

Remplacée par une version plus récente



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

X.746

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

(04/95)

**RÉSEAUX DE COMMUNICATION DE DONNÉES ET
COMMUNICATION ENTRE SYSTÈMES OUVERTS
GESTION OSI**

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

Recommandation UIT-T X.746

Remplacée par une version plus récente

(Antérieurement «Recommandation du CCITT»)

Remplacée par une version plus récente

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. Au sein de l'UIT-T, qui est l'entité qui établit les normes mondiales (Recommandations) sur les télécommunications, participent quelque 179 pays membres, 84 exploitations de télécommunications reconnues, 145 organisations scientifiques et industrielles et 38 organisations internationales.

L'approbation des Recommandations par les membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution n° 1 de la Conférence mondiale de normalisation des télécommunications (CMNT) (Helsinki, 1993). De plus, la CMNT, qui se réunit tous les quatre ans, approuve les Recommandations qui lui sont soumises et établit le programme d'études pour la période suivante.

Dans certains secteurs de la technologie 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. Le texte de la Recommandation X.746 de l'UIT-T a été approuvé le 10 avril 1995. Son texte est publié, sous forme identique, comme Norme internationale ISO/CEI 10164-15.

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.

© UIT 1996

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

Remplacée par une version plus récente

RECOMMANDATIONS UIT-T DE LA SÉRIE X

RÉSEAUX DE COMMUNICATION DE DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS

(Février 1994)

ORGANISATION DES RECOMMANDATIONS DE LA SÉRIE X

Domaine	Recommandations
RÉSEAUX PUBLICS POUR DONNÉES	
Services et services complémentaires	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
INTERCONNEXION DES SYSTÈMES OUVERTS	
Modèle et notation	X.200-X.209
Définition 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 de couche	X.280-X.289
Test de conformité	X.290-X.299
INTERFONCTIONNEMENT DES RÉSEAUX	
Considérations générales	X.300-X.349
Systèmes mobiles de transmission de données	X.350-X.369
Gestion	X.370-X.399
SYSTÈMES DE MESSAGERIE	X.400-X.499
ANNUAIRE	X.500-X.599
RÉSEAUTAGE OSI ET ASPECTS DES SYSTÈMES	
Réseautage	X.600-X.649
Dénomination, adressage et enregistrement	X.650-X.679
Notation de syntaxe abstraite numéro un (ASN.1)	X.680-X.699
GESTION OSI	X.700-X.799
SÉCURITÉ	X.800-X.849
APPLICATIONS OSI	
Engagement, concomitance et rétablissement	X.850-X.859
Traitement des transactions	X.860-X.879
Opérations distantes	X.880-X.899
TRAITEMENT OUVERT RÉPARTI	X.900-X.999

Remplacée par une version plus récente

TABLE DES MATIÈRES

	<i>Page</i>
1	Domaine d'application 1
2	Références normatives 1
2.1	Recommandations CCITT/UIT-T et Normes internationales identiques 1
2.2	Paires de Recommandations CCITT/UIT-T Normes internationales équivalentes par leur contenu technique 2
2.3	Autres références 3
3	Définitions 3
3.1	Définitions du modèle de référence de base 3
3.2	Définitions de la notation de syntaxe abstraite numéro un 3
3.3	Définitions du cadre général de gestion 3
3.4	Définitions du service commun d'information de gestion 3
3.5	Définitions de la présentation générale de la gestion-système 4
3.6	Définitions du modèle d'information de gestion 4
3.7	Définitions supplémentaires 4
4	Abréviations 4
5	Conventions 5
6	Prescriptions 5
7	Modèle 5
7.1	Mécanisme de programmation interne 6
7.2	Mécanisme de programmation externe 6
7.3	Modes de programmation 7
7.3.1	Programmation d'intervalle 7
7.3.2	Programmation de déclenchement 7
7.3.3	Programmation d'opération 8
7.4	Relations entre objets SO et objets SMO 8
8	Définitions génériques 8
8.1	Information de gestion requise pour la programmation interne 8
8.2	Objets gérés 8
8.2.1	Programmateur 9
8.2.2	Programmateur journalier 10
8.2.3	Programmateur hebdomadaire 10
8.2.4	Programmateur mensuel 10
8.2.5	Programmateur périodique 10
8.2.6	Programmateur d'opération journalier 10
8.2.7	Programmateur d'opération hebdomadaire 11
8.2.8	Programmateur d'opération mensuel 11
8.2.9	Programmateur d'opération périodique 11
8.3	Blocs 12
8.3.1	Bloc objet programmateur 12
8.3.2	Bloc objets gérés programmés 12
8.3.3	Comportement commun des programmeurs d'intervalles 13
8.3.4	Bloc de programmation journalière multiple 13
8.3.5	Bloc de programmation hebdomadaire multiple 13
8.3.6	Bloc de programmation mensuelle multiple 14
8.3.7	Bloc programmation périodique 15
8.3.8	Bloc mode resynchronisation 16
8.3.9	Bloc synchronisation de période 17
8.3.10	Bloc programmation d'opérations 17
8.3.11	Bloc notification d'opération 17

Remplacée par une version plus récente

Page

8.4	Propriétés des objets SMO	17
8.4.1	Blocs et attributs utilisés dans les objets SMO	17
8.4.2	Comportement de l'objet SMO pour la programmation d'intervalle.....	18
8.4.3	Comportement de l'objet SMO pour la programmation de déclenchement périodique et aperiodique	19
8.5	Conformité	19
8.6	Définitions génériques de la fonction de gestion d'objet.....	19
8.7	Définitions génériques de la fonction de gestion d'état	19
8.8	Définitions génériques de la fonction de gestion de compte rendu	19
8.9	Définitions génériques de la fonction de gestion de test	19
8.10	Définitions génériques de la fonction de récapitulation	19
9	Définition de service.....	20
10	Unités fonctionnelles	20
11	Protocole et syntaxe abstraite	20
11.1	Objets gérés	20
11.2	Attributs de gestion	21
11.3	Actions de gestion	21
11.4	Notifications de gestion.....	21
12	Relations avec d'autres fonctions.....	21
13	Conformité.....	21
13.1	Conformité statique	21
13.2	Conformité dynamique.....	22
13.3	Prescriptions relatives à la déclaration de conformité de réalisation de gestion.....	22
Annexe A	– Définition de l'information de gestion	23
A.1	Définition des classes d'objets	23
A.2	Liens de dénomination	24
A.3	Blocs de propriétés	25
A.4	Attributs.....	26
A.5	Notifications	28
A.6	Définitions ASN.1	28
Annexe B	– Formulaire MCS	31
B.1	Introduction	31
B.1.1	Purpose and structure	31
B.1.2	Instructions for completing the MCS proforma to produce an MCS	31
B.1.3	Symbols, abbreviations and terms.....	31
B.2	Identification of the implementation	31
B.2.1	Date of statement.....	31
B.2.2	Identification of the implementation	32
B.2.3	Contact	32
B.3	Identification of the Recommendation International Standard in which the management information is defined	32
B.3.1	Technical corrigenda implemented	32
B.3.2	Amendments implemented.....	32
B.4	Management conformance summary.....	32
Annexe C	– Formulaire PICS	37
C.1	Instructions for completing the PICS proforma	37
C.1.1	Purpose and structure	37
C.1.2	Symbols, abbreviation and terms	37
C.1.3	Nesting rules.....	37
C.1.4	Instructions for completing the PICS	37
C.2	Capabilities.....	37
C.2.1	Systems management functional unit negotiation support	37
C.2.2	MAPDU support	38

Remplacée par une version plus récente

Page

Annexe D – Formulaire MOCS	40
D.1 Introduction	40
D.1.1 Symbols, abbreviations and terms	40
D.1.2 Instructions for completing the MOCS proforma to produce a MOCS.....	40
D.2 Daily Scheduler managed object class	40
D.3 Weekly scheduler managed object class	44
D.4 Monthly scheduler managed object class	47
D.5 Periodic scheduler managed object class	51
D.6 Daily Operation Scheduler managed object class	55
D.7 Weekly operation scheduler managed object class	58
D.8 Monthly operation scheduler managed object class	62
D.9 Periodic operation scheduler managed object class	65
D.10 Operation result record managed object class	69
D.10.1 Statement of conformance to the managed object class	69
D.10.2 Packages	70
D.10.3 Attributes	71
Annexe E – Formulaire MIDS	73
Annexe F – Formulaire MRCS	74
F.1 Name bindings.....	74
F.2 Instructions for completing the MRCS proforma for name bindings to produce a MRCS	74
F.3 Statement of conformance to the name binding	74

Remplacée par une version plus récente

Résumé

La présente Recommandation | Norme internationale spécifie un modèle d'information de gestion pour la création et la gestion par un gestionnaire distant de programmes d'activité relatifs à l'activité de gestion; ceci englobe des éléments tels que des programmes de collecte de données ou d'essais périodiques ou programmés. La présente Spécification est d'application générique et peut être utilisée par de nombreux types d'applications. Il est prévu qu'elle soit utilisée par les RGT.

