de jour: identifie l'instance de la classe d'objets contrôleur de type de jour (nom RDN);
- type de jour actuel: indique le type de jour actuellement valide;
- liste de traduction de date: table contenant les correspondances entre les dates et les valeurs de type de jour;
- liste de traduction du jour de la semaine: table contenant les correspondances entre les jours de la semaine et les valeurs du type de jour.

## **8.4 Propriétés des objets SMO**

### **8.4.1 Paquetages et attributs utilisés dans les objets SMO**

La relation entre l'objet SMO et le programmeur est représentée dans l'objet SMO de l'une des manières suivantes:

- par l'attribut nom de programmeur externe;
- par un attribut dérivé de l'attribut nom de programmeur externe;
- par le paquetage "programmeur externe".

Le paquetage "programmeur externe" est défini dans la Rec. CCITT X.734 | ISO/CEI 10164-5. L'attribut nom de programmeur externe est défini au § 8.4.1.1.

L'état d'une activité peut être spécifié dans un objet SMO par:

- l'état disponibilité défini dans la Rec. CCITT X.731 | ISO/CEI 10164-2; ou
- l'attribut "en service" défini au § 8.4.1.2.

#### **8.4.1.1 Attribut nom de programmeur externe**

L'attribut nom de programmeur externe est incorporé dans une définition d'objet géré quand une activité au sein d'un objet SMO doit être programmée par un programmeur externe. Il spécifie le nom d'un ou de plusieurs objets gérés programmeurs externes qui sont en relation avec une activité dans l'objet SMO. Cette relation implique que l'activité sera contrôlée par le ou les objets programmeurs externes. Si de multiples activités doivent être programmées dans l'objet SMO, des attributs supplémentaires dérivés de cet attribut doivent être inclus dans l'objet.

L'ensemble vide signifie qu'aucun objet programmeur n'est spécifié (par exemple si l'objet SO a été supprimé).

Des notifications de changement d'attribut sont générées si cet attribut est modifié.

### 8.4.1.2 Attribut "en service"

L'attribut "en service", du type lecture seulement, est utilisé pour indiquer l'état d'une activité programmée au sein d'un objet SMO. Cet attribut identifie l'activité spécifique de l'objet SMO auquel le programme doit s'appliquer. Son identificateur fait partie de l'attribut objets gérés programmés de l'objet SO (voir § A.4.4). Quand cet attribut porte la valeur "Vrai" (respectivement Faux), il indique que le programme prévoit que l'activité sera à l'état actif (respectivement inactif). Un attribut "en service" doit être défini en propre par dérivation de l'attribut "en service" pour chacune des activités au sein d'un objet SMO s'il est nécessaire d'en programmer séparément les diverses activités (voir § A.4.2).

### 8.4.1.3 Paquetage "fenêtre requise"

Le paquetage "fenêtre requise" est défini dans la Rec. UIT-T X.745 | ISO/CEI 10164-12. Ce paquetage peut être importé dans un objet SMO s'il est nécessaire de gérer la fenêtre de temps à l'intérieur de laquelle s'exécute une activité.

## 8.4.2 Comportement de l'objet SMO pour la programmation d'intervalle

Chaque instance d'objet programmeur d'intervalle peut piloter un nombre quelconque d'instances d'objets gérés.

Au moment de la création d'un programmeur d'intervalle et de la mise en place de la relation de programmation, les activités des objets SMO seront mises sur "en service" et "hors service" conformément au programme défini par le programmeur d'intervalle pour cet instant particulier. Si le programmeur d'intervalle est créé à un instant qui se trouve à l'extérieur des intervalles de fonctionnement définis par le programme, l'activité au sein de l'objet SMO sera mise hors service. Dans ces conditions, le comportement de chaque activité sera défini par la clause de comportement de la définition de la classe de l'objet SMO. Elle pourra stipuler par exemple que toute activité exécutée à cet instant se poursuivra jusqu'à son achèvement, mais qu'aucune autre activité ne sera lancée.

Quand l'exécution d'un programmeur d'intervalle est suspendue, les activités programmées au sein des objets SMO seront hors service. Si l'activité d'un programmeur d'intervalle est suspendue durant l'un quelconque des intervalles d'exécution définis pour le programme, les activités programmées au sein des objets SMO seront mises hors service. Dans ces conditions, le comportement de chaque activité sera défini par la clause de comportement de la définition de la classe de l'objet SMO. Elle pourra stipuler par exemple que toute activité exécutée à cet instant se poursuivra jusqu'à son achèvement, mais qu'aucune autre activité ne sera lancée.

Un attribut d'état peut être défini pour chaque activité programmée dans un objet SMO afin d'indiquer que cette activité est programmée. Pour une activité programmée par un seul objet SO, si l'état administratif de l'objet SO est positionné sur verrouillé ou si l'état opérationnel de l'objet SO est désactivé, l'attribut d'état pour l'activité sélectionnée dans l'objet SMO est positionné pour indiquer qu'il n'est pas programmé. S'il n'y a qu'une seule activité dans un objet SMO, l'état de disponibilité défini dans la Rec. CCITT X.731 | ISO/CEI 10164-2 peut être utilisé pour indiquer l'état programmé. Quand l'état administratif de l'objet SO est mis sur déverrouillé ou que l'état opérationnel devient "en service", l'attribut état de cette activité est modifié en fonction du programme défini pour l'objet SO. La définition de l'objet SMO doit spécifier comment celui-ci est affecté par les objets SO. Pour un objet SMO avec une seule activité par exemple, l'état de disponibilité est positionné sur "hors service" quand le programme de l'objet SO indique "hors service".

Quand un programmeur d'intervalle est réactivé, les activités programmées au sein des objets SMO seront mises sur "en service" ou "hors service" conformément au programme du programmeur d'intervalle.

Un objet SMO contenant son propre programme périodique peut être programmé par un programmeur d'intervalle. Dans ce cas, le programme périodique de l'objet SMO n'est actif que durant les intervalles spécifiés dans l'objet SO, comme l'indique la Figure 4.

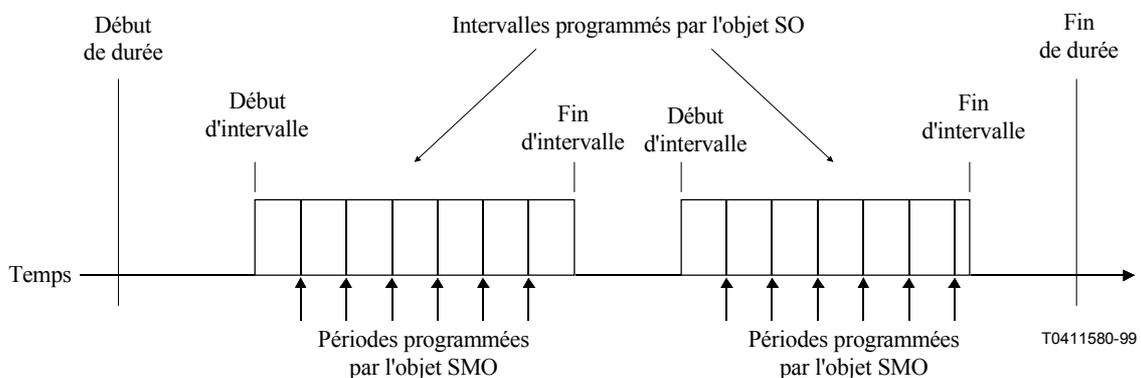


Figure 4 – Exemple de programmation périodique avec intervalle programmé

### 8.4.3 Comportement de l'objet SMO pour la programmation de déclenchement périodique et apériodique

Quand un objet géré "programmeur" du type déclencheur (périodique ou apériodique) est créé et que la relation de programmation est mise en place, les activités programmées des objets SMO seront déclenchées à partir du début du fonctionnement du déclencheur et avec une périodicité ou à des instants de déclenchement conformes à leur programme.

