



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

X.744

(10/96)

SERIE X: REDES DE DATOS Y COMUNICACIÓN
ENTRE SISTEMAS ABIERTOS

Gestión de interconexión de sistemas abiertos

**Tecnología de la información – Interconexión de
sistemas abiertos – Gestión de sistemas:
Función de gestión de soporte lógico**

Recomendación UIT-T X.744

(Anteriormente Recomendación del CCITT)

RECOMENDACIONES DE LA SERIE X DEL UIT-T

REDES DE DATOS Y COMUNICACIÓN ENTRE SISTEMAS ABIERTOS

REDES PÚBLICAS DE DATOS	X.1–X.199
Servicios y facilidades	X.1–X.19
Interfaces	X.20–X.49
Transmisión, señalización y conmutación	X.50–X.89
Aspectos de redes	X.90–X.149
Mantenimiento	X.150–X.179
Disposiciones administrativas	X.180–X.199
INTERCONEXIÓN DE SISTEMAS ABIERTOS	X.200–X.299
Modelo y notación	X.200–X.209
Definiciones de los servicios	X.210–X.219
Especificaciones de los protocolos en modo conexión	X.220–X.229
Especificaciones de los protocolos en modo sin conexión	X.230–X.239
Formularios para declaraciones de conformidad de implementación de protocolo	X.240–X.259
Identificación de protocolos	X.260–X.269
Protocolos de seguridad	X.270–X.279
Objetos gestionados de capa	X.280–X.289
Pruebas de conformidad	X.290–X.299
INTERFUNCIONAMIENTO ENTRE REDES	X.300–X.399
Generalidades	X.300–X.349
Sistemas de transmisión de datos por satélite	X.350–X.399
SISTEMAS DE TRATAMIENTO DE MENSAJES	X.400–X.499
DIRECTORIO	X.500–X.599
GESTIÓN DE REDES DE INTERCONEXIÓN DE SISTEMAS ABIERTOS Y ASPECTOS DE SISTEMAS	X.600–X.699
Gestión de redes	X.600–X.629
Eficacia	X.630–X.649
Denominación, direccionamiento y registro	X.650–X.679
Notación de sintaxis abstracta uno	X.680–X.699
GESTIÓN DE INTERCONEXIÓN DE SISTEMAS ABIERTOS	X.700–X.799
Marco y arquitectura de la gestión de sistemas	X.700–X.709
Servicio y protocolo de comunicación de gestión	X.710–X.719
Estructura de la información de gestión	X.720–X.729
Funciones de gestión	X.730–X.799
SEGURIDAD	X.800–X.849
APLICACIONES DE INTERCONEXIÓN DE SISTEMAS ABIERTOS	X.850–X.899
Cometimiento, concurrencia y recuperación	X.850–X.859
Tratamiento de transacciones	X.860–X.879
Operaciones a distancia	X.880–X.899
PROCESAMIENTO DISTRIBUIDO ABIERTO	X.900–X.999

Para más información, véase la Lista de Recomendaciones del UIT-T.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la UIT. En el UIT-T, que es la entidad que establece normas mundiales (Recomendaciones) sobre las telecomunicaciones, participan unos 179 países miembros, 84 empresas de explotación de telecomunicaciones, 145 organizaciones científicas e industriales y 38 organizaciones internacionales.

Las Recomendaciones las aprueban los Miembros del UIT-T de acuerdo con el procedimiento establecido en la Resolución N.º 1 de la CMNT (Helsinki, 1993). Adicionalmente, la Conferencia Mundial de Normalización de las Telecomunicaciones (CMNT), que se celebra cada cuatro años, aprueba las Recomendaciones que para ello se le sometan y establece el programa de estudios para el periodo siguiente.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI. El texto de la Recomendación UIT-T X.744 se aprobó por la CMNT (Ginebra 9-18 de octubre de 1996). Su texto se publica también, en forma idéntica, como Norma Internacional ISO/CEI 10164-18.

NOTA

En esta Recomendación, la expresión «Administración» se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

© UIT 1997

Es propiedad. Ninguna parte de esta publicación puede reproducirse o utilizarse, de ninguna forma o por ningún medio, sea éste electrónico o mecánico, de fotocopia o de microfilm, sin previa autorización escrita por parte de la UIT, excepto en los casos indicados en las notas de pie de página 3) a 7) de los Anexos B a F respectivamente.