NORME INTERNATIONALE

RECOMMANDATION UIT-T

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

1 Domaine d'application

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

2.1 Recommandations CCITT/UIT-T et Normes internationales identiques

- Recommandation X.701 du CCITT (1992) | ISO/CEI 10040:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Aperçu général de la gestion des systèmes.*
- Recommandation X.720 du CCITT (1992) | ISO/CEI 10165-1:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: Modèle d'information de gestion.*

- Recommandation X.721 du CCITT (1992) | ISO/CEI 10165-2:1992, *Technologie de l'information – Interconnexion des systèmes ouverts – Structure des informations de gestion: Définition des informations de gestion.*
- Recommandation X.722 du CCITT (1992) | ISO/CEI 10165-4:1992, *Technologie 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 X.730 du CCITT (1992) | ISO/CEI 10164-1:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de gestion des objets.*
- Recommandation X.731 du CCITT (1992) | ISO/CEI 10164-2:1993, *Technologie de l'information – Interconnexion des systèmes ouverts – Gestion des systèmes: Fonction de gestion d'états.*
- Recommandation X.734 du CCITT (1992) | ISO/CEI 10164-5:1993, *Technologie de l'information – Interconnexion de systèmes ouverts – Gestion des 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 des systèmes: Fonction de récapitulation.*
- Recommandation UIT-T X.739 (1993) | ISO/CEI 10164-11:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestion des 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 des systèmes: Fonction de gestion des tests.*

2.2 Paires de Recommandations CCITT/UIT-T | Normes internationales équivalentes par leur contenu technique

- Recommandation X.200 du CCITT (1988), *Modèle de référence pour l'interconnexion des systèmes ouverts pour les applications du CCITT.*
ISO 7498:1984, *Systèmes de traitement de l'information – Interconnexion des systèmes ouverts – Modèle de Référence de base.*
- Recommandation X.208 du CCITT (1988), *Spécification de la syntaxe abstraite numéro un (ASN.1).*
ISO/CEI 8824:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – Spécification de la notation de syntaxe abstraite numéro un (ASN.1).*
- Recommandation X.209 du CCITT (1988), *Spécification des règles de codage de base pour la notation de syntaxe abstraite numéro un (ASN.1).*
ISO/CEI 8825:1990, *Technologies de l'information – Interconnexion de systèmes ouverts – Spécification de règles de base pour coder la notation de syntaxe abstraite numéro un (ASN.1).*
- Recommandation X.210 du CCITT (1988), *Conventions relatives à la définition de service des couches de l'interconnexion de systèmes ouverts.*
ISO/TR 8509:1987, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Conventions de service.*
- Recommandation X.291 du CCITT (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 – Cadre général et méthodologie des tests de conformité OSI – Partie 2: Spécification des suites de tests abstraites.*
- Recommandation UIT-T X.296¹⁾, *Cadre général et méthodologie des tests de conformité OSI pour les Recommandations sur les protocoles pour les applications de l'UIT-T – Déclarations de conformité d'instance.*

¹⁾ En projet.

ISO/CEI 9646-7:1995, *Technologies de l'information – Interconnexion de systèmes ouverts – Cadre général et méthodologie des tests de conformité OSI – Partie 7: Implementation Conformance Statement – Requirements and guidance on ICS and ICS proforma.*

- Recommandation X.700 du CCITT (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.*

- Recommandation X.710 du CCITT (1991), *Définition du service commun de transfert d'informations de gestion pour les applications du CCITT.*

ISO/CEI 9595:1991, *Technologies de l'information – Interconnexion de systèmes ouverts – Définition du service commun d'informations de gestion.*

- Recommandation X.711 du CCITT (1991), *Spécification du protocole commun de transfert d'informations de gestion pour les applications du CCITT.*

ISO/CEI 9596:1991, *Technologies de l'information – Interconnexion de systèmes ouverts – Protocole commun d'information de gestion – Partie 1: Spécification.*

2.3 Autres références

- ISO/CEI 9545:1989, *Technologies de l'information – Interconnexion de systèmes ouverts – 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. X.200 du CCITT | ISO 7498:

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

3.2 Définitions de la notation de syntaxe abstraite numéro un

La présente Recommandation | Norme internationale utilise les termes suivants définis dans la Rec. X.208 du CCITT | ISO/CEI 8824:

- identificateur d'objet.

3.3 Définitions du cadre général de gestion

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

- 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. X.710 du CCITT | ISO/CEI 9595:

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

3.5 Définitions de la présentation générale de la gestion-système

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

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

3.6 Définitions du modèle d'information de gestion

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

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

ASN.1	Notation de syntaxe abstraite numéro un (<i>abstract syntax notation one</i>)
CMIS	Service commun d'information de gestion (<i>common management information service</i>)
ICS	Déclaration de conformité d'information (<i>information 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>)
MRCS	Déclaration de conformité de relation de gestion (<i>management relationship conformance statement</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 B à F) utilisent les notations communes définies dans la Rec. X.291 du CCITT | ISO/CEI 9646-2 et dans la Rec. X.296 du CCITT | ISO/CEI 9646-7.

6 Prescriptions

En termes de fonctionnalité, les prescriptions à satisfaire sont de:

- 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.

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.

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 7.3 deux autres modes de programmation: la programmation d'intervalle et la programmation périodique. 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 interne 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 du système. 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.

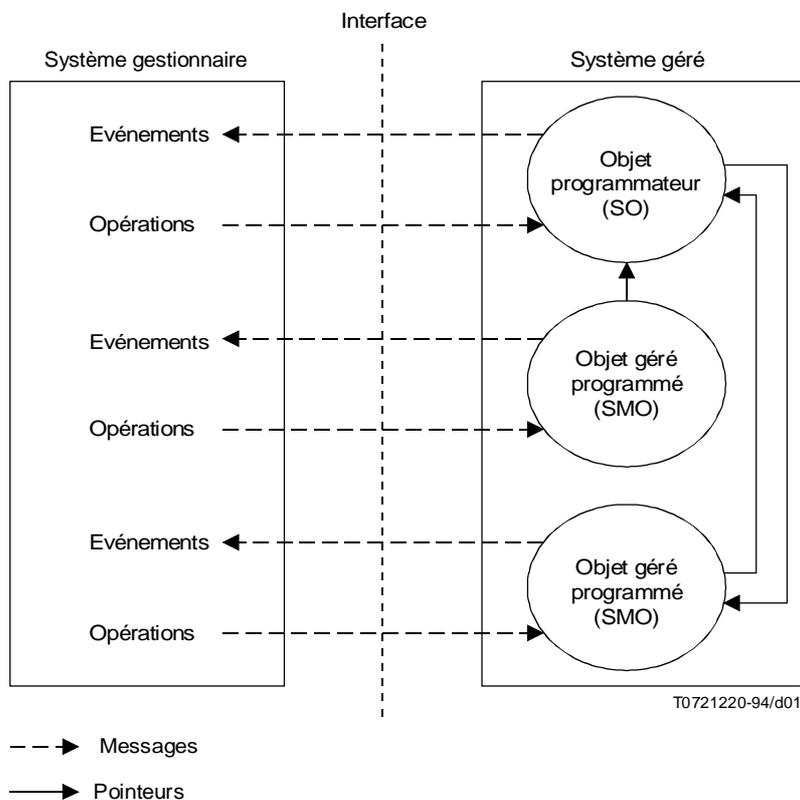


Figure 1 – Modèle d'objet programmeur

7.3 Modes de programmation

Il existe trois modes spécifiques de programmation: la programmation d'intervalle, la programmation de déclenchement et la programmation d'opération. Le présent document décrit trois modes de programmation d'intervalle – la programmation d'intervalle journalière, hebdomadaire et mensuelle (voir 7.3.1); deux modes de programmation de déclenchement (voir 7.3.2) et la programmation d'opération. Ces modes de programmation sont définis par des blocs 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 blocs de programmation sont définis dans la Rec. X.734 du CCITT | 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.

L'opération 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 aperiodique

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 aperiodique 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 aperiodique journalière, hebdomadaire ou mensuelle est spécifié par la sélection, à l'intérieur de la classe d'objets programmeur appropriée, des paramètres d'instant 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.

7.4 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 du programmeur externe). D'une manière optionnelle, l'objet SO peut avoir un attribut pointant sur l'objet SMO (l'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.

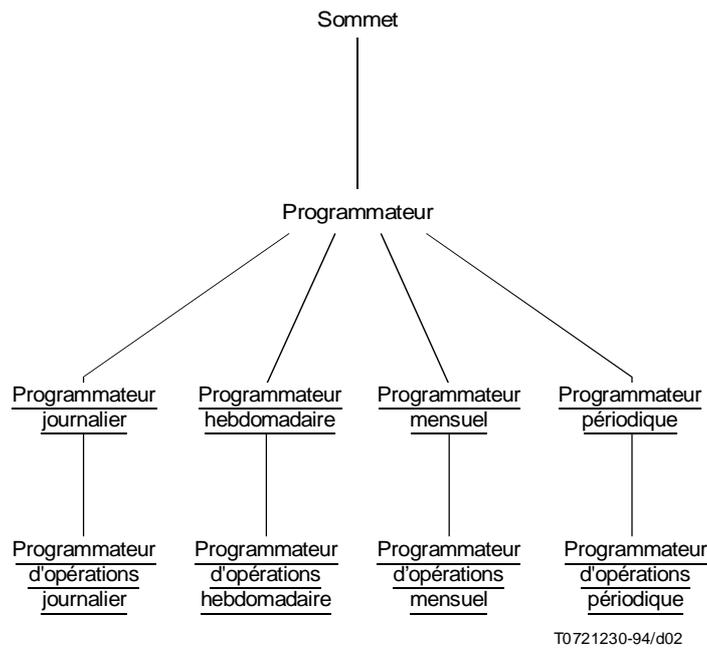
8 Définitions génériques

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 bloc 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 bloc avec les activités appropriées dans la clause de comportement. (Les blocs de programmation journalière et hebdomadaire sont définis dans la Rec. X.734 du CCITT | 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.



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

Figure 2 – Structure d'héritage de la classe programmeur

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

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 Blocs du programmeur

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

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

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

- bloc 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 Blocs du programmeur journalier

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

- bloc 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 Blocs du programmeur hebdomadaire

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

- bloc programmation hebdomadaire multiple.

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 Blocs du programmeur mensuel

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

- bloc 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 Blocs du programmeur périodique

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

- bloc programmation périodique.

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

- bloc mode resynchronisation;
- bloc 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 Blocs du programmeur d'opération journalier

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

- bloc programmation d'opérations.

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

- bloc résultat d'opération.

Le bloc 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 Blocs du programmeur d'opération hebdomadaire

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

- bloc programmation d'opérations.

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

- bloc résultat d'opération.

Le bloc 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 Blocs du programmeur d'opération mensuel

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

- bloc programmation d'opérations.

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

- bloc résultat d'opération.

Le bloc 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 Blocs du programmeur d'opération périodique

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

- bloc programmation d'opérations.

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

- bloc résultat d'opération.

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

8.3 Blocs

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

8.3.1 Bloc objet programmeur

8.3.1.1 Aperçu général

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

8.3.1.2 Attributs du bloc objet programmeur

Le bloc 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. X.731 du CCITT | ISO/CEI 10164-2.
- c) *état opérationnel* – Cet attribut est défini dans la Rec. X.731 du CCITT | ISO/CEI 10164-2.

8.3.1.3 Notifications du bloc objet programmeur

Le bloc objet programmeur contient les notifications suivantes:

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

8.3.1.4 Comportement du bloc objet programmeur

Le bloc 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 bloc 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 bloc objet programmeur.