Quand l'activité d'un programmeur périodique ou apériodique est suspendue, les activités programmées au sein des objets SMO ne seront pas déclenchées.

### 8.4.4 Comportement d'un objet SMO pour la programmation par index

L'objet SMO contiendra, en vue de l'utilisation du mécanisme de programmation par index, un attribut de programmation qui met en correspondance chaque valeur d'index avec l'activité adéquate au sein de cet objet SMO. La forme générale de l'attribut est une instruction SET OF SEQUENCE dans laquelle le terme SEQUENCE contient un champ "index" et un autre champ (décrivant l'activité) sous la forme d'une valeur numérique unique ou sous la forme d'une série d'opérations complexes.

Ceci est illustré par l'exemple suivant:

#### OC A (SMO)

```
namingAttributeA
schedulingAttribute
    {
        -- SET OF
        {
            -- SEQUENCE
            index: 1,
            scheduledAttribute: B1
        },
        {
            -- SEQUENCE
            index: 2,
            scheduledAttribute: B2
        },
        {
            -- SEQUENCE
            index: 3,
            scheduledAttribute: B3
        }
    }
activeScheduledAttribute
```

#### OC B (pointed at OC)

namingAttributeB (avec les valeurs B1, B2, B3 possibles)

#### OC multiScheduler

```
.../... from X.746
schedulingData
    {
        -- SET OF
        {
            -- SEQUENCE
            indexOrOperSpec: index: 1,...
```

Cet attribut contient les informations de programmation pour trois valeurs d'index.

## 8.5 Conformité

Les classes d'objets gérés prennent en charge les fonctions définies dans la présente Recommandation | Norme internationale en incorporant la spécification de l'information de gestion par la référence aux squelettes de gestion définis dans l'Annexe A. Le mécanisme de référence est défini dans la Rec. CCITT X.722 | ISO/CEI 10165-4.

## 8.6 Définitions génériques de la fonction de gestion d'objet

La présente Recommandation | Norme internationale utilise les définitions génériques suivantes de la Rec. CCITT X.730 | ISO/CEI 10164-1:

- notification de changement de valeur d'attribut;
- notification de création d'objet;
- notification de suppression d'objet.

### 8.7 Définitions génériques de la fonction de gestion d'état

La présente Recommandation | Norme internationale utilise les définitions génériques suivantes de la Rec. CCITT X.731 | ISO/CEI 10164-2:

- état administratif;
- état opérationnel;
- notification de changement d'état;
- descripteur d'état de disponibilité.

### 8.8 Définitions génériques de la fonction de gestion de compte rendu

La présente Recommandation | Norme internationale utilise les définitions génériques suivantes de la Rec. CCITT X.734 | ISO/CEI 10164-5:

- paquetage "durée";
- paquetage "programmeur externe".

### 8.9 Définitions génériques de la fonction de gestion de test

La présente Recommandation | Norme internationale utilise les définitions génériques suivantes de la Rec. UIT-T X.745 | ISO/CEI 10164-12:

- paquetage "fenêtre temporelle requise".

### 8.10 Définitions génériques de la fonction de récapitulation

La présente Recommandation | Norme internationale utilise les définitions génériques suivantes de la Rec. UIT-T X.738 | ISO/CEI 10164-13:

- paquetage "synchronisation de période".

## 9 Définition de service

La présente Recommandation | Norme internationale ne définit aucun service. Les services utilisés définis dans d'autres spécifications sont les suivants:

- le service PT-EVENT défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service PT-GET défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service PT-SET défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service PT-CREATE défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service PT-DELETE défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service de compte rendu de création d'objet défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service de compte rendu de suppression d'objet défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1;
- le service de compte rendu de changement de valeur d'attribut défini dans la Rec. CCITT X.730 | ISO/CEI 10164-1; et
- le service de compte rendu de changement d'état défini dans la Rec. CCITT X.731 | ISO/CEI 10164-2.

## 10 Unités fonctionnelles

Les unités fonctionnelles suivantes définies dans la Rec. CCITT X.730 | ISO/CEI 10164-1 peuvent être négociées à des fins de gestion des objets programmeurs:

- tous événements;
- pilotage;
- surveillance;
- événements objets.

## ISO/CEI 10164-15:2002 (F)

Les unités fonctionnelles suivantes définies dans la Rec. CCITT X.731 | ISO/CEI 10164-2 peuvent être négociées à des fins de gestion des objets programmeurs:

- compte rendu de changement d'état.

## 11 Protocole et syntaxe abstraite

### 11.1 Objets gérés

La présente Recommandation | Norme internationale définit les objets programmeurs suivants dont la syntaxe est spécifiée dans l'Annexe A:

- programmeur;
- programmeur journalier;
- programmeur hebdomadaire;
- programmeur mensuel;
- programmeur périodique;
- programmeur d'opérations journalier;
- programmeur d'opérations hebdomadaire;
- programmeur d'opérations mensuel;
- programmeur d'opérations périodique;
- résultat d'opération;
- programmeur multiple;
- contrôleur de type de jour.

### 11.2 Attributs de gestion

La présente Recommandation | Norme internationale définit les attributs suivants dont la syntaxe abstraite est spécifiée dans l'Annexe A:

- nom de programmeur externe;
- en service;
- spécification d'opération;
- résultat d'opération;
- mode de resynchronisation;
- objets gérés programmés;
- identificateur de programmeur;
- séquence journalière;
- séquence mensuelle;
- séquence hebdomadaire;
- période de temps;
- données de programmation;
- index par défaut;
- instance de contrôleur de type de jour;
- identificateur de contrôleur de type de jour;
- type de jour actuel;
- liste de traduction de date;
- liste de traduction de week-end.

### 11.3 Actions de gestion

Aucune action spécifique de gestion n'est définie pour cette fonction de gestion-systèmes.

### 11.4 Notifications de gestion

La présente Recommandation | Norme internationale définit la notification suivante, dont la syntaxe abstraite est spécifiée dans l'Annexe A:

- résultat d'opération.

La présente Recommandation | Norme internationale fait référence aux notifications de gestion suivantes:

- changement de valeur d'attribut;
- changement d'état;
- création d'objet;
- suppression d'objet.

## 12 Relations avec d'autres fonctions

La présente Recommandation | Norme internationale utilise des services définis dans la Rec. CCITT X.730 | ISO/CEI 10164-1 à des fins de création et de suppression d'objets gérés, de restitution d'attributs et de notifications de changements d'attributs, la définition de la Rec. CCITT X.731 | ISO/CEI 10164-2 pour la notification de changements d'état, la définition de la Rec. UIT-T X.739 | ISO/CEI 10164-11 pour la syntaxe de l'attribut période de temps. Elle fait également référence à de nombreux objets, paquetages et attributs définis dans la Rec. CCITT X.721 | ISO/CEI 10165-2.

## 13 Conformité

Toute implémentation déclarée conforme à la présente Recommandation | Norme internationale répondra aux prescriptions des paragraphes suivants.