ÍNDICE

	<i>Página</i>
1 Alcance.....	1
2 Referencias normativas	2
2.1 Recomendaciones Normas Internacionales idénticas.....	2
2.2 Pares de Recomendaciones Normas Internacionales de contenido técnico equivalente	3
2.3 Referencias adicionales.....	3
3 Definiciones	3
3.1 Definiciones de la estructura de gestión.....	4
3.2 Definiciones de la descripción general de gestión de sistemas	4
3.3 Definiciones del servicio común de información de gestión	4
3.4 Definiciones del modelo de información de gestión	4
3.5 Directrices para la definición de definiciones de objetos gestionados	4
3.6 Definiciones del formulario de declaración de conformidad de implementación	4
3.7 Definiciones adicionales	4
4 Símbolos y abreviaturas	5
5 Convenios.....	6
6 Requisitos para la gestión de soporte lógico	6
7 Modelo para la función de gestión de soporte lógico	6
7.1 Funciones de gestión de soporte lógico	7
7.1.1 Creación	7
7.1.2 Supresión	7
7.1.3 Entrega	7
7.1.4 Ejecución de programa	7
7.1.5 Obtención.....	7
7.1.6 Instalación	7
7.1.7 Reversión	8
7.1.8 Fijación del estado administrativo	8
7.1.9 Terminación de validación.....	8
7.1.10 Validación	8
7.1.11 Notificaciones	8
7.2 Otras funciones	8
7.2.1 Copia de seguridad.....	9
7.2.2 Restablecimiento.....	9
7.3 Objeto gestionado unidad de soporte lógico	9
7.3.1 Ciclo vital.....	9
7.3.2 Estados del objeto gestionado unidad de soporte lógico.....	10
7.3.3 Operaciones en el objeto gestionado unidad de soporte lógico	11
7.4 Objeto gestionado soporte lógico ejecutable	11
7.4.1 Estados adicionales para el objeto gestionado soporte lógico ejecutable	11
7.4.2 Operaciones adicionales para el objeto gestionado soporte lógico ejecutable	12
7.5 Objeto gestionado distribución de soporte lógico	12
7.5.1 Operaciones en el objeto gestionado distribución de soporte lógico	12
7.5.2 Estados del objeto gestionado distribución de soporte lógico	13
7.6 Relaciones	13
8 Definiciones genéricas	13
8.1 Objetos gestionados	13
8.1.1 Soporte lógico	13
8.1.2 Unidad de soporte lógico	14
8.1.3 Soporte lógico ejecutable	15
8.1.4 Distribución de soporte lógico	15