8.3.2 Bloc objets gérés programmés

8.3.2.1 Aperçu général

Le bloc 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 bloc objets gérés programmés

Le bloc 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 bloc 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 Bloc de programmation journalière multiple

8.3.4.1 Aperçu général

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

8.3.4.2 Attributs du bloc de programmation journalière multiple

Le bloc 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 bloc de programmation journalière multiple

Le bloc 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 Bloc de programmation hebdomadaire multiple

8.3.5.1 Aperçu général

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

8.3.5.2 Attributs du bloc programmation hebdomadaire multiple

Le bloc programmation hebdomadaire multiple possède l'attribut:

- *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 séquence hebdomadaire 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 séquence hebdomadaire 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 bloc programmation hebdomadaire multiple

Le bloc 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 mis en œuvre lorsque l'objet est activé. Chacun des programmes suivants sera mis en œuvre 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 bloc durée.

Le composant intervalles de jour à l'intérieur de l'attribut séquence hebdomadaire 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 hebdomadaire 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 hebdomadaire est modifié.

8.3.6 Bloc de programmation mensuelle multiple

8.3.6.1 Aperçu général

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

8.3.6.2 Attributs du bloc programmation mensuelle multiple

Le bloc 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 bloc programmation mensuelle multiple

Le bloc 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 bloc 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 Bloc programmation périodique

8.3.7.1 Aperçu général

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

8.3.7.2 Attributs du bloc programmation périodique

Le bloc programmation périodique possède l'attribut:

- *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 bloc programmation périodique

Le bloc 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 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 resynchronisation est faux, ou sera synchronisé sur l'instant de reprise si le mode de resynchronisation est vrai.

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

8.3.8.1 Aperçu général

Le bloc mode 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 bloc mode resynchronisation

Le bloc mode 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 bloc mode resynchronisation

L'attribut mode 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. L'exemple de la Figure 3 suppose que l'état opérationnel est en service.

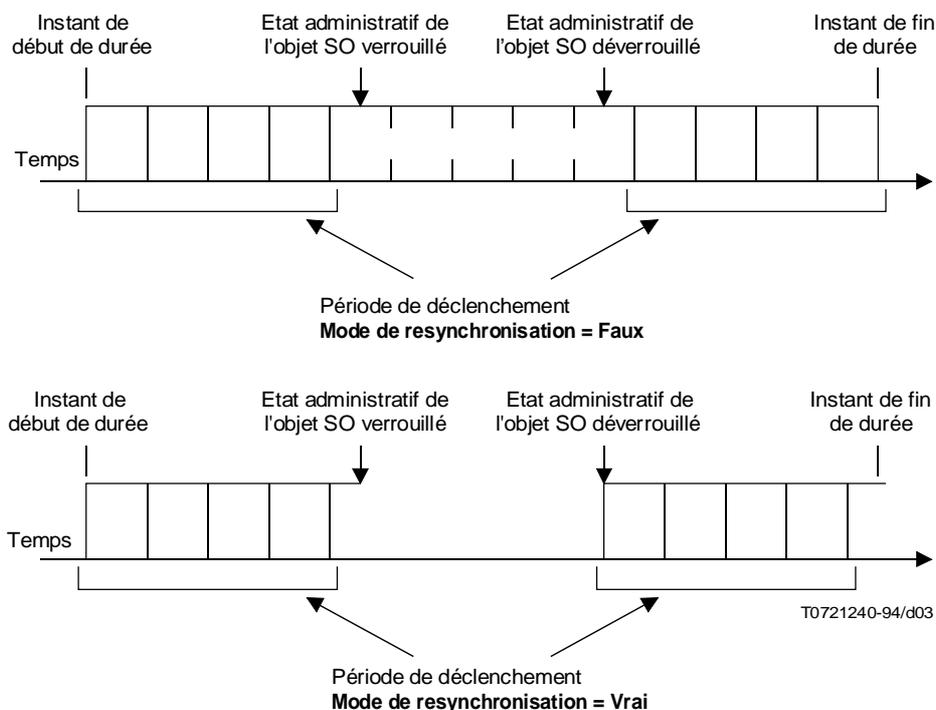


Figure 3 – Exemple d'utilisastion de l'attribut mode resynchronisation

8.3.9 Bloc synchronisation de période

Le bloc 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 Bloc programmation d'opérations

8.3.10.1 Aperçu général

Le bloc de programmation d'opérations identifie les opérations spécifiques devant être programmées dans l'objet géré programmé. Un objet SO contenant ce bloc 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 bloc programmation d'opérations

Le bloc 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 Bloc notification d'opération

8.3.11.1 Aperçu général

Le bloc 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 bloc notification d'opération

Le bloc notification d'opération contient la notification suivante:

- *résultat d'opération* – La notification 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.

8.4 Propriétés des objets SMO

8.4.1 Blocs 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 bloc programmeur externe.

Le bloc programmeur externe est défini dans la Rec. X.734 du CCITT | 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. X.731 du CCITT | 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 Bloc fenêtre requise

Le bloc fenêtre requise est défini dans la Rec. X.745 du CCITT | ISO/CEI 10164-12. Ce bloc 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 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.

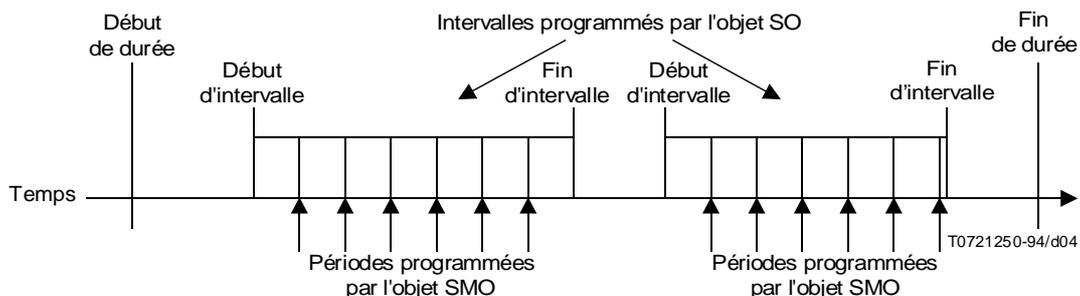


Figure 4 – Exemple de programmation périodique avec intervalle programmé

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

Quand un objet géré programmeur du type déclencheur (périodique ou aperiodique) 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 aperiodique est suspendue, les activités programmées au sein des objets SMO ne seront pas déclenchées.

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. X.722 du CCITT | 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. X.730 du CCITT | 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. X.731 du CCITT | 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. X.734 du CCITT | ISO/CEI 10164-5:

- bloc durée,
- bloc 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:

- bloc 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:

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

10 Unités fonctionnelles

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

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

Les unités fonctionnelles suivantes définies dans la Rec. X.731 du CCITT | 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.

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 hebdomadaire,
- séquence mensuelle,
- période de temps.

11.3 Actions de gestion

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

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. X.730 du CCITT | 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. X.731 du CCITT | 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, blocs et attributs définis dans la Rec. X.721 du CCITT | ISO/CEI 10165-2.

13 Conformité

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

13.1 Conformité statique

La réalisation 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 B.1. Si une déclaration de conformité est faite pour la prise en charge du rôle de gestionnaire, la réalisation doit prendre en charge au moins un des attributs, notifications, actions ou objets gérés décrits dans le Tableau B.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, la réalisation prendra en charge au moins un des attributs, actions ou objets gérés décrits dans le Tableau B.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.

La réalisation prendra en charge la syntaxe de transfert dérivée des règles de codage spécifiées dans la Rec. X.209 du CCITT | ISO/CEI 8825 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 réalisation 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 de réalisation de gestion

Tout formulaire de récapitulatif MCS (conformité de gestion), ou de déclaration PICS (conformité d'instance 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 B, C, D et E et en conserver la numérotation de tableaux et d'articles, et n'en différera que par la pagination et les en-têtes de pages.

Le fournisseur d'une réalisation déclarée conforme à la présente Recommandation | Norme internationale remplira une copie du récapitulatif de conformité de gestion MCS fourni dans l'Annexe A 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 réalisation 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 la réalisation.

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 de l'information 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 "An instance supports it.";
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 "an instance supports it and the periodSynchronizationPackage package is not present",
 "Rec. X.738 | ISO/IEC 10164-13":periodSynchronizationPackage PRESENT IF "Synchronization to a specified time other than the duration start time is required, and the resynchronizeMode package is not present.";
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 "the results of the operation need to be reported or the operation performed is a GET operation";
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 "the results of the operation
    performed is a GET operation";
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 "the results of the operation
    performed is a GET operation";
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 "the results of the operation
    performed is a GET operation";
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
    operationResultRecordBehaviour BEHAVIOUR
    DEFINED AS                    "see 8.3.11.2";
ATTRIBUTES
    operationResult              GET;;;
REGISTERED AS {schedMo 10};
```

A.2 Liens de dénomination

Additional Name Bindings may be defined and registered for each scheduler object class.

A.2.1 Liens de dénomination de programmeurs

The following NAME-BINDING template provides the name binding currently defined for naming instances of the scheduler managed object classes.