### 13.1 Conformité statique

L'implémentation sera conforme aux dispositions de la présente Recommandation | Norme internationale pour le rôle de gestionnaire, d'agent ou les deux rôles. Une déclaration de conformité à un des rôles au moins sera faite dans le Tableau C.1. Si une déclaration de conformité est faite pour la prise en charge du rôle de gestionnaire, l'implémentation doit prendre en charge au moins un des attributs, notifications, actions ou objets gérés décrits dans le Tableau C.2. Une déclaration de conformité dans le rôle de gestionnaire requiert la prise en charge d'au moins une des notifications ou opérations de gestion spécifiées par ces définitions de gestion.

Si une déclaration de conformité est faite pour la prise en charge du rôle d'agent, l'implémentation prendra en charge au moins un des attributs, actions ou objets gérés décrits dans le Tableau C.3. Une déclaration de conformité dans le rôle d'agent requiert la prise en charge de toutes les opérations obligatoires et notifications obligatoires spécifiées par ces définitions de gestion.

L'implémentation prendra en charge la syntaxe de transfert dérivée des règles de codage spécifiées dans la Rec. UIT-T X.690 | ISO/CEI 8825-1 nommée {joint-iso-ccitt asn1(1) basicEncoding(1)} pour les types abstraits de données pointés par les définitions pour lesquelles la prise en charge est revendiquée.

### 13.2 Conformité dynamique

Toute implémentation revendiquant être conforme à la présente Recommandation | Norme internationale prendra en charge les éléments de procédure et les définitions sémantiques correspondant aux définitions pour lesquelles la prise en charge est revendiquée.

### 13.3 Prescriptions relatives à la déclaration de conformité d'implémentation de gestion

Tout formulaire de récapitulatif MCS (conformité de gestion), ou de déclaration PICS (conformité d'implémentation de protocole), MOCS (conformité d'objet géré) ou MIDS (conformité de définition d'information de gestion) se conformant à la présente Recommandation | Norme internationale doit être techniquement identique aux formulaires spécifiés dans les Annexes C, D, E et F et en conserver la numérotation de tableaux et d'articles, et n'en différera que par la pagination et les en-têtes de pages.

## **ISO/CEI 10164-15:2002 (F)**

Le fournisseur d'une implémentation déclarée conforme à la présente Recommandation | Norme internationale remplira une copie du récapitulatif de conformité de gestion MCS fourni dans l'Annexe C dans le cadre des prescriptions de conformité, ainsi que, s'il y a lieu, tous les autres formulaires de déclaration de conformité imposés par ce récapitulatif. Un récapitulatif MCS, une déclaration MIDS, MOCS, MRCS ou PICS conforme à la présente Recommandation | Norme internationale doit:

- décrire une implémentation conforme à la présente Recommandation | Norme internationale;
- être complété conformément aux instructions données dans la Rec. UIT-T X.724 | ISO/CEI 10165-6;
- inclure l'information nécessaire pour identifier sans ambiguïté à la fois le fournisseur et l'implémentation.

Toute déclaration de conformité à l'information de gestion définie dans la présente Recommandation | Norme internationale dans des classes d'objets gérés définies par ailleurs doit inclure des prescriptions des formulaires de déclaration MIDS dans les formulaires de déclaration MOCS pour la classe d'objets gérés.

## Annexe A

### Définition des informations de gestion

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

#### A.1 Définition des classes d'objets

##### A.1.1 Définition de l'objet programmeur

scheduler MANAGED OBJECT CLASS  
 DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2 : 1992":top;  
 CHARACTERIZED BY  
     schedulerObjectPackage,  
     "Rec. X.721 | ISO/IEC 10165-2 : 1992":duration;  
 CONDITIONAL PACKAGES  
     scheduledManagedObjectsPackage  
     PRESENT IF "pris en charge par une instance.";  
 REGISTERED AS {schedMo 1};

##### A.1.2 Définition de l'objet programmeur journalier

dailyScheduler MANAGED OBJECT CLASS  
 DERIVED FROM scheduler;  
 CHARACTERIZED BY  
     multipleDailyScheduling;  
 REGISTERED AS {schedMo 2};

##### A.1.3 Définition de l'objet programmeur hebdomadaire

weeklyScheduler MANAGED OBJECT CLASS  
 DERIVED FROM scheduler;  
 CHARACTERIZED BY  
     multipleWeeklyScheduling;  
 REGISTERED AS {schedMo 3};

##### A.1.4 Définition de l'objet programmeur mensuel

monthlyScheduler MANAGED OBJECT CLASS  
 DERIVED FROM scheduler;  
 CHARACTERIZED BY  
     multipleMonthlyScheduling;  
 REGISTERED AS {schedMo 4};

##### A.1.5 Définition de l'objet programmeur périodique

periodicScheduler MANAGED OBJECT CLASS  
 DERIVED FROM scheduler;  
 CHARACTERIZED BY  
     periodicSchedulingPackage;  
 CONDITIONAL PACKAGES  
     resynchronizeModePackage PRESENT IF "pris en charge par une instance  
     et le paquetage 'synchronisation périodique' n'est pas  
     présent",  
     "Rec. X.739 (1993) | ISO/IEC 10164-11:1994":periodSynchronizationPackage  
     PRESENT IF "une synchronisation est requise pour  
     un instant autre que l'instant de début de la durée, et le paquetage  
     'mode de resynchronisation' n'est pas présent.";  
 REGISTERED AS {schedMo 5};

##### A.1.6 Définition de l'objet programmeur d'opérations journalier

dailyOperationScheduler MANAGED OBJECT CLASS  
 DERIVED FROM dailyScheduler;  
 CHARACTERIZED BY  
     operationsSchedulingPackage;  
 CONDITIONAL PACKAGES  
     operationNotificationPackage PRESENT IF "le résultat de l'opération nécessite un  
     compte rendu ou l'opération GET est effectuée";  
 REGISTERED AS {schedMo 6};

### A.1.7 Définition de l'objet programmeur d'opérations hebdomadaire

weeklyOperationScheduler MANAGED OBJECT CLASS  
DERIVED FROM weeklyScheduler;  
CHARACTERIZED BY  
    operationsSchedulingPackage;  
CONDITIONAL PACKAGES  
    operationNotificationPackage PRESENT IF "le résultat de l'opération nécessite un  
    compte rendu ou l'opération GET est effectuée";  
REGISTERED AS {schedMo 7};

### A.1.8 Définition de l'objet programmeur d'opérations mensuel

monthlyOperationScheduler MANAGED OBJECT CLASS  
DERIVED FROM monthlyScheduler;  
CHARACTERIZED BY  
    operationsSchedulingPackage;  
CONDITIONAL PACKAGES  
    operationNotificationPackage PRESENT IF "le résultat de l'opération nécessite un compte  
    rendu ou l'opération GET est effectuée";  
REGISTERED AS {schedMo 8};

### A.1.9 Définition de l'objet programmeur d'opérations périodique

periodicOperationScheduler MANAGED OBJECT CLASS  
DERIVED FROM periodicScheduler;  
CHARACTERIZED BY  
    operationsSchedulingPackage;  
CONDITIONAL PACKAGES  
    operationNotificationPackage PRESENT IF "le résultat de l'opération nécessite un compte  
    rendu ou l'opération GET est effectuée";  
REGISTERED AS {schedMo 9};