8.2	Atributos	15
8.2.1	Parches aplicados (appliedPatches)	15
8.2.2	Suma de control (checksum)	16
8.2.3	Fecha de entregado (dataDelivered)	16
8.2.4	Fecha de instalado (dateInstalled).....	16
8.2.5	Fecha de creación (dateOfCreation)	16
8.2.6	Fecha de la última modificación (dateOfLastModification)	16
8.2.7	Ubicación de fichero (fileLocation).....	16
8.2.8	Tamaño de fichero (fileSize).....	16
8.2.9	Tipo de fichero (fileType).....	16
8.2.10	Destino futuro de copia de seguridad automática (futureAutoBackupDestination).....	16
8.2.11	Umbral futuro de activación de copia de seguridad automática (futureAutoBackupTriggerThreshold)	17
8.2.12	Permitido restablecimiento automático futuro (futureAutoRestoreAllowed)	17
8.2.13	Fuente de restablecimiento automático futuro (futureAutoRestoreSource).....	17
8.2.14	Identidad de creador (identityOfCreator).....	17
8.2.15	Identidad de último modificador (identityOfLastModifier).....	17
8.2.16	Destino de última copia de seguridad (lastBackupDestination).....	17
8.2.17	Tiempo de última copia de seguridad (lastBackupTime).....	17
8.2.18	Fuente de último restablecimiento (lastRestoreSource).....	17
8.2.19	Tiempo de último restablecimiento (lastRestoreTime).....	17
8.2.20	Campo de nota (noteField).....	17
8.2.21	Id de distribución de soporte lógico (softwareDistributorId).....	17
8.3	Acciones.....	18
8.3.1	Copia de seguridad.....	18
8.3.2	Entrega	18
8.3.3	Ejecución de programa	18
8.3.4	Instalación	18
8.3.5	Restablecimiento	18
8.3.6	Reversión	19
8.3.7	Terminación de validación.....	19
8.3.8	Validación	19
8.4	Notificaciones	19
8.4.1	Informe de copia de seguridad automático (autoBackupReport).....	20
8.4.2	Informe de restablecimiento automático (autoRestoreReport).....	20
8.4.3	Notificación de resultado de entrega (deliverResultNotification).....	20
8.5	Parámetros.....	20
8.5.1	Parámetro de efecto de alarma en servicio (alarmEffectOnServiceParameter)	20
8.5.2	Parámetro fallo de procesamiento de soporte lógico (softwareProcessingFailureParameter)	20
8.6	Vinculaciones de nombre.....	21
8.6.1	Distribución de soporte lógico – Subsistema	21
8.6.2	Distribución de soporte lógico – Sistema.....	21
8.6.3	Unidad de soporte lógico – Subsistema	21
8.6.4	Unidad de soporte lógico – Sistema.....	21
9	Definiciones de servicios	21
9.1	Introducción	21
9.2	Servicio copia de seguridad (backup)	21
9.3	Servicio de entrega (deliver)	21
9.4	Servicio ejecución de programa (execute program).....	21
9.5	Servicio instalación (install).....	23
9.6	Servicio restablecimiento (restore)	24
9.7	Servicio reversión (revert)	24
9.8	Servicio terminación de validación (terminate validation)	24
9.9	Servicio validación (validate)	26
9.10	Servicio informe de copia de seguridad automática (Auto backup report).....	26
9.11	Servicio informe de restablecimiento automático (auto restore report)	26
9.12	Servicio notificación de resultado de entrega (Deliver result notification).....	27

	<i>Página</i>
10 Unidades funcionales	28
11 Protocolo	28
11.1 Elementos de procedimiento	28
11.1.1 Procedimiento de copia de seguridad.....	28
11.1.2 Procedimiento de entrega.....	29
11.1.3 Procedimiento de ejecución de programa	29
11.1.4 Procedimiento de instalación	30
11.1.5 Procedimiento de restablecimiento	31
11.1.6 Procedimiento de reversión.....	31
11.1.7 Procedimiento de terminación de validación	32
11.1.8 Procedimiento de validación.....	32
11.1.9 Procedimiento de informe de copia de seguridad automática.....	33
11.1.10 Procedimiento de informe de restablecimiento automático	34
11.1.11 Procedimiento de notificación de resultado de entrega.....	34
11.2 Sintaxis abstracta	35
11.2.1 Objetos.....	35
11.2.2 Lotes.....	35
11.2.3 Atributos	36
11.2.4 Notificaciones	36
11.2.5 Acciones.....	36
11.2.6 Vinculaciones de nombre.....	37
11.3 Negociación de unidades funcionales	37
12 Relación con otras funciones.....	37
13 Conformidad	38
13.1 Conformidad estática	38
13.2 Conformidad dinámica.....	38
13.3 Requisitos de la declaración de conformidad de implementación de gestión	38
Anexo A – Definición de información de gestión	39
A.1 Managed Object Classes	39
A.1.1 Software	39
A.1.2 softwareUnit.....	39
A.1.3 executableSoftware	40
A.1.4 softwareDistributor	40
A.2 Packages.....	40
A.2.1 appliedPatchPackage.....	40
A.2.2 checksumPackage.....	41
A.2.3 executeProgramPackage	41
A.2.4 fileInformationPackage.....	41
A.2.5 filePackage.....	41
A.2.6 informationAutoBackupPackage	41
A.2.7 informationAutoRestorePackage	41
A.2.8 informationBackupPackage	42
A.2.9 informationRestorePackage	42
A.2.10 installPackage	42
A.2.11 noteFieldPackage	42
A.2.12 processingErrorAlarmOnServicePackage.....	42
A.2.13 revertPackage.....	42
A.2.14 terminateValidationPackage	42
A.2.15 usageStatePackage	43
A.2.16 validationPackage	43