```
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};
```

A.3 Blocs de propriétés**A.3.1 Bloc de programmation journalière multiple**

```

multipleDailyScheduling          PACKAGE
  BEHAVIOUR                      multipleDailySchedulingBehaviour  BEHAVIOUR
  DEFINED AS                      "See 8.3.3 and 8.3.4.3";
  ATTRIBUTES
  sequenceOfDays                  DEFAULT VALUE
                                   Schedule-ASN1Module.defaultSequenceOfDays

  GET-REPLACE
  ADD-REMOVE;
REGISTERED AS    {schedPkg 1};

```

A.3.2 Bloc de programmation mensuelle multiple

```

multipleMonthlyScheduling        PACKAGE
  BEHAVIOUR                      multipleMonthlySchedulingBehaviour  BEHAVIOUR
  DEFINED AS                      "See 8.3.3 and 8.3.6.3";
  ATTRIBUTES
  sequenceOfMonths                DEFAULT VALUE
                                   Schedule-ASN1Module.defaultSequenceOfMonths

  GET-REPLACE
  ADD-REMOVE;
REGISTERED AS    {schedPkg 2};

```

A.3.3 Bloc de programmation hebdomadaire multiple

```

multipleWeeklyScheduling         PACKAGE
  BEHAVIOUR                      multipleWeeklySchedulingBehaviour  BEHAVIOUR
  DEFINED AS                      "See 8.3.3 and 8.3.5.3";
  ATTRIBUTES
  sequenceOfWeeks                 DEFAULT VALUE
                                   Schedule-ASN1Module.defaultSequenceOfWeeks

  GET-REPLACE
  ADD-REMOVE;
REGISTERED AS    {schedPkg 3};

```

A.3.4 Bloc de programmation périodique

```

periodicSchedulingPackage        PACKAGE
  BEHAVIOUR                      periodicSchedulingBehaviour        BEHAVIOUR
  DEFINED AS                      "See 8.3.7.3";
  ATTRIBUTES
  timePeriod                       DEFAULT VALUE
                                   Schedule-ASN1Module.defaultTimePeriod

  GET ADD-REMOVE;
REGISTERED AS    {schedPkg 4};

```

A.3.5 Bloc de mode de resynchronisation

```

resynchronizeModePackage         PACKAGE
  BEHAVIOUR                      resynchronizeModeBehaviour        BEHAVIOUR
  DEFINED AS                      "See 8.3.8.3";
  ATTRIBUTES
  resynchronizeMode                GET-REPLACE;
REGISTERED AS    {schedPkg 5};

```

A.3.6 Bloc d'objets gérés programmés

```

scheduledManagedObjectsPackage  PACKAGE
  BEHAVIOUR                      scheduledManagedObjectBehaviour  BEHAVIOUR
  DEFINED AS                      "See 8.3.2";
  ATTRIBUTES
  scheduledManagedObjects          GET;
REGISTERED AS    {schedPkg 6};

```

A.3.7 Bloc d'objet programmeur

```

schedulerObjectPackage          PACKAGE
    BEHAVIOUR                    schedulerObjectBehaviour  BEHAVIOUR
    DEFINED AS                    "See 8.3.1.4";;
    ATTRIBUTES
        schedulerID              GET,
        "Rec. X.721 | ISO/IEC 10165-2:1992":administrativeState
                                GET-REPLACE,
        "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 Bloc de programmation d'opérations

```

operationsSchedulingPackage      PACKAGE
    BEHAVIOUR                    operationsSchedulingBehaviour  BEHAVIOUR
    DEFINED AS                    "See 8.3.10";;
    ATTRIBUTES
        operationSpecifications  GET-REPLACE ADD-REMOVE;
REGISTERED AS    {schedPkg 8};
    
```

A.3.9 Bloc de notification d'opération

```

operationNotificationPackage     PACKAGE
    BEHAVIOUR                    operationNotificationBehaviour  BEHAVIOUR
    DEFINED AS                    "See 8.3.11";;
    NOTIFICATIONS
        operationResult;
REGISTERED AS    {schedPkg 9};
    
```

A.4 Attributs

A.4.1 Attribut nom de programmeur externe

This attribute is included in Scheduled Managed objects. It specifies the SO instance that controls the activity in the SMO.

```

externalSchedulerName           ATTRIBUTE
    BEHAVIOUR                    externalSchedulerNameBehaviour  BEHAVIOUR
    DEFINED AS                    "See 8.4.1.1";;
    WITH ATTRIBUTE SYNTAX
        Schedule-ASN1Module.ExternalSchedulerName;
    MATCHES FOR EQUALITY;
REGISTERED AS    {schedAtt 1};
    
```

A.4.2 Attribut en service

This attribute is included in SMOs. It specifies the current status of the activity in the SMO.

```

onDuty    ATTRIBUTE
    BEHAVIOUR                    onDutyBehaviour    BEHAVIOUR
    DEFINED AS                    "See 8.4.1.2";;
    WITH ATTRIBUTE SYNTAX
        Schedule-ASN1Module.OnDuty;
    MATCHES FOR EQUALITY;
REGISTERED AS    {schedAtt 2};
    
```

A.4.3 Attribut mode de resynchronisation

This attribute defines the mode of synchronization of a periodic scheduler's triggering periods.

```

resynchronizeMode              ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
        Schedule-ASN1Module.ResynchronizeMode;
    MATCHES FOR EQUALITY;
    
```

BEHAVIOUR
DEFINED AS resynchronizeModeBehaviour **BEHAVIOUR**
 "See 8.3.8.2";;
REGISTERED AS {schedAtt 3};

A.4.4 Attribut objets gérés programmés

This attribute is defined in the Scheduler object to specify the SMO instances and the attribute identifier of the attribute associated with the activities within those instances that are controlled by the SO.

scheduledManagedObjects **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Schedule-ASN1Module.ScheduledManagedObjectsList;
MATCHES FOR SET-COMPARISON, SET-INTERSECTION;
BEHAVIOUR scheduledManagedObjectsBehaviour **BEHAVIOUR**
DEFINED AS "See 8.3.2.2";;
REGISTERED AS {schedAtt 4};

A.4.5 Attribut identificateur de programmeur

This attribute is the distinguished attribute for naming instances of a SO.

schedulerID **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Schedule-ASN1Module.SimpleNameType;
MATCHES FOR EQUALITY, SUBSTRINGS;
BEHAVIOUR schedulerIDBehaviour **BEHAVIOUR**
DEFINED AS "See 8.3.1.2";;
REGISTERED AS {schedAtt 5};

A.4.6 Attribut séquence journalière

This structured attribute defines a sequence of intervals of day.

sequenceOfDays **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Schedule-ASN1Module.SequenceOfDays;
MATCHES FOR EQUALITY;
BEHAVIOUR sequenceOfDaysBehaviour **BEHAVIOUR**
DEFINED AS "See 8.3.4.2";;
REGISTERED AS {schedAtt 6};

A.4.7 Attribut séquence mensuelle

This structured attribute defines a sequence of month masks.

sequenceOfMonths **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Schedule-ASN1Module.SequenceOfMonths;
MATCHES FOR EQUALITY;
BEHAVIOUR sequenceOfMonthsBehaviour **BEHAVIOUR**
DEFINED AS "See 8.3.6.2";;
REGISTERED AS {schedAtt 7};

A.4.8 Attribut séquence hebdomadaire

This structured attribute defines a sequence of week masks.

sequenceOfWeeks **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Schedule-ASN1Module.SequenceOfWeeks;
MATCHES FOR EQUALITY;
BEHAVIOUR sequenceOfWeeksBehaviour **BEHAVIOUR**
DEFINED AS "See 8.3.5.2";;
REGISTERED AS {schedAtt 8};

A.4.9 Attribut période de temps

This attribute defines the length of the time period for the periodic triggering of an activity in a SMO by the Periodic scheduling object.

timePeriod **ATTRIBUTE**
WITH ATTRIBUTE SYNTAX Schedule-ASN1Module.TimePeriod;
MATCHES FOR EQUALITY, ORDERING;
BEHAVIOUR timePeriodBehaviour **BEHAVIOUR**
DEFINED AS "See 8.3.7.2";;
REGISTERED AS {schedAtt 9};

A.4.10 Attribut spécifications d'opération

This attribute defines the operations which may be scheduled for a SMO by an operations scheduling object.

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

A.4.11 Attribut résultat d'opération

This attribute is included in operation result record objects. It specifies the result of operations performed by operations schedulers.

```
operationResult ATTRIBUTE
    BEHAVIOUR operationResultBehaviour BEHAVIOUR
    DEFINED AS "See 8.3.11.2";
    WITH ATTRIBUTE SYNTAX
        Schedule-ASN1Module.OperationResult;
    MATCHES FOR EQUALITY;
REGISTERED AS {schedAtt 11};
```

A.5 Notifications

A.5.1 Notification de résultat d'opération

```
operationResult NOTIFICATION
    WITH INFORMATION SYNTAX
        Schedule-ASN1Module.OperationResult
        AND ATTRIBUTE IDS
            OperationResult operationResult;
    BEHAVIOUR operationResultBehaviour;
REGISTERED AS {schedNotif 1};
```

A.6 Définitions ASN.1

```
Schedule-ASN1Module { joint-iso-ccitt ms(9) function(2) part15(15) modules (1) }
DEFINITIONS IMPLICIT TAGS ::=
```

```
BEGIN
```

```
-- EXPORTS everything
```

```
IMPORTS
```

```
Attribute,AttributeId,ObjectInstance,ActionResult,SetResult,SetListError,
ActionError,ModifyOperator,ActionInfo,GetResult,GetListResult,GetArgument FROM CMIP-1
{joint-CCITT-iso ms(9) cmip(1) modules(0) protocol(3)}
```

```
SimpleNameType,defaultStopTime FROM Attribute-ASN1Module
{joint-CCITT-iso ms(9) smi(3) part2(2) asn1Module(2) 1}
```

```
TimePeriod FROM metricModule
{joint-CCITT-iso ms(9) smf(4) part11(11) asn1Module(2) 1};
```

```
schedMo OBJECT IDENTIFIER ::=
    {joint-iso-ccitt ms(9) function(2) part15(15) managedObjectClass(3)}
schedAtt OBJECT IDENTIFIER ::=
    {joint-iso-ccitt ms(9) function(2) part15(15) attribute(7)}
schedNotif OBJECT IDENTIFIER ::=
    {joint-iso-ccitt ms(9) function(2) part15(15) notification(10)}
schedPkg OBJECT IDENTIFIER ::=
    {joint-iso-ccitt ms(9) function(2) part15(15) package(4)}
schedNb OBJECT IDENTIFIER ::=
    {joint-iso-ccitt ms(9) function(2) part15(15) nameBinding(6)}
```

-- default value definitions

```
defaultDaysOfMonth      DaysOfMonth ::= {{'11111111111111111111111111111111'B},
                                         {'11111111111111111111111111111111'B}}
```

```
defaultIntervalsOfDayWps IntervalsOfDayWps ::= {{
    intervalStart {hour 0, minute 0, second 0},
    intervalEnd {hour 0, minute 0, second 0}}}
```

```
defaultResynchronizeMode ResynchronizeMode ::= FALSE
```

```
defaultSequenceOfDays SequenceOfDays ::= {defaultIntervalsOfDayWps}
```

```
defaultSequenceOfMonths SequenceOfMonths ::= {{
    daysOfMonth      defaultDaysOfMonth,
    intervalsOfDayWps defaultIntervalsOfDayWps }}
```

```
defaultSequenceOfWeeks SequenceOfWeeks ::= {defaultWeekMaskWps}
```

```
defaultTimePeriod TimePeriod ::= second 0
```

```
defaultWeekMaskWps WeekMaskWps ::= {{
    daysOfWeek      '1111111'B,
    intervalsOfDayWps defaultIntervalsOfDayWps}}
```