### A.1.10 Définition de l'objet enregistrement de résultat d'opération

operationResultRecord MANAGED OBJECT CLASS  
DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2" : eventLogRecord;  
CHARACTERIZED BY  
    operationResultRecordPackage PACKAGE  
BEHAVIOUR operationResultRecordBehaviour BEHAVIOUR  
    DEFINED AS "se référer au paragraphe 8.3.11.2";  
    ATTRIBUTES  
        operationResult GET;;;  
REGISTERED AS {schedMo 10};

### A.1.11 Définition de l'objet programmeur multiple

multiScheduler MANAGED OBJECT CLASS  
DERIVED FROM scheduler;  
CHARACTERIZED BY  
    multiSchedulerPackage PACKAGE  
    BEHAVIOUR multiSchedulerPackageBeh BEHAVIOUR  
    DEFINED AS "se référer au paragraphe 8.2.10 'programmeur multiple.'";  
    ATTRIBUTES  
        schedulingData  
        GET-REPLACE  
        ADD-REMOVE;;;  
    CONDITIONAL PACKAGES  
        intervalSchedulingPackage PACKAGE  
        BEHAVIOUR intervalSchedulingPackageBeh BEHAVIOUR  
        DEFINED AS "Comme les programmes définis dans l'attribut 'données de programmation' sont indépendants,  
il se peut qu'aucun des intervalles ne soit valide. La valeur indiquée par l'index par défaut s'applique dans une telle  
situation. Les contraintes suivantes s'appliquent à l'attribut 'données de programmation':  
– l'index est le seul choix valide pour le champ IndexOrOperSpec,  
– la structure intervalsOfDayWps est le seul champ valide pour la structure TimeOfDayWps.  
Lorsque le programmeur d'intervalle est créé ou repris à un instant qui trouve au sein d'une durée définie par le  
programmeur, les activités au sein de l'objet SMO seront fixées conformément à l'index défini par le programmeur  
d'intervalle. L'activité au sein de l'objet SMO sera fixée conformément à l'index par défaut lorsque le  
programmeur d'intervalle est supprimé ou suspendu, ou lorsque l'instant se trouve en-dehors de la durée définie par le  
programmeur.";;  
    ATTRIBUTES  
        defaultIndex

```

REPLACE-WITH-DEFAULT
DEFAULT VALUE Schedulerev1-ASN1Module.defaultDefaultIndex
GET-REPLACE;
REGISTERED AS { schedPkg 1};
PRESENT IF "L'instance fournit la programmation d'intervalle",
    triggerSchedulingPackage PACKAGE
    BEHAVIOUR triggerSchedulingPackageBeh BEHAVIOUR
    DEFINED AS "Les contraintes suivantes s'appliquent à l'attribut 'données de programmation':
    – la structure TimesOfDayWps doit contenir le champ triggerTimes;
    – le champ 'priorité' optionnel sera absent.
    Lorsque le programmeur de déclenchement est créé ou repris à un instant qui se trouve au sein d'une durée définie
    par le programmeur, les activités au sein de l'objet SMO seront fixées conformément à l'index défini par le programme.
    L'activité au sein de l'objet SMO ne sera pas déclenchée lorsque le programmeur d'intervalle est supprimé ou suspendu,
    ou lorsque l'instant se trouve en-dehors de la durée définie par le programmeur.";;
REGISTERED AS { schedPkg 2};
PRESENT IF "L'instance fournit la programmation de déclenchement",
    TypeOfDayControllerInstancePackage PACKAGE
    BEHAVIOUR typeOfDayControllerInstancePackageBeh BEHAVIOUR
    DEFINED AS "L'attribut 'instance de contrôleur de type de jour' identifie l'instance de la classe d'objets
    'contrôleur de type de jour' qui correspond à cette instance de la classe d'objet 'programmeur multiple'.";;
ATTRIBUTES
    typeOfDayControllerInstance
GET-REPLACE;
REGISTERED AS { schedPkg 3};
PRESENT IF "Plusieurs instances de la classe d'objets 'contrôleur de type de jour' peuvent exister dans l'élément
géré et l'instance du programmeur utilise la programmation par type de jour ",
    "ITU-T Recommendation X.746":operationNotificationPackage
PRESENT IF "Le résultat d'une opération nécessite un compte rendu ou l'opération GET
est effectuée.";;
REGISTERED AS { schedMo 11};

```

### A.1.12 Contrôleur de type de jour

```

typeOfDayController MANAGED OBJECT CLASS
DERIVED FROM "ITU-T Recommendation X.721:1992":top;
CHARACTERIZED BY
    typeOfDayControllerPackage PACKAGE
    BEHAVIOUR typeOfDayControllerPackageBeh BEHAVIOUR
    DEFINED AS " Comme décrit dans le paragraphe traitant du contrôleur de type de jour. La correspondance est
    maintenue entre les attributs 'liste de traduction de date' et 'liste de traduction du jour de la semaine'. La valeur du type de
    jour spécifiée pour certaines dates (au moyen de l'attribut liste de traduction de date) a priorité par rapport à la valeur
    spécifiée par la valeur du type de jour pour un jour dans la semaine (au moyen de l'attribut liste de traduction du jour de la
    semaine). ";;
ATTRIBUTES
    typeOfDayControllerId
    GET,
    currentTypeOfDay
    INITIAL VALUE DERIVATION RULE
        currentTypeOfDayAlgorithm
    GET,
    dateTranslationList
    REPLACE-WITH-DEFAULT
    DEFAULT VALUE Schedulerev1-ASN1Module.defaultDateTranslationList
    GET-REPLACE
    ADD-REMOVE,
    weekDayTranslationList
    REPLACE-WITH-DEFAULT
    DEFAULT VALUE Schedulerev1-ASN1Module.defaultWeekDayTranslationList
    GET-REPLACE;;;
    "ITU-T Recommendation M.3100: 1995":objectManagementNotificationsPackage;
REGISTERED AS { schedMo 11};

```

## A.2 Liens de dénomination

Des liens de dénomination supplémentaires peuvent être définis et enregistrés pour chaque classe d'objets "programmeur".

## ISO/CEI 10164-15:2002 (F)

### A.2.1 Liens de dénomination de programmeurs

Le squelette suivant le lien de dénomination fournit les liens de dénomination définis actuellement pour nommer des instances de la classe d'objets gérés programmeur.

```
scheduler-system NAME BINDING
  SUBORDINATE OBJECT CLASS
  scheduler AND SUBCLASSES;
  NAMED BY
  SUPERIOR OBJECT CLASS
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":system
    AND SUBCLASSES;
  WITH ATTRIBUTE
    schedulerID;
  CREATE
  WITH-REFERENCE-OBJECT,
  WITH-AUTOMATIC-INSTANCE-NAMING
  "Rec. X.738|ISO/IEC 10164-13:1992":conflictingPackagesRequestedError;
  DELETE
  ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS {schedNb 1};
scheduler-managedElement NAME BINDING
  SUBORDINATE OBJECT CLASS
  scheduler AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS
    "ITU-T Recommendation M.3100": managedElement
    AND SUBCLASSES;
  WITH ATTRIBUTE
    schedulerID;
  CREATE
  WITH-REFERENCE-OBJECT,
  WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE;
REGISTERED AS { schedNb 2};
```