	<i>Página</i>
A.3 Attributes.....	43
A.3.1 appliedPatches.....	43
A.3.2 checkSum.....	43
A.3.3 dateDelivered.....	43
A.3.4 dateInstalled.....	44
A.3.5 dateOfCreation.....	44
A.3.6 dateOfLastModification.....	44
A.3.7 fileLocation.....	44
A.3.8 fileSize.....	45
A.3.9 fileType.....	45
A.3.10 futureAutoBackupDestination.....	45
A.3.11 futureAutoBackupTriggerThreshold.....	45
A.3.12 futureAutoRestoreAllowed.....	46
A.3.13 futureAutoRestoreSource.....	46
A.3.14 identityOfCreator.....	46
A.3.15 identityOfLastModifier.....	46
A.3.16 lastBackupDestination.....	47
A.3.17 lastBackupTime.....	47
A.3.18 lastRestoreSource.....	47
A.3.19 lastRestoreTime.....	47
A.3.20 noteField.....	48
A.3.21 softwareDistributorId.....	48
A.4 Name Bindings.....	48
A.4.1 softwareDistributor-subsystem.....	48
A.4.2 softwareDistributor-system.....	48
A.4.3 softwareUnit-subsystem.....	49
A.4.4 softwareUnit-system.....	49
A.5 Actions.....	49
A.5.1 deliver.....	49
A.5.2 executeProgram.....	49
A.5.3 install.....	50
A.5.4 revert.....	50
A.5.5 terminateValidation.....	51
A.5.6 validate.....	51
A.6 Notifications.....	51
A.6.1 autoBackupReport.....	51
A.6.2 autoRestoreReport.....	52
A.6.3 deliverResultNotification.....	52
A.7 Parameters.....	52
A.7.1 alarmEffectOnServiceParameter.....	52
A.7.2 softwareProcessingFailureParameter.....	53
A.8 Supporting Productions.....	53
A.9 Backup and Restore Actions.....	56
A.9.1 backup.....	56
A.9.2 restore.....	56
A.9.3 Backup Restore Supporting Productions.....	57
Anexo B – Formulario de MCS.....	58
B.1 Introduction.....	58
B.1.1 Purpose and structure.....	58
B.1.2 Instructions for completing the MCS proforma to produce an MCS.....	58
B.1.3 Symbols, abbreviations and terms.....	58
B.1.4 Table format.....	58
B.2 Identification of the implementation.....	59
B.2.1 Date of statement.....	59
B.2.2 Identification of the implementation.....	60
B.2.3 Contact.....	60

B.3	Identification of the Recommendations International Standards in which the management information is defined	60
B.3.1	Technical corrigenda implemented	60
B.3.2	Amendments implemented.....	60
B.4	Management conformance summary	60
Anexo C	– Formulario de MICS	64
C.1	Introduction.....	64
C.2	Instructions for completing the MICS proforma to produce an MICS	64
C.3	Symbols, abbreviations and terms.....	64
C.4	Statement of conformance to the management information.....	64
C.4.1	Attributes.....	64
C.4.2	Actions	66
C.4.3	Notifications.....	71
C.4.4	Create and delete management operations	81
Anexo D	– Formulario de MOCS	82
D.1	Introduction.....	82
D.2	Instructions for completing the MOCS proforma to produce a MOCS	82
D.3	Symbols, abbreviations and terms.....	82
D.4	Software unit managed object class	82
D.4.1	Statement of conformance to the managed object class.....	82
D.4.2	Packages.....	83
D.4.3	Attributes.....	84
D.4.4	Attribute groups	86
D.4.5	Actions	86
D.4.6	Notifications.....	90
D.4.7	Parameters.....	98
D.5	Executable software managed object class	98
D.5.1	Statement of conformance to the managed object class.....	98
D.5.2	Packages.....	99
D.5.3	Attributes.....	100
D.5.4	Attribute groups	102
D.5.5	Actions	103
D.5.6	Notifications.....	108
D.5.7	Parameters.....	117
D.6	Software distributor managed object class.....	117
D.6.1	Statement of conformance to the managed object class.....	117
D.6.2	Packages.....	118
D.6.3	Attributes.....	118
D.6.4	Attribute groups	118
D.6.5	Actions	119
D.6.6	Notifications.....	120
D.6.7	Parameters.....	122
Anexo E	– Formulario de MRCS para vinculación de nombre	123
E.1	Introduction.....	123
E.2	Instructions for completing the MRCS proforma for name binding to produce an MRCS	123
E.3	Symbols, abbreviations and terms.....	123
E.4	Statement of conformance to the name binding.....	123
Anexo F	– Formulario MIDS	125
F.1	Actions	125