-- supporting productions

```
DaysOfMonth ::=SEQUENCE {
    daysFromFirst BITSTRING (SIZE (31)) DEFAULT {'B},
    daysFromLast BITSTRING (SIZE (31)) DEFAULT {'B}}
```

```
DaysOfWeek ::=BIT STRING {
    sunday(0),monday(1),tuesday(2),wednesday(3),thursday(4),
    friday(5),saturday(6)} (SIZE(7))
```

```
ExternalSchedulerName ::= SET OF ObjectInstance
```

```
IntervalsOfDayWps ::= SET OF SEQUENCE {
    intervalStart Time24Wps,
    intervalEnd Time24Wps}
```

```
MonthMask ::= SET OF SEQUENCE {
    daysOfMonth DaysOfMonth,
    timesOfDayWps TimesOfDayWps}
```

-- Specifies the TimesOfDayWps for different days of the same month if
-- they have different intervals.

```
ModificationList ::= SET OF SEQUENCE{
    modifyOperator [2] IMPLICIT ModifyOperator DEFAULT replace,
    attributeId AttributeId,
    attributeValue ANY DEFINED BY attributeId OPTIONAL
    -- absent for setToDefault }
```

```
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 [1] IMPLICIT SetListError,
    actionResult [4] IMPLICIT ActionResult,
    commonError [5] IMPLICIT CommonError}
```

```
CommonError ::=SEQUENCE{
    managedObjectClass ObjectClass,
    managedObjectInstance ObjectInstance,
```

```

errorId                Remote-Operation-Notation.ERROR,
errorValue             ANY DEFINED BY errorId OPTIONAL}

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

ResynchronizeMode ::= BOOLEAN

SequenceOfDays ::= SEQUENCE OF IntervalsOfDayWps

SequenceOfWeeks ::= SEQUENCE OF WeekMaskWps

SequenceOfMonths ::= SEQUENCE OF MonthMask

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}

TimesOfDay ::= CHOICE {
    intervalsOfDayWps  IntervalsOfDayWps,
    triggerTimes       TriggerTimes}

TriggerTimes ::= SET OF Time24Wps

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

END    -- end of supporting productions

```

Annexe B

Formulaire MCS²⁾

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

B.1 Introduction**B.1.1 Purpose and structure**

The management conformance summary (MCS) is a statement by a supplier that identifies an implementation and provides information on whether the implementation claims conformance to any of the listed set of documents that specify conformance requirements to OSI management.

The MCS proforma is a document, in the form of a questionnaire that, when completed by the supplier of an implementation, becomes the MCS.

B.1.2 Instructions for completing the MCS proforma to produce an MCS³⁾

The supplier of the implementation shall enter an explicit statement in each of the boxes provided. Specific instruction is provided in the text which precedes each table.

B.1.3 Symbols, abbreviations and terms

For all annexes of this Recommendation | International Standard, the following common notations, defined in CCITT Rec. X.291 | ISO/IEC 9646-2 and CCITT Rec. X.296 | ISO/IEC 9646-7 are used for the Status column:

- m Mandatory;
- o Optional;
- c Conditional;
- x Prohibited;
- Not applicable or out of scope.

NOTES

- 1 'c', 'm', and 'o' are prefixed by a 'c:' when nested under a conditional or optional item of the same table;
- 2 'o' may be suffixed by '.n' (where n is a unique number) for mutually exclusive or selectable options among a set of status values. Support of at least one of the choices (from the items with the same values of n) is required.

For all annexes of this Recommendation | International Standard, the following common notations, defined in CCITT Rec. X.291 | ISO/IEC 9646-2 and CCITT Rec. X.296 | ISO/IEC 9646-7 are used for the Support column:

- Y Implemented;
- N Not implemented;
- No answer required;
- Ig The item is ignored (i.e. processed syntactically but not semantically).

B.2 Identification of the implementation**B.2.1 Date of statement**

The supplier of the implementation shall enter the date of this statement in the box below. Use the format DD-MM-YYYY.

Date of statement

²⁾ Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MCS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété.

³⁾ Les instructions pour compléter le formulaire MCS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 10165-6.

B.2.2 Identification of the implementation

The supplier of the implementation shall enter information necessary to uniquely identify the implementation and the system(s) in which it may reside, in the box below.

B.2.3 Contact

The supplier of the implementation shall provide information on whom to contact if there are any queries concerning the content of the MCS, in the box below.

B.3 Identification of the Recommendation | International Standard in which the management information is defined

The supplier of the implementation shall enter the title, reference number and date of the publication of the Recommendation | International Standard which specifies the management information to which conformance is claimed, in the box below.

Recommendation | International Standard to which conformance is claimed

B.3.1 Technical corrigenda implemented

The supplier of the implementation shall enter the reference numbers of implemented technical corrigenda which modify the identified Recommendation | International Standard, in the box below.

B.3.2 Amendments implemented

The supplier of the implementation shall state the titles and reference numbers of implemented amendments to the identified Recommendation | International Standard, in the box below.

B.4 Management conformance summary

The supplier of implementation shall state the capabilities and features supported and provide summary of conformance claims to Recommendations | International Standards using the tables in this annex.

The supplier of the implementation shall specify the roles that are supported, in Table B.1.

Table B.1 – Roles

Index	Roles supported	Status	Support	Additional information
1	Manager role support	o.1		
2	Agent role support	o.1		

The supplier of the implementation shall specify support for the general purpose platform profile and systems management functional units, in Table B.2.

Table B.2 – General purpose platform profile

Index	Capability	Manager		Agent		Additional information
		Status	Support	Status	Support	
1	Support for the general purpose platform profile	o		o		
2	objectEvents functional unit	c1		c2		
3	allEvents functional unit	c1		c2		
4	control functional unit	c1		c2		
5	monitor functional unit	c1		c2		
6	stateChange functional unit	c1		c2		
c1 If B2/1a then m else o. c2 If B2/1b then m else o.						

The supplier of the implementation shall specify support for management information in the manager role, in Table B.3.

Table B.3 – Manager role minimum conformance requirement

Index	Item	Status	Support	Table reference	Additional information
1	OperationResultNotification	c3			
2	dailyScheduling Managed Object	c3			
3	weeklyScheduling Managed Object	c3			
4	monthlyScheduling Managed Object	c3			
5	periodicScheduling Managed Object	c3			
6	dailyOperationScheduling Managed Object	c3			
7	weeklyOperationScheduling Managed Object	c3			
8	monthlyOperationScheduling Managed Object	c3			
9	periodicOperationScheduling Managed Object	c3			
10	externalSchedulerName	c3			
11	onDuty	c3			
c3 If B1/1a then o.2 else –. NOTES 1 At least one applicable systems management operation shall be supported for at least one of the managed objects listed in items 2 through 9 and the attributes in items 10 and 11 (see clause 13). 2 The Table reference column in the above table is the (notification, attribute, action, name binding or managed object class) table reference that the supplier of implementation has filled in.					

The supplier of the implementation shall specify support for management information in the agent role, in Table B.4.

Table B.4 – Agent role minimum conformance requirement

Index	Item	Status	Support	Table reference	Additional information
1	OperationResultNotification	c4			
2	dailyScheduling Managed Object	c4			
3	weeklyScheduling Managed Object	c4			
4	monthlyScheduling Managed Object	c4			
5	periodicScheduling Managed Object	c4			
6	dailyOperationScheduling Managed Object	c4			
7	weeklyOperationScheduling Managed Object	c4			
8	monthlyOperationScheduling Managed Object	c4			
9	periodicOperationScheduling Managed Object	c4			
10	externalSchedulerName attribute	c4			
11	onDuty attribute	c4			
12	multipleDailyScheduling package	c4			
13	multipleWeeklyScheduling package	c4			
14	multipleMonthlyScheduling package	c4			
15	periodicScheduling package	c4			
c4 If B1/2a then o.3 else –. NOTE – The Table reference column in the above table is the (notification, attribute, action, name binding or managed object class) table reference of the MOCS supplied by the supplier of the managed object which claims to import the notification or attribute from this Recommendation International Standard.					

Table B.5 – Logging of event records

Index		Status	Support	Additional information
1	Does the implementation support logging of event records in agent role?	o		

NOTE – Conformance to this Recommendation | International Standard does not require conformance to CCITT Rec. X.735 | ISO/IEC 10164-6.

The supplier of the implementation shall provide information on claims of conformance to any of the Recommendation | International Standards summarized in Tables B.6 to B.8. For each Recommendation | International Standard that the supplier of the implementation claims conformance to, the corresponding conformance statement(s) shall be completed, or referenced by, the MCS. The supplier of the implementation shall complete the Support, Table numbers and Additional information columns.

In Tables B.6, B.7 and B.8, the Status column is used to indicate whether the supplier of the implementation is required to complete the referenced tables or referenced items. Conformance requirements are as specified in the referenced tables or referenced items and are not changed by the value of the MCS Status column. Similarly, the Support column is used by the supplier of the implementation to indicate completion of the referenced tables or referenced items.

Table B.6 – PICS support summary

Index	Identification of the document that includes the PICS proforma	Table numbers of PICS proforma	Description	Constraints and values	Status	Support	Table numbers of PICS	Additional information
1	ITU-T Rec X.746 ISO/IEC 10164-15	Annex C all tables	–	–	m			
2	CCITT Rec X.730 ISO/IEC 10164-1	Annex E all tables	SM application Context	–	m			
3	CCITT Rec X.730 ISO/IEC 10164-1	Annex B MAPDU tables	Object Management MAPDUs	–	m			
4	CCITT Rec X.731 ISO/IEC 10164-2	Annex B MAPDU tables	State Management MAPDUs	–	m			

Table B.7 – MOCS support summary

Index	Identification of the document that includes the MOCS proforma	Table numbers of MOCS proforma	Description	Constraints and values	Status	Support	Table numbers of MOCS	Additional information
1	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Daily Scheduler	–	o.5			
2	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Weekly Scheduler	–	o.5			
3	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Monthly Scheduler	–	o.5			
4	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Periodic Scheduler	–	o.5			
5	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Daily Operation Scheduler	–	o.5			
6	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Weekly Operation Scheduler	–	o.5			
7	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Monthly Operation Scheduler	–	o.5			
8	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Periodic Operation Scheduler	–	o.5			
9	ITU-T Rec X.746 ISO/IEC 10164-15	Annex D all tables	Operation Result Record	–	o.5			

Table B.8 – MRCS support summary

Index	Identification of the document that includes the MRCS proforma	Table numbers of MRCS proforma	Description	Constraints and values	Status	Support	Table numbers of MRCS	Additional information
1	CCITT Rec X.735 ISO/IEC 10164-6	Annex F	logRecord-Log	–	o			
2	CCITT Rec X.746 ISO/IEC 10164-15	Annex F	scheduler-System	–	o			

Annexe C

Formulaire PICS⁴⁾

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

C.1 Instructions for completing the PICS proforma**C.1.1 Purpose and structure**

The purpose of this PICS proforma is to provide a mechanism whereby a supplier of an implementation of ITU-T Rec. X.746 | ISO/IEC 10164-15 may provide information in a standard form. The PICS proforma is subdivided into clauses for the following categories of information:

- protocol details;
- overall conformance claim;
- implementation capabilities.