### A.2.2 Liens de dénomination de contrôleur de type de jour

Le squelette suivant le lien de dénomination fournit les liens de dénomination définis actuellement pour nommer des instances de la classe d'objets gérés contrôleur de type de jour.

```
typeOfDayController-managedElement NAME BINDING
  SUBORDINATE OBJECT CLASS
  typeOfDayController AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS
    "ITU-T Recommendation M.3100": managedElement
    AND SUBCLASSES;
  WITH ATTRIBUTE
    typeOfDayControllerId;
  CREATE
  WITH-REFERENCE-OBJECT,
  WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE;
REGISTERED AS { schedNb 3};

typeOfDayController-system NAME BINDING
  SUBORDINATE OBJECT CLASS
  TypeOfDayController AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS
    "Rec. X.721 | ISO/IEC 10165-2 : 1992":system AND SUBCLASSES;
  WITH ATTRIBUTE
    typeOfDayControllerId;
  CREATE
  WITH-REFERENCE-OBJECT,
  WITH-AUTOMATIC-INSTANCE-NAMING;
  DELETE;
REGISTERED AS { schedNb 4};
```

### A.3 Paquetages

#### A.3.1 Paquetage programmation journalière multiple

```
multipleDailyScheduling PACKAGE
  BEHAVIOUR multipleDailySchedulingBehaviour BEHAVIOUR
  DEFINED AS "se référer aux paragraphes 8.3.3 et 8.3.4.3";
  ATTRIBUTES
  sequenceOfDays DEFAULT VALUE
  Schedulerrev1-ASN1Module.defaultSequenceOfDays
  GET-REPLACE;
REGISTERED AS {schedPkg 10};
```

#### A.3.2 Paquetage programmation mensuelle multiple

```
multipleMonthlyScheduling PACKAGE
  BEHAVIOUR multipleMonthlySchedulingBehaviour BEHAVIOUR
  DEFINED AS "se référer aux paragraphes 8.3.3 et 8.3.6.3";
  ATTRIBUTES
  sequenceOfMonths DEFAULT VALUE
  Schedulerrev1-ASN1Module.defaultSequenceOfMonths
  GET-REPLACE;
REGISTERED AS {schedPkg 11};
```

#### A.3.3 Paquetage programmation hebdomadaire multiple

```
multipleWeeklyScheduling PACKAGE
  BEHAVIOUR multipleWeeklySchedulingBehaviour BEHAVIOUR
  DEFINED AS "se référer aux paragraphes 8.3.3 et 8.3.5.3";
  ATTRIBUTES
  sequenceOfWeeks DEFAULT VALUE
  Schedulerrev1-ASN1Module.defaultSequenceOfWeeks
  GET-REPLACE;
REGISTERED AS {schedPkg 12};
```

#### A.3.4 Paquetage programmation périodique

```
periodicSchedulingPackage PACKAGE
  BEHAVIOUR periodicSchedulingBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.7.3";
  ATTRIBUTES
  timePeriod DEFAULT VALUE
  Schedulerrev1-ASN1Module.defaultTimePeriod
  GET-REPLACE;
REGISTERED AS {schedPkg 4};
```

#### A.3.5 Paquetage mode de resynchronisation

```
resynchronizeModePackage PACKAGE
  BEHAVIOUR resynchronizeModePackageBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.8.3";
  ATTRIBUTES
  resynchronizeMode GET-REPLACE;
REGISTERED AS {schedPkg 5};
```

#### A.3.6 Paquetage objets gérés programmés

```
scheduledManagedObjectsPackage PACKAGE
  BEHAVIOUR scheduledManagedObjectBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.2";
  ATTRIBUTES
  scheduledManagedObjects GET;
REGISTERED AS {schedPkg 6};
```

#### A.3.7 Paquetage objet programmeur

```
schedulerObjectPackage PACKAGE
  BEHAVIOUR schedulerObjectBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.1.4";
  ATTRIBUTES
  schedulerID GET,
  "Rec. X.721 | ISO/IEC 10165-2 : 1992":administrativeState
  GET-REPLACE,
```

## ISO/CEI 10164-15:2002 (F)

"Rec. X.721 | ISO/IEC 10165-2 : 1992":operationalState GET;  
NOTIFICATIONS  
"Rec. X.721 | ISO/IEC 10165-2 : 1992":attributeValueChange,  
"Rec. X.721 | ISO/IEC 10165-2 : 1992":stateChange,  
"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectCreation,  
"Rec. X.721 | ISO/IEC 10165-2 : 1992":objectDeletion;

REGISTERED AS {schedPkg 7};

### A.3.8 Paquetage programmation d'opérations

operationsSchedulingPackage PACKAGE  
BEHAVIOUR operationsSchedulingBehaviour BEHAVIOUR  
DEFINED AS "se référer au paragraphe 8.3.10";  
ATTRIBUTES  
operationSpecifications GET-REPLACE ADD-REMOVE;  
REGISTERED AS {schedPkg 8};

### A.3.9 Paquetage notification d'opération

operationNotificationPackage PACKAGE  
BEHAVIOUR operationNotificationBehaviour BEHAVIOUR  
DEFINED AS "se référer au paragraphe 8.3.11";  
NOTIFICATIONS  
operationResultNotification;  
REGISTERED AS {schedPkg 9};

## A.4 Attributs

### A.4.0.1 Attribut type de jour actuel

currentTypeOfDay ATTRIBUTE  
WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.TypeOfDay;  
MATCHES FOR EQUALITY;  
REGISTERED AS {schedAtt 12};

### A.4.0.2 Attribut liste de traduction de date

dateTranslationList ATTRIBUTE  
WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.DateTranslationList;  
MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;  
BEHAVIOUR dateTranslationListBeh BEHAVIOUR  
DEFINED AS

"La liste de traduction de date est un ensemble qui contient le mappage d'une date donnée vers un type de jour. Le type de jour peut être, soit du type 'jour de la semaine' (jour ouvrable ou 'week-end'), soit l'un des jours spéciaux qui sont définis. Une date donnée figure une seule fois dans l'attribut 'liste de traduction de date.'";

REGISTERED AS {schedAtt 13};

### A.4.0.3 Index par défaut

defaultIndex ATTRIBUTE  
WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.DefaultIndex;  
MATCHES FOR EQUALITY;  
BEHAVIOUR defaultIndexBeh BEHAVIOUR  
DEFINED AS

"Comme dans le paragraphe 8.2.10 'programmeur multiple.'";

REGISTERED AS {schedAtt 14};

### A.4.1 Attribut nom de programmeur externe

Cet attribut est contenu dans les objets gérés programmés. Il spécifie l'instance d'objet programmé qui pilote l'activité dans l'objet SMO.

externalSchedulerName ATTRIBUTE  
WITH ATTRIBUTE SYNTAX  
Schedulerev1-ASN1Module.ExternalSchedulerName;  
MATCHES FOR EQUALITY;  
BEHAVIOUR externalSchedulerNameBehaviour BEHAVIOUR  
DEFINED AS "se référer au paragraphe 8.4.1.1";  
REGISTERED AS {schedAtt 1};

**A.4.2 Attribut "en service"**

Cet attribut est contenu dans les objets SMO. Il spécifie le statut d'activité actuel de l'objet SMO.

```
onDuty ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        Schedulerev1-ASN1Module.OnDuty;
    MATCHES FOR EQUALITY;
    BEHAVIOUR onDutyBehaviour BEHAVIOUR
        DEFINED AS "se référer au paragraphe 8.4.1.2";;
REGISTERED AS {schedAtt 2};
```

**A.4.3 Attribut mode de resynchronisation**

Cet attribut définit le mode de synchronisation d'une période de déclenchement du programmeur périodique.

```
resynchronizeMode ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.ResynchronizeMode;
    MATCHES FOR EQUALITY;
    BEHAVIOUR resynchronizeModeBehaviour BEHAVIOUR
        DEFINED AS "se référer au paragraphe 8.3.8.2";;
REGISTERED AS {schedAtt 3};
```