	<i>Página</i>
Anexo G – Clase de objeto soporte lógico de la Recomendación M.3100	126
G.1 Software Object Class	126
G.2 Packages.....	126
G.2.1 Administrative Operational States Package	126
G.2.2 Affected Object List Package	127
G.2.3 Attribute Value Change Notification Package	127
G.2.4 Create Delete Notifications Package.....	127
G.2.5 Current Problem List Package	127
G.2.6 Software Processing Error Alarm Package	127
G.2.7 State Change Notification Package	127
G.2.8 User Label Package.....	127
G.2.9 Vendor Name Package.....	127
G.2.10 Version Package.....	127
G.3 Attributes.....	128
G.3.1 Affected Object List.....	128
G.3.2 Alarm Status.....	128
G.3.3 Current Problem List.....	128
G.3.4 Software Id.....	129
G.3.5 User Label.....	130
G.3.6 Vendor Name	130
G.3.7 Version.....	130
Anexo H – Correspondencia entre el modelo POSIX y el modelo de gestión de soporte lógico	131

Resumen

Los sistemas de operaciones (OS, *operation system*), elementos de red (NE, *network element*) y otros sistemas están sujetos ocasionalmente a cambios de soporte lógico. Estos cambios pueden incluir códigos de programa, tales como una nueva versión de algún programa relacionado con el procesamiento de llamada o cuadros de encaminamiento revisados. La necesidad de una red es poder gestionar el soporte lógico que puede ser cambiado o modificado por algunos sistemas de operaciones distantes. En este contexto la gestión puede incluir: control sobre el proceso de telecarga para transferir un producto de soporte lógico desde un sistema de operaciones a otro sistema tales como un OS o NE; control de instalación de soporte lógico telecargado en la serie de programas existente en el sistema; y la capacidad para verificar la versión de soporte lógico instalado en un sistema. Esta Recomendación | Norma Internacional proporciona un medio basado en el mensaje X.700 estándar para gestionar este proceso de soporte lógico con el objeto de satisfacer estas necesidades. El método de transferencia de soportes lógicos entre un sistema y otro no se especifica ni se ve constreñido en modo alguno por esta Recomendación | Norma Internacional pero, prácticamente, podría incluir transferencia de ficheros, correo electrónico o servicios postales con discos flexibles. El cometido de esta Recomendación | Norma Internacional en esta transferencia es gestionar el comienzo de la entrega a otro sistema manteniendo información sobre el comienzo de una entrega de soporte lógico y relacionando esto con la recepción real, con lo cual el sistema de recepción produce una notificación de entrega exitosa.

NORMA INTERNACIONAL

RECOMENDACIÓN UIT-T

**TECNOLOGÍA DE LA INFORMACIÓN – INTERCONEXIÓN
DE SISTEMAS ABIERTOS – GESTIÓN DE SISTEMAS: FUNCIÓN
DE GESTIÓN DE SOPORTE LÓGICO****1 Alcance**

La función de gestión de soporte lógico comprende la gestión de un sistema para entrega de soporte lógico y también la gestión del soporte lógico dentro de un sistema.

Hay dos aspectos del soporte lógico que deben considerarse separadamente. Aspectos que se podrían describir como la visión «latente» y la visión «activa» del soporte lógico.

La visión latente del soporte lógico está relacionada con los datos que se almacenan en un sistema gestionado y la manera en que son entregados e instalados. Por lo general, los datos son información almacenada, tales como ficheros y cuadros de datos, pero también pueden ser ficheros que contengan un código ejecutable. Esta Recomendación | Norma Internacional abarca la visión latente del soporte lógico.

La visión activa del soporte lógico está relacionada con la gestión de recursos que utilizan el soporte lógico. No existe una diferencia real entre esta visión y la visión normal de la gestión