C.1.2 Symbols, abbreviation and terms

The PICS proforma contained in this annex is comprised of information in a tabular form in accordance with the guidelines presented in CCITT Rec. X.291 | ISO/IEC 9646-2.

The notations used in the Status and Support columns are specified in B.1.3.

Within this PICS proforma, space has been provided for the supplier of the implementation to specify support for individual items and if appropriate provide additional information. It is recommended that references to additional specifications are included where appropriate (for example, to list the OBJECT IDENTIFIER values and/or ranges supported), and that these additional specifications be appended to the completed PICS proforma.

C.1.3 Nesting rules

In the “Status” column of the tables in this Recommendation | International Standard, a mandatory element contained within an optional or conditional constructor parameter is mandatory only if the option or condition is taken. The “c:” notation, specified in CCITT Rec X.296 | ISO/IEC 9646-7 is used to express these nesting rules.

C.1.4 Instructions for completing the PICS

The supplier of the implementation shall enter an explicit statement in each of the boxes provided using the notation described in B.1.3. Specific instruction is provided in the text which precedes each table.

C.2 Capabilities**C.2.1 Systems management functional unit negotiation support**

The supplier of the implementation shall state the capability for negotiating the use of the object management functional units, in Table C.1.

Table C.1 – SMFU negotiation support

Index	Negotiation capability	Status	Support	Additional information
1	Does the implementation support the negotiation of the object management functional units?	o		

The tables for the functional unit negotiation mechanism are specified in Annex E of the first amendment to CCITT Rec. X.730 | ISO/IEC 10164-1.

⁴⁾ Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire PICS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété.

C.2.2 MAPDU support

The supplier of the implementation shall state support for the MAPDU in the management role(s) for which conformance is claimed, in Table C.2.

Table C.2 – MAPDUs

Index	MAPDU (agent sending) (manager receiving)	Status	Support	Additional information
1	OperationResultNotification (agent sending)	c1		
2	OperationResultNotification (manager receiving)	c2		
c1 If B.3/1a then m else –. c2 If B.2/1a then m else –.				

A standard mechanism for configuring event forwarding characteristics of an open system has been defined in CCITT Rec. X.734 | ISO/IEC 10164-5. For systems not using this mechanism, the supplier of the implementation shall state the condition under which event reports will be forwarded by the system, in the box below.

--

If support for the OperationResultNotification MAPDU in the agent role is claimed (B.3/1a), then the supplier of the implementation shall state whether or not each parameter of the MAPDU is supported in Table C.3.

Table C.3 – OperationResultNotification MAPDU (Agent sending)

Index	Parameter name	Constraints and values	Status	Support	Additional information
1	OperationResult		m		
1.1	getResult	ManagedObjectClass and ManagedObjectInstance parameters shall be present	o.5		
1.2	getListError	ManagedObjectClass and ManagedObjectInstance parameters shall be present	o.5		
1.3	setResult	ManagedObjectClass and ManagedObjectInstance parameters shall be present	o.5		
1.4	setListError	ManagedObjectClass and ManagedObjectInstance parameters shall be present	o.5		
1.5	actionResult	ManagedObjectClass and ManagedObjectInstance parameters shall be present	o.5		
1.6	commonError	ManagedObjectClass and ManagedObjectInstance parameters shall be present	o.5		
1.6.1	managedObjectClass		m		
1.6.2	managedObjectInstance		m		
1.6.3	errorId		m		
1.6.4	errorValue		m		

If support for the object creation MAPDU in the manager role is claimed (B.2/1a), then the supplier of the implementation shall state whether or not each parameter of the MAPDU is supported in Table C.4.

Table C.4 – OperationResultNotification MAPDU (Manager receiving)

Index	Parameter name	Constraints and values	Status	Support	Additional information
1	OperationResult		m		
1.1	getResult		m		
1.2	getListError		m		
1.3	setResult		m		
1.4	setListError		m		
1.5	actionResult		m		
1.6	commonError		m		
1.6.1	managedObjectClass		m		
1.6.2	managedObjectInstance		m		
1.6.3	errorId		m		
1.6.4	errorValue		m		

Annexe D

Formulaire MOCS⁵⁾

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

D.1 Introduction

The purpose of this MOCS proforma is to provide a mechanism for a supplier of an implementation of this Recommendation | International Standard which claims conformance to a managed object class to provide conformance information in a standard form.

D.1.1 Symbols, abbreviations and terms

The MOCS proforma contained in this annex is comprised of information in a tabular form in accordance with the guidelines presented in ISO/IEC 9646-2.

The following common notations, defined in ISO/IEC 9646-2 are used for the status column:

- c Conditional
- m Mandatory
- o Optional
- x Prohibited
- Not applicable

The following common notations, defined in ISO/IEC 9646-2 are used for the support column:

- Ig The item is ignored (i.e. processed systematically but not semantically)
- N Not implemented
- Y Implemented
- Not applicable

The following abbreviations are used throughout this Recommendation | International Standard:

- dmiAtt joint-iso-ccitt ms(9) smi(3) part2(2) attribute(7)
- dmiPkg joint-iso-ccitt ms(9) smi(3) part2(2) package(4)
- dmiNotif joint-iso-ccitt ms(9) smi(3) part2(2) notification(10)
- schedMo joint-iso-ccitt ms(9) function(2) part15(15) managedObjectClass(3)
- schedAtt joint-iso-ccitt ms(9) function(2) part15(15) attribute(7)
- schedNotif joint-iso-ccitt ms(9) function(2) part15(15) notification(10)
- schedPkg joint-iso-ccitt ms(9) function(2) part15(15) package(4)
- schedNb joint-iso-ccitt ms(9) function(2) part15(15) nameBinding(6)

D.1.2 Instructions for completing the MOCS proforma to produce a MOCS⁶⁾

The supplier of the implementation shall state which items are supported in the tables below and, if necessary provide additional information.

D.2 Daily Scheduler managed object class

D.2.1 Statement of conformance to the managed object class

See Table D.1.

⁵⁾ Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MOCS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire une fois celui-ci complété. Les instructions pour compléter le formulaire MOCS sont spécifiées en 5.2.2 de la Rec. UIT-T X.724 | ISO/CEI 10165-6.

⁶⁾ Les instructions pour le formulaire MOCS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 10165-6, article 5.

Table D.1

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	dailyScheduler	{schedMo 2}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.2.

Table D.2

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.2.2 Packages

See Table D.3.

Table D.3 – Daily scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c1		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	multipleDailyScheduling	{schedPkg 1}		m		
c1 If D.3/3 or D.3/6 then m else –.						

D.2.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.4). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.4 – Daily scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c2		x	
4	allomorphs	{dmiAtt 50}		–		c3		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrative State	{dmiAtt 31}		m		m		m	
7	operational State	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduled Managed Objects	{schedAtt 4}		x		c4		x	
11	sequenceOf Days	{schedAtt 6}		m		m		m	

c2 If D.3/2 then m else –.
c3 If D.3/3 then m else –.
c4 If D.3/6 then m else –.

Table D.4 (end) – Daily scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	m		m		o		

D.2.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below. The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes (see Table D.5).

Table D.5 – Daily scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{ dmiNotif 1 }		m			
2	StateChange	{ dmiNotif 14 }		m			
3	ObjectCreation	{ dmiNotif 6 }		m			
4	ObjectDeletion	{ dmiNotif 7 }		m			

Table D.5 (continued) – Daily scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{ dmiAtt 7 }		o		
	1.2	AttributeIdentifierList	{ dmiAtt 8 }		o		
	1.3	AttributeValueChangeDefinition	{ dmiAtt 10 }		m		
	1.4	NotificationIdentifier	{ dmiAtt 16 }		c20		
	1.5	CorrelatedNotifications	{ dmiAtt 12 }		o		
	1.6	AdditionalText	{ dmiAtt 7 }		o		
	1.7	AdditionalInformation	{ dmiAtt 6 }		–		
2	2.1	SourceIndicator	{ dmiAtt 7 }		o		
	2.2	AttributeIdentifierList	{ dmiAtt 8 }		o		
	2.3	StateChangeDefinition	{ dmiAtt 28 }		m		
	2.4	NotificationIdentifier	{ dmiAtt 16 }		c21		
	2.5	CorrelatedNotifications	{ dmiAtt 12 }		o		
	2.6	AdditionalText	{ dmiAtt 7 }		o		
	2.7	AdditionalInformation	{ dmiAtt 6 }		–		
3	3.1	SourceIndicator	{ dmiAtt 7 }		o		
	3.2	AttributeList	{ dmiAtt 9 }		o		
	3.3	NotificationIdentifier	{ dmiAtt 16 }		c22		
	3.4	CorrelatedNotifications	{ dmiAtt 12 }		o		
	3.5	AdditionalText	{ dmiAtt 7 }		o		
	3.6	AdditionalInformation	{ dmiAtt 6 }		–		

Table D.5 (end) – Daily scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c23		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	AdditionalInformation	{dmiAtt 6}		–		
c20 If D.5/1.5 then m else o. c21 If D.5/2.5 then m else o. c22 If D.5/3.4 then m else o. c23 If D.5/4.4 then m else o.							

D.3 Weekly scheduler managed object class

D.3.1 Statement of conformance to the managed object class

See Table D.6.

Table D.6

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	WeeklyScheduler	{schedMo 3}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.7.

Table D.7

Index	Managed object class template label	Value of object identifier for actual class	Additional information

D.3.2 Packages

See Table D.8.

Table D.8 – Weekly scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c5		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	multipleWeeklyScheduling	{schedPkg 3}		m		
c5 If D.8/3 or D.8/6 then m else –.						