**A.4.4 Attribut objets gérés programmés**

Cet attribut est défini dans l'objet programmeur pour spécifier les instances d'objet SMO et l'identificateur de l'attribut associé à toutes les activités au sein de ces instances qui sont pilotées par l'objet SO.

```
scheduledManagedObjects ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        Schedulerev1-ASN1Module.ScheduledManagedObjectsList;
    MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
    BEHAVIOUR scheduledManagedObjectsBehaviour BEHAVIOUR
        DEFINED AS "se référer au paragraphe 8.3.2.2";;
REGISTERED AS {schedAtt 4};
```

**A.4.5 Attribut identificateur de programmeur**

Cet attribut contient le nom distinctif pour des instances d'un objet SO.

```
schedulerID ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SimpleNameType;
    MATCHES FOR EQUALITY, SUBSTRINGS;
    BEHAVIOUR schedulerIDBehaviour BEHAVIOUR
        DEFINED AS "se référer au paragraphe 8.3.1.2";;
REGISTERED AS {schedAtt 5};
```

**A.4.6 Attribut données de programmation**

```
schedulingData ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SchedulingData;
    MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
    BEHAVIOUR schedulingDataBeh BEHAVIOUR
        DEFINED AS "Comme indiqué dans le paragraphe 8.2.10 décrivant le programmeur multiple, les intervalles
        contenus dans un membre de l'ensemble ne se chevauchent pas, mais les intervalles définis dans des membres différents peuvent se
        chevaucher. Dans un tel cas, le membre 'priorité' de la séquence sera présent pour chaque membre dont les intervalles se
        chevauchent. La priorité associée à chaque membre détermine l'activité qui sera programmée. Les priorités sont ordonnées par
        valeurs numériques croissantes.";;
REGISTERED AS {schedAtt 15};
```

**A.4.7 Attribut séquence journalière**

Cet attribut est une structure qui définit une succession d'intervalles au sein d'un jour.

```
sequenceOfDays ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SequenceOfDays;
    MATCHES FOR EQUALITY;
    BEHAVIOUR sequenceOfDaysBehaviour BEHAVIOUR
        DEFINED AS "se référer au paragraphe 8.3.4.2";;
REGISTERED AS {schedAtt 6};
```

## ISO/CEI 10164-15:2002 (F)

### A.4.8 Attribut séquence mensuelle

Cet attribut est une structure qui définit une succession de masques mensuels.

```
sequenceOfMonths ATTRIBUTE
  WITH ATTRIBUTE SYNTAX
    Schedulerev1-ASN1Module.SequenceOfMonths;
  MATCHES FOR EQUALITY;
  BEHAVIOUR sequenceOfMonthsBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.6.2";;
REGISTERED AS {schedAtt 7};
```

### A.4.9 Attribut séquence hebdomadaire

Cet attribut est une structure qui définit une succession de masques hebdomadaires.

```
sequenceOfWeeks ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.SequenceOfWeeks;
  MATCHES FOR EQUALITY;
  BEHAVIOUR sequenceOfWeeksBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.5.2";;
REGISTERED AS {schedAtt 8};
```

### A.4.10 Attribut période de temps

Cet attribut définit la longueur d'une période de temps pour le déclenchement périodique d'une activité dans un objet SMO par l'objet déclenchement périodique.

```
timePeriod ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.TimePeriod;
  MATCHES FOR EQUALITY, ORDERING;
  BEHAVIOUR timePeriodBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.7.2";;
REGISTERED AS {schedAtt 9};
```

### A.4.11 Attribut spécifications d'opération

Cet attribut définit les opérations qui peuvent être programmées pour un objet SMO par un objet programmation d'opérations.

```
operationSpecifications ATTRIBUTE
  WITH ATTRIBUTE SYNTAX
    Schedulerev1-ASN1Module.OperationSpecifications;
  MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
  BEHAVIOUR operationSpecificationsBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.10.2";;
REGISTERED AS {schedAtt 10};
```

### A.4.12 Attribut résultat d'opération

Cet attribut est contenu dans des objets d'enregistrement de compte rendu d'opération. Il spécifie le résultat des opérations effectuées par des programmeurs d'opération.

```
operationResult ATTRIBUTE
  WITH ATTRIBUTE SYNTAX
    Schedulerev1-ASN1Module.OperationResult;
  MATCHES FOR EQUALITY;
  BEHAVIOUR operationResultBehaviour BEHAVIOUR
  DEFINED AS "se référer au paragraphe 8.3.11.2";;
REGISTERED AS {schedAtt 11};
```

### A.4.13 Attribut instance de contrôleur de type de jour

```
typeOfDayControllerInstance ATTRIBUTE
  WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.ObjectInstance;
  MATCHES FOR EQUALITY;
  BEHAVIOUR typeOfDayControllerInstanceBeh BEHAVIOUR
  DEFINED AS
    "Cet attribut pointe vers une instance de la classe d'objets 'contrôleur de type de jour'. ";;
REGISTERED AS {schedAtt 16};
```

**A.4.14 Attribut identificateur de contrôleur de type de jour**

```

typeOfDayControllerId ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.NameType;
    MATCHES FOR EQUALITY;
REGISTERED AS {schedAtt 17};

```

**A.4.15 Attribut liste de traduction du jour de la semaine**

```

weekDayTranslationList ATTRIBUTE
    WITH ATTRIBUTE SYNTAX Schedulerev1-ASN1Module.
WeekDayTranslationList;
    MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR weekDayTranslationListBeh BEHAVIOUR
    DEFINED AS

```

"La liste de traduction est un ensemble qui définit la correspondance entre un jour donné dans la semaine et un type de jour. Le type de jour peut être un jour de la semaine (jour ouvrable ou week-end) ou l'un des jours spéciaux qui ont été définis. Il existera une correspondance et une seule pour tout jour de la semaine.";

```
REGISTERED AS {schedAtt 18};
```

**A.4.16 Valeur initiale du comportement pour le type de jour actuel**

```

currentTypeOfDayAlgorithm BEHAVIOUR
    DEFINED AS

```

"L'attribut type de jour actuel est défini conformément aux valeurs des attributs 'liste de traduction de date' et 'liste de traduction du jour de la semaine'.";

**A.5 Notifications****A.5.1 Notification de résultat d'opération**

```

operationResultNotification NOTIFICATION
    BEHAVIOUR operationResultNotificationBehaviour BEHAVIOUR
    DEFINED AS "se référer au paragraphe 8.3.11.2";
    WITH INFORMATION SYNTAX Schedulerev1-ASN1Module.OperationResult;
REGISTERED AS {schedNotif 1};

```

**A.6 Définitions ASN.1**

```
Schedulerev1-ASN1Module { joint-iso-itu-t ms(9) function(2) part15(15) modules (2) }
```

```
DEFINITIONS IMPLICIT TAGS ::=
```

```
BEGIN
```

```
-- EXPORTER tout
```

```
IMPORTS
```

```

CMIP-ATTRIBUTE, AttributeSet,
Attribute,AttributeId,ObjectClass,ObjectInstance,ActionResult,SetResult,SetListError,ActionError,ModifyOperator,Action
Info,GetResult,GetListError,GetArgument FROM CMIP-1
    { joint-iso-itu-t ms(9) cmip(1) modules(0) protocol(3) }