D.3.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.9). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.9 – Weekly scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c6		x	
4	allomorphs	{dmiAtt 50}		–		c7		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduledManagedObjects	{schedAtt 4}		x		c8		x	
11	sequenceOfWeeks	{schedAtt 8}		m		m		m	
c6 If D.8/2 then m else –.									
c7 If D.8/3 then m else –.									
c8 If D.8/6 then m else –.									

Table D.9 (end) – Weekly scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	m		m		o		

D.3.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.10). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.10 – Weekly scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			

Table D.10 (end) – Weekly scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c24		
	1.5	CorrelatedNotifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c25		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c26		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c27		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	AdditionalInformation	{dmiAtt 6}		–		
c24 If D.10/1.5 then m else o. c25 If D.10/2.5 then m else o. c26 If D.10/3.4 then m else o. c27 If D.10/4.4 then m else o.							

D.4 Monthly scheduler managed object class

D.4.1 Statement of conformance to the managed object class

See Table D.11.

Table D.11

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	MonthlyScheduler	{ schedMo 4 }		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.12.

Table D.12

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.4.2 Packages

See Table D.13.

D.13 – Monthly scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{ dmiPkg 16 }		c9		
3	allomorphicPackage	{ dmiPkg 17 }		o		
4	schedulerObjectPackage	{ schedPkg 7 }		m		
5	duration	{ dmiPkg 26 }		m		
6	scheduledManagedObjectsPackage	{ schedPkg 6 }		o		
7	multipleMonthlyScheduling	{ schedPkg 2 }		m		
c5 If D.13/3 or D.13/6 then m else –.						

D.4.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.14). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.14 – Monthly scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c10		x	
4	allomorphs	{dmiAtt 50}		–		c11		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduledManagedObjects	{schedAtt 4}		x		c12		x	
11	sequenceOfMonths	{schedAtt 7}		m		m		m	
c10 If D.13/2 then m else –. c11 If D.13/3 then m else –. c12 If D.13/6 then m else –.									

Table D.14 (end) – Monthly scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	m		m		o		

D.4.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.15). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.15 – Monthly scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			

Table D.15 (continued) – Monthly scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c28		
	1.5	CorrelatedNotifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c29		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c30		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		

Table D.15 (end) – Monthly scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c31		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	AdditionalInformation	{dmiAtt 6}		–		
c28 If D.15/1.5 then m else o. c29 If D.15/2.5 then m else o. c30 If D.15/3.4 then m else o. c31 If D.15/4.4 then m else o.							

D.5 Periodic scheduler managed object class

D.5.1 Statement of conformance to the managed object class

See Table D.16.

Table D.16

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	Periodic scheduler	{schedMo 5}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.17.

Table D.17

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.5.2 Packages

See Table D.18.

Table D.18 – Periodic scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c13		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	periodicSchedulingPackage	{schedPkg 4}		m		
8	ResynchronizeModePackage	{schedPkg 5}		o		
9	periodSynchronizationPackage	{joint-iso-ccitt ms(9) function(2) part13(13) package(4) 10}		o		

c13 If D.18/3 or D.18/6 or D.18/8 or D.18/9 then m else –.

D.5.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.19). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.19 – Periodic scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c14		x	
4	allomorphs	{dmiAtt 50}		–		c15		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	

Table D.19 (continued) – Periodic scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
10	scheduledManagedObjects	{ schedAtt 4 }		x		c16		x	
11	timePeriod	{ schedAtt 9 }		m		m		m	
12	resynchronizeMode	{ schedAtt 3 }		m		c17		c17	
13	periodSynchronizationTime	{ joint-iso-ccitt ms(9) function(2) part13(13) attribute(7) 24 }		m		c19		c19	
c14 If D.18/2 then m else –. c15 If D.18/3 then m else –. c16 If D.18/6 then m else –. c17 If D.18/8 then m else –. c19 If D.18/9 then m else –.									

Table D.19 (end) – Periodic scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	–		–		o		
12	–		–		c18		
13	–		–		–		
c18 If D.18/8 then o else –.							

D.5.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.20). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.20 – Periodic scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			

Table D.20 (continued) – Periodic scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c32		
	1.5	CorrelatedNotifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c33		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c34		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		

Table D.20 (end) – Periodic scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
4	4.1	SourceIndicator	{ dmiAtt 7 }		o		
	4.2	AttributeList	{ dmiAtt 9 }		o		
	4.3	NotificationIdentifier	{ dmiAtt 16 }		c35		
	4.4	CorrelatedNotifications	{ dmiAtt 12 }		o		
	4.5	AdditionalText	{ dmiAtt 7 }		o		
	4.6	AdditionalInformation	{ dmiAtt 6 }		–		
c32 If D.20/1.5 then m else o. c33 If D.20/2.5 then m else o. c34 If D.20/3.4 then m else o. c35 If D.20/4.4 then m else o.							

D.6 Daily Operation Scheduler managed object class**D.6.1 Statement of conformance to the managed object class**

See Table D.21.

Table D.21

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	dailyOperation Scheduler	{ schedMo 6 }		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.22.

Table D.22

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.6.2 Packages

See Table D.23.

Table D.23 – Daily operation scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c36		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	operationsSchedulingPackage	{schedPkg 8}		m		
8	multipleDailyScheduling	{schedPkg 1}		m		
9	operationNotificationPackage	{schedPkg 9}		o		
c36 If D.23/3 or D.23/6 or D.23/9 then m else –.						

D.6.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.24). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.24 – Daily operation scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c37		x	
4	allomorphs	{dmiAtt 50}		–		c38		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduledManagedObjects	{schedAtt 4}		x		c39		x	
11	sequenceOfDays	{schedAtt 6}		m		m		m	
12	operationSpecifications	{schedAtt 10}		m		m		m	
c37 If D.23/2 then m else –.									
c38 If D.23/3 then m else –.									
c39 If D.23/6 then m else –.									

Table D.24 (end) – Daily operation scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	m		m		o		
12	m		m		o		

D.6.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.25). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.25 – Daily operation scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			
5	OperationResult	{schedNotif 1}		o			

Table D.25 (end) – Daily operation scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c40		
	1.5	CorrelatedNotifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c41		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c42		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c43		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	AdditionalInformation	{dmiAtt 6}		–		
5	5.1	OperationResult	{schedAtt 11}		m		
c40 If D.25/1.5 then m else o. c41 If D.25/2.5 then m else o. c42 If D.25/3.4 then m else o. c43 If D.25/4.4 then m else o.							

D.7 Weekly operation scheduler managed object class

D.7.1 Statement of conformance to the managed object class

See Table D.26.

Table D.26

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	WeeklyOperation Scheduler	{schedMo 7}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.27.

Table D.27

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.7.2 Packages

See Table D.28.

Table D.28 – Weekly operation scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c44		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	operationsSchedulingPackage	{schedPkg 8}		m		
8	multipleWeeklyScheduling	{schedPkg 3}		m		
9	operationNotificationPackage	{schedPkg 9}		o		
c44 If D.28/3 or D.28/6 or D.28/9 then m else –.						

D.7.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (See Table D.29). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.29 – Weekly operation scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c45		x	
4	allomorphs	{dmiAtt 50}		–		c46		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduledManagedObjects	{schedAtt 4}		x		c47		x	
11	sequenceOfWeeks	{schedAtt 8}		m		m		m	
12	operationSpecifications	{schedAtt 10}		m		m		m	
c45 If D.28/2 then m else –. c46 If D.28/3 then m else –. c47 If D.28/6 then m else –.									

Table D.29 (end) – Weekly operation scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	m		m		o		
12	m		m		o		

D.7.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.30). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.30 – Weekly operation scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			
5	OperationResult	{schedNotif 1}		o			

Table D.30 (continued) – Weekly operation scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c48		
	1.5	CorrelatedNotifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c49		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c50		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		

Table D.30 (end) – Weekly operation scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c51		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	AdditionalInformation	{dmiAtt 6}		–		
5	5.1	OperationResult	{schedAtt 11}		m		
c48 If D.30/1.5 then m else o. c49 If D.30/2.5 then m else o. c50 If D.30/3.4 then m else o. c51 If D.30/4.4 then m else o.							

D.8 Monthly operation scheduler managed object class

D.8.1 Statement of conformance to the managed object class

See Table D.31.

Table D.31

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	MonthlyOperation Scheduler	{schedMo 8}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.32.

Table D.32

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.8.2 Packages

See Table D.33.

Table D.33 – Monthly operation scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
y2	packagePackage	{dmiPkg 16}		c52		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	operationsSchedulingPackage	{schedPkg 8}		m		
8	multipleMonthlyScheduling	{schedPkg 2}		m		
9	operationNotificationPackage	{schedPkg 9}		o		
c52 If D.33/3 or D.33/6 or D.33/9 then m else –.						

D.8.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.34). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.34 – Monthly operation scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c53		x	
4	allomorphs	{dmiAtt 50}		–		c54		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduledManagedObjects	{schedAtt 4}		x		c55		x	
11	sequenceOfMonths	{schedAtt 7}		m		m		m	
12	operationSpecifications	{schedAtt 10}		m		m		m	
c53 If D.33/2 then m else –.									
c54 If D.33/3 then m else –.									
c55 If D.33/6 then m else –.									

Table D.34 (end) – Monthly operation scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		
9	–		–		o		
10	x		x		x		
11	m		m		o		
12	m		m		o		

D.8.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.35). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.35 – Weekly operation scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			
5	OperationResult	{schedNotif 1}		o			

Table D.35 (end) – Monthly operation scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c56		
	1.5	CorrelatedNotifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c57		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c58		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c59		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	AdditionalInformation	{dmiAtt 6}		–		
5	5.1	OperationResult	{schedAtt 11}		m		
c56 If D.35/1.5 then m else o. c57 If D.35/2.5 then m else o. c58 If D.35/3.4 then m else o. c59 If D.35/4.4 then m else o							

D.9 Periodic operation scheduler managed object class

D.9.1 Statement of conformance to the managed object class

See Table D.36.

Table D.36

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	PeriodicOperation Scheduler	{schedMo 9}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.37.

Table D.37

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.9.2 Packages

See Table D.38.