```

```

DMI-TYPE-IDENTIFIER, SimpleNameType,defaultStopTime FROM Attribute-ASN1Module
    { joint-iso-itu-t ms(9) smi(3) part2(2) asn1Module(2) 1}

```

```

TimePeriod FROM MetricModule
{ joint-iso-itu-t ms(9) function(2) part11(11) asn1Module(2) 0};

```

```

schedMo OBJECT IDENTIFIER ::=
    {joint-iso-itu-t ms(9) function(2) part15(15) managedObjectClass(3)}

```

```

schedAtt OBJECT IDENTIFIER ::=
    {joint-iso-itu-t ms(9) function(2) part15(15) attribute(7)}

```

```

schedNotif OBJECT IDENTIFIER ::=
    {joint-iso-itu-t ms(9) function(2) part15(15) notification(10)}

```

```

schedPkg OBJECT IDENTIFIER ::=
    {joint-iso-itu-t ms(9) function(2) part15(15) package(4)}

```

```

schedNb OBJECT IDENTIFIER ::=
    {joint-iso-itu-t ms(9) function(2) part15(15) nameBinding(6)}

```

```
-- définition de valeurs par défaut
```



```

OnDuty ::=    BOOLEAN

OperationSpecifications ::=    SET OF SEQUENCE{
    scheduledObjects    SET OF ObjectInstance,
    scheduledOperations CHOICE{
    set                 [0] IMPLICIT ModificationList,
    action              [1] IMPLICIT ActionInfo,
    get                 [2] IMPLICIT GetArgument}}

OperationResult ::= SET OF CHOICE{
    getResult           [0] IMPLICIT GetResult,
    getListError        [1] IMPLICIT GetListError,
    setResult           [2] IMPLICIT SetResult,
    setListError        [3] IMPLICIT SetListError,
    actionResult        [4] IMPLICIT ActionResult,
    commonError         [5] IMPLICIT CommonError}

SCHED-ERROR ::= CLASS {
    &id    ErrorId UNIQUE,
    &Value }

CommonError ::= SEQUENCE{
    managedObjectClass    ObjectClass,
    managedObjectInstance ObjectInstance,
    errorId                SCHED-ERROR.&id ({CommonErrorSet}),
    errorValue             SCHED-ERROR.&Value ({CommonErrorSet} {@.errorId})

OPTIONAL}
CommonErrorSet SCHED-ERROR ::= {...}
ErrorId ::= CHOICE {
    localValue    INTEGER,
    globalValue   OBJECT IDENTIFIER}

Schedule ::= CHOICE {
    daily    [0] SequenceOfDays,
    weekly   [1] SequenceOfWeeks,
    monthly  [2] SequenceOfMonths,
    typeOfDay [3] SequenceOfTypeOfDays}

ScheduledManagedObjectsList ::= SET OF SEQUENCE {
    objectInstance ObjectInstance,
    activity AttributeId OPTIONAL}

SchedulingData ::= SET OF SEQUENCE {
    indexOrOperSpec [0] IndexOrOperSpec,
    schedule [1] Schedule,
    priority [2] INTEGER OPTIONAL}

ResynchronizeMode ::= BOOLEAN

SequenceOfDays ::= SEQUENCE OF TimesOfDayWps
SequenceOfWeeks ::= SEQUENCE OF WeekMaskWps
SequenceOfMonths ::= SEQUENCE OF MonthMask

SequenceOfTypeOfDays ::= SEQUENCE SIZE(1) OF TypeOfDaysMaskWps
-- la taille SIZE(1) est utilisée parce qu'une valeur supérieure n'a pas de signification, étant donné qu'il n'existe
-- pas de périodicité implicite, contrairement à ce qui est le cas pour les séquences de jours, de semaine ou de mois.

Time24Wps ::= SEQUENCE {
    hour [1] INTEGER (0..23),
    minute [2] INTEGER (0..59) OPTIONAL,
    second [3] INTEGER (0..59) OPTIONAL,
    milliseconds [4] INTEGER (0..999) OPTIONAL,
    microseconds [5] INTEGER (0..999999) OPTIONAL,
    nanoseconds [6] INTEGER (0..999999999) OPTIONAL,
    picoseconds [7] INTEGER (0..999999999999) OPTIONAL}

TimesOfDayWps ::= CHOICE {
    intervalsOfDayWps [0] IMPLICIT IntervalsOfDayWps,
    triggerTimes [1] IMPLICIT TriggerTimes}

TriggerTimes ::= SET OF Time24Wps

TypeOfDay ::= INTEGER {
    workday (0),
    weekend (1),

```

```

specialDay1 (2),
specialDay2 (3),
specialDay3 (4),
specialDay4 (5),
specialDay5 (6),
specialDay6 (7),
specialDay7 (8),
specialDay8 (9),
specialDay9 (10),
specialDay10 (11),
specialDay11 (12),
specialDay12 (13),
specialDay13 (14),
specialDay14 (15),
specialDay15 (16),
specialDay16 (17) }

```

```

TypeOfDaysMaskWps ::= SET OF SEQUENCE {
    typeOfDay    TypeOfDay,
    timesOfDay   TimesOfDayWps}

```

```

WeekDay ::= ENUMERATED {
    sunday (0),
    monday (1),
    tuesday (2),

    wednesday (3),
    thursday (4),
    friday (5),
    saturday (6)}

```

```

WeekDayTranslationList ::= SET OF SEQUENCE {
    weekDay [1] WeekDay,
    typeOfDay [2] TypeOfDay}

```

```

WeekMaskWps ::= SET OF SEQUENCE {
    daysOfWeek    DaysOfWeek,
    timesOfDayWps TimesOfDayWps}

```

```

END -- fin des productions auxiliaires

```

## Annexe B

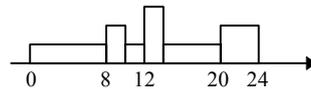
### Exemples de valeurs de données de programmation

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

#### B.1 Exemple 1: utilisation de plusieurs valeurs d'index

##### B.1.1 Solution avec intervalles

Utilisation de l'index 1 pour chaque jour de 8:00 à 10:00 heures et de 20:00 à 24:00 heures, utilisation de l'index 2 de 12:00 à 14:00 heures, sinon utilisation de l'index 0.

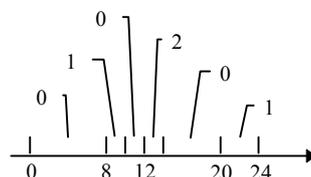


L'attribut donnée de programmation se présente de la manière suivante:

```
schedulingData:
  { -- SET OF
    { -- SEQUENCE
      indexOrOperSpec: index: 1,
      schedule: daily: -- journalier, de sorte que
      { -- SEQUENCE OF TimesOfDayWps
        intervalsOfDayWps:
          { -- SET OF
            {
              intervalStart: { hour: 8 },
              intervalEnd: { hour: 10 }
            },
            {
              intervalStart: { hour: 20 },
              intervalEnd: { hour: 0 }
            }
          }
        }
      }
    },
    { -- SEQUENCE
      indexOrOperSpec: index: 2,
      schedule: daily: -- journalier, de sorte que
      { -- SEQUENCE OF TimesOfDayWps
        intervalsOfDayWps:
          { -- SET OF
            {
              intervalStart: { hour: 12 },
              intervalEnd: { hour: 14 }
            }
          }
        }
      },
      -- sans priorité
    }
  }
defaultIndex: 0
```

NOTE – Il n'est pas nécessaire d'utiliser des priorités parce que les intervalles ne se recouvrent pas.

##### B.1.2 Solution avec déclenchement



L'attribut donnée de programmation se présente de la manière suivante:

```

{ -- SET OF
  { -- SEQUENCE
    indexOrOperSpec: index: 0,
    schedule: daily: -- journalier, de sorte que
    { -- SEQUENCE OF TimesOfDayWps
      triggerTimes:
        { -- SET OF Times24Wps
          {hour: 0},
          {hour: 10},
          {hour: 14}
        }
      }
    }
  },
  { -- SEQUENCE
    indexOrOperSpec: index: 1,
    schedule: daily: -- journalier, de sorte que
    { -- SEQUENCE OF TimesOfDayWps
      triggerTimes:
        { -- SET OF Times24Wps
          {hour: 8},
          {hour: 20}
        }
      }
    }
  },
  { -- SEQUENCE
    indexOrOperSpec: index: 2,
    schedule: daily: -- journalier, de sorte que
    { -- SEQUENCE OF TimesOfDayWps
      triggerTimes:
        { -- SET OF Times24Wps
          {hour: 12}
        }
      }
    }
  }
}

```

## B.2 Exemple 2: utilisation du type de jour

Utilisation de l'index 3 pour chaque jour ouvrable de 0:00 à 6:00 heures et de 20:00 à 00:00 heures, ainsi que pour la totalité des jours de week-end, sinon utilisation de l'index 0.

### B.2.1 Solution avec intervalles

```

schedulingData:
  { -- SET OF
    { -- SEQUENCE
      indexOrOperSpec: index: 3,
      schedule: typeOfDay: -- typeOfDay, de sorte que
      { -- SEQUENCE OF TypeOfDaysMaskWps
        { -- SET OF
          {
            typeOfDays: { workday },
            timesOfDay: intervalsOfDayWps:
              { -- SET OF
                {
                  intervalStart: { hour: 0 },
                  intervalEnd: { hour: 6 }
                },
                {
                  intervalStart: { hour: 20 },
                  intervalEnd: { hour: 0 }
                }
              }
            }
          }
        }
      }
    }
  }

```

```

        typeOfDay: { weekend },
        timesOfDay: intervalsOfDayWps:
        { -- SET OF
            {
                intervalStart: { hour: 0 },
                intervalEnd: { hour: 0 }
            }
        }
    },
    },
    -- sans priorité
}
},
defaultIndex: 0

```

NOTE – Il n'est pas nécessaire d'utiliser des priorités parce que les intervalles ne se recouvrent pas.

### B.2.2 Solution avec déclenchement

Déclenchement avec l'index 3 pour chaque jour ouvrable de 0:00 et à 20:00 heures et avec l'index 0 pour chaque jour ouvrable à 6:00 heures (la totalité du week-end aura l'index 3).

L'attribut donnée de programmation se présente de la manière suivante:

```

{ -- SET OF
  { -- SEQUENCE
    indexOrOperSpec: index: 3,
    schedule: typeOfDay -- typeOfDay, de sorte que
    { -- SEQUENCE OF TypeOfDayMaskWps
      { -- SET OF
        { -- SEQUENCE
          typeOfDay: { workday }, --jour ouvrable,
          timesOfDay : triggerTimes:
          { -- SET OF Time24Wps
            {hour: 20}
          }
        }
      }
    }
  },
  -- sans priorité
},
{ -- SEQUENCE
  indexOrOperSpec: index: 0,
  schedule: typeOfDay -- typeOfDay, de sorte que
  { -- SEQUENCE OF TypeOfDayMaskWps
    { -- SET OF
      { -- SEQUENCE
        typeOfDay: { workday }, --jour ouvrable,
        timesOfDay : triggerTimes:
        { -- SET OF Time24Wps
          {hour: 6}
        }
      }
    }
  }
},
-- sans priorité
}
}

```

### B.3 Exemple 3: recouvrement d'intervalles, répétition implicite (mensuelle) et divers types de programmeurs

Utilisation de l'index 2 pour le deuxième jour du mois à partir de 8:00 heures jusqu'au cinquième jour du mois à 18:00 heures, pour chaque dernier mois du trimestre (mars, juin, septembre, décembre), sinon utilisation de l'index 3 le lundi et le vendredi de 12:00 à 16:00 heures, sinon utilisation de l'index 0. Le positionnement du lundi et du vendredi a une priorité supérieure au positionnement de l'index 2.

**ISO/CEI 10164-15:2002 (F)**

schedulingData:

```

{ -- SET OF
  { -- SEQUENCE
    indexOrOperSpec: index: 2,
    schedule: monthly:
      { -- SEQUENCE OF Monthmask
        { -- SET OF
          {
            daysOfMonth:
              {
                daysFromFirst: { 'B },
                daysFromLast: { 'B }
              }
            timesOfDaysWps: intervalsOfDayWps: {} -- ensemble vide
          }
        }, -- janvier, avril, juillet et octobre
        { -- SET OF
          {
            daysOfMonth:
              {
                daysFromFirst: { 'B },
                daysFromLast: { 'B }
              }
            timesOfDaysWps: intervalsOfDayWps: {} -- ensemble vide
          }
        }, -- février, mai, août et novembre
        { -- SET OF
          {
            daysOfMonth:
              {
                daysFromFirst: { '01'B }, -- deuxième jour
                daysFromLast: { 'B }
              }
            timesOfDaysWps: intervalsOfDayWps:
              { -- SET OF
                {
                  intervalStart: { hour: 8 },
                  intervalEnd: { hour: 0 }
                }
              }
          }
        },
        {
          daysOfMonth:
            {
              daysFromFirst: { '0011'B }, -- 3e et 4e jour
              daysFromLast: { 'B }
            }
          timesOfDaysWps: intervalsOfDayWps:
            { -- SET OF
              {
                intervalStart: { hour: 0 },
                intervalEnd: { hour: 0 }
              }
            }
        },
        {
          daysOfMonth:
            {
              daysFromFirst: { '00001'B }, -- 5e jour
              daysFromLast: { 'B }
            }
          timesOfDaysWps: intervalsOfDayWps:
            { -- SET OF
              {
                intervalStart: { hour: 0 },
                intervalEnd: { hour: 18 }
              }
            }
        }
      }
    }
  }
}

```

```

    }
    priority: 1
  },
  { -- SEQUENCE
    indexOrOperSpec: index: 3,
    schedule: weekly:
      { -- SEQUENCE OF WeekMaskWps
        { -- SET OF
          {
            daysOfWeek: { '0100010'B } -- lundi et vendredi
            timesOfDayWps: intervalsOfDayWps:
              { -- SET OF
                {
                  intervalStart: { hour: 12 },
                  intervalEnd: { hour: 16 }
                }
              }
            }
          }
        }
      }
    }
  },
  priority: 2
}
},
defaultIndex: 0

```

**Annexe C**

**Formulaire MCS**

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Cette annexe appelle une étude ultérieure.

**Annexe D**

**Formulaire PICS**

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Cette annexe appelle une étude ultérieure.

**Annexe E**

**Formulaire MOCS**

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Cette annexe appelle une étude ultérieure.

**Annexe F**

**Formulaire MIDS**

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Cette annexe appelle une étude ultérieure.

**Annexe G**

**Formulaire MRCS**

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Cette annexe appelle une étude ultérieure.



## SÉRIES DES RECOMMANDATIONS UIT-T

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