Table D.38 – Periodic operation scheduler packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c60		
3	allomorphicPackage	{dmiPkg 17}		o		
4	schedulerObjectPackage	{schedPkg 7}		m		
5	duration	{dmiPkg 26}		m		
6	scheduledManagedObjectsPackage	{schedPkg 6}		o		
7	periodicSchedulingPackage	{schedPkg 1}		m		
8	ResynchronizeModePackage	{schedPkg 5}		o		
9	periodSynchronizationPackage	{joint-iso-ccitt ms(9) function(2) part13(13) package(4) 10}		o		
10	operationsSchedulingPackage	{schedPkg 8}		m		
11	operationNotificationPackage	{schedPkg 9}		o		
c60 If D.38/3 or D.38/6 or D.38/8 or D.38/9 or D.38/11 then m else –.						

D.9.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.39). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.39 – Periodic operation scheduler attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c61		x	
4	allomorphs	{dmiAtt 50}		–		c62		x	
5	schedulerId	{schedAtt 5}		m		m		x	
6	administrativeState	{dmiAtt 31}		m		m		m	
7	operationalState	{dmiAtt 35}		–		m		x	
8	startTime	{dmiAtt 68}		m		m		m	
9	stopTime	{dmiAtt 69}		m		m		m	
10	scheduledManagedObjects	{schedAtt 4}		x		c63		x	
11	timePeriod	{schedAtt 4}		m		m		m	
12	resynchronizeMode	{schedAtt 5}		m		c64		c64	
13	periodSynchronizationTime	{joint-iso-ccitt ms(9) function(2) part13(13) attribute(7) 24}		m		c65		c65	
14	operationSpecifications	{schedAtt 10}		m		m		m	

c61 If D.38/2 then m else –.
c62 If D.38/3 then m else –.
c63 If D.38/6 then m else –.
c64 If D.38/8 then m else –.
c65 If D.38/9 then m else –.

Table D.39 (continued) – Periodic operation scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		x		
4	x		x		x		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		o		

Table D.39 (end) – Periodic operation scheduler attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
9	–		–		o		
10	x		x		x		
11	–		–		o		
12	–		–		c66		
13	–		–		–		
14	m		m		o		

c66 If D.38/8 then o else –.

D.9.4 Notifications

The supplier of the implementation shall state whether or not notifications specified by all packages instantiated in a managed object of this class are supported, in the “Support” and “Additional information” columns below (see Table D.40). The supplier of the implementation shall indicate support in terms of the confirmed and non-confirmed modes.

Table D.40 – Periodic operation scheduler notification support

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information
					Cnf	Non-Cnf	
1	AttributeValueChange	{dmiNotif 1}		m			
2	StateChange	{dmiNotif 14}		m			
3	ObjectCreation	{dmiNotif 6}		m			
4	ObjectDeletion	{dmiNotif 7}		m			
5	OperationResult	{schedNotif 1}		o			

Table D.40 (end) – Periodic operation scheduler notification support

Index	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	1.1	SourceIndicator	{dmiAtt 7}		o		
	1.2	AttributeIdentifierList	{dmiAtt 8}		o		
	1.3	AttributeValueChangeDefinition	{dmiAtt 10}		m		
	1.4	NotificationIdentifier	{dmiAtt 16}		c67		
	1.5	Correlated Notifications	{dmiAtt 12}		o		
	1.6	AdditionalText	{dmiAtt 7}		o		
	1.7	AdditionalInformation	{dmiAtt 6}		–		
2	2.1	SourceIndicator	{dmiAtt 7}		o		
	2.2	AttributeIdentifierList	{dmiAtt 8}		o		
	2.3	StateChangeDefinition	{dmiAtt 28}		m		
	2.4	NotificationIdentifier	{dmiAtt 16}		c68		
	2.5	CorrelatedNotifications	{dmiAtt 12}		o		
	2.6	AdditionalText	{dmiAtt 7}		o		
	2.7	AdditionalInformation	{dmiAtt 6}		–		
3	3.1	SourceIndicator	{dmiAtt 7}		o		
	3.2	AttributeList	{dmiAtt 9}		o		
	3.3	NotificationIdentifier	{dmiAtt 16}		c69		
	3.4	CorrelatedNotifications	{dmiAtt 12}		o		
	3.5	AdditionalText	{dmiAtt 7}		o		
	3.6	AdditionalInformation	{dmiAtt 6}		–		
4	4.1	SourceIndicator	{dmiAtt 7}		o		
	4.2	AttributeList	{dmiAtt 9}		o		
	4.3	NotificationIdentifier	{dmiAtt 16}		c70		
	4.4	CorrelatedNotifications	{dmiAtt 12}		o		
	4.5	AdditionalText	{dmiAtt 7}		o		
	4.6	Additional Information	{dmiAtt 6}		–		
5	5.1	OperationResult	{schedAtt 11}		m		
c67 If D.40/1.5 then m else o. c68 If D.40/2.5 then m else o. c69 If D.40/3.4 then m else o. c70 If D.40/4.4 then m else o.							

D.10 Operation result record managed object class**D.10.1 Statement of conformance to the managed object class**

See Table D.41.

Table D.41

Index	Managed object class template label	Value of object identifier for the class	Support of all mandatory features	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
	OperationResultRecord	{schedMo 10}		

If the answer to the actual class question in the managed object class support table is no, the supplier of the implementation shall fill in the actual class support Table D.42.

Table D.42

Index	Actual managed object class template label	Value of object identifier for actual class	Additional information

D.10.2 Packages

See Table D.43

Table D.43 – Operation result record packages

Index	Package Name	Value of object identifier	Constraints and values	Status	Support	Additional information
1	topPackage	–		m		
2	packagePackage	{dmiPkg 16}		c71		
3	allomorphicPackage	{dmiPkg 17}		o		
4	logRecordPackage	–		m		
5	eventLogRecordPackage	–		m		
6	eventTimePackage	{dmiPkg 11}		o		
7	notificationIdentifierPackage	{dmiPkg 24}		o		
8	correlatedNotificationsPackage	{dmiPkg 23}		o		
9	additionalTextPackage	{dmiPkg 19}		o		
10	additionalInformationPackage	{dmiPkg 18}		o		
11	operationResultRecordPackage	{schedPkg 10}		m		

c71 If D.43/3 or D.43/6 or D.43/7 or D.43/8 or D.43/9 or D.43/10 then m else –.

D.10.3 Attributes

The supplier of the implementation shall state whether or not the attributes specified by all the packages instantiated in a managed object class definition are supported, in the “Support” and “Additional Information” columns below (see Table D.44). The supplier of the implementation shall indicate support for each of the operations supported.

Table D.44 – Operation result record attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	ObjectClass	{dmiAtt 65}		–		m		x	
2	nameBinding	{dmiAtt 63}		o		m		x	
3	packages	{dmiAtt 66}		o		c72		x	
4	allomorphs	{dmiAtt 50}		–		c73		x	
5	logRecordId	{dmiAtt 3}		x		m		x	
6	loggingTime	{dmiAtt 59}		x		m		m	
7	managedObjectClass	{dmiAtt 60}		x		m		x	
8	managedObjectInstance	{dmiAtt 61}		x		m		x	
9	eventType	{dmiAtt 14}		x		m		x	
10	eventTime	{dmiAtt 13}		x		c74		x	
11	notificationIdentifier	{dmiAtt 16}		x		c75		x	
12	correlatedNotifications	{dmiAtt 12}		x		c76		x	
13	additionalText	{dmiAtt 7}		x		c77		x	
14	additionalInformation	{dmiAtt 6}		x		c78		x	
15	operationResult	{schedAtt 11}		x		m		x	
<p>c72 If D.43/2 then m else –.</p> <p>c73 If D.43/3 then m else –.</p> <p>c74 If D.43/6 then m else –.</p> <p>c75 If D.43/7 then m else –.</p> <p>c76 If D.43/8 then m else –.</p> <p>c77 If D.43/9 then m else –.</p> <p>c78 If D.43/10 then m else –.</p>									

Table D.44 (end) – Operation result record attribute support

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		–		
2	–		–		–		
3	x		x		–		
4	x		x		–		
5	–		–		–		
6	–		–		–		
7	–		–		–		
8	–		–		–		
9	–		–		–		
10	–		–		–		
11	–		–		–		
12	x		x		–		
13	–		–		–		
14	x		x		–		
145	x		x		–		

Annexe E

Formulaire MIDS⁷⁾

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

Table E.1 – Attribute support for Scheduled managed objects

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	SetByCreate		Get		Replace	
				Status	Support	Status	Support	Status	Support
1	externalSchedulerName	{schedAtt 1}		o		m		m	
2	onDuty	{schedAtt 2}		x		m		x	

Table E.1 (end) – Attribute support for Scheduled managed objects

Index	Add		Remove		SetToDefault		Additional information
	Status	Support	Status	Support	Status	Support	
1	–		–		x		
2	–		–		x		

⁷⁾ Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MIDS de la présente annexe pour utiliser celui-ci conformément à son objet. Les instructions pour compléter le formulaire MIDS sont spécifiées en 5.2.2 de la Rec. UIT-T X.724 | ISO/CEI 10165-6.

Annexe F

Formulaire MRCS⁸⁾

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

F.1 Name bindings

The purpose of this MRCS for name bindings is to provide a mechanism for a supplier which claims conformance to a name binding to provide conformance information in a standard form.

F.2 Instructions for completing the MRCS proforma for name bindings to produce a MRCS⁹⁾

The supplier of the implementation shall state which name bindings in which instances of the class can be subordinate are supported, in the “Support” and “Additional Information” columns below.

F.3 Statement of conformance to the name binding

See Table F.1.

Table F.1 – Name binding support

Index	Name binding template label	Value of object identifier for name binding	Constraints and values	Status	Support	Additional information
1	scheduler-system	{ schedNb 1 }		m		

Table F.1 (end) – Name binding support

Index	Subindex	Operation	Constraints and values	Status	Support	Additional information
1	1.1	Create support		m		
	1.2	Create with reference object		m		
	1.3	Create with automatic instance naming		m		
	1.4	Delete support		m		
	1.5	Delete only if no contained objects		m		
	1.6	Delete contained objects		x		

⁸⁾ Les utilisateurs de la présente Recommandation | Norme internationale sont autorisés à reproduire le formulaire MRCS de la présente annexe pour utiliser celui-ci conformément à son objet. Ils sont également autorisés à publier le formulaire MRCS une fois celui-ci complété. Les instructions pour compléter le formulaire MRCS sont spécifiées en 5.2.2 de la Rec. UIT-T X.724 | ISO/CEI 10165-6.

⁹⁾ Les instructions pour le formulaire MRCS sont spécifiées dans la Rec. UIT-T X.724 | ISO/CEI 10165-6, article 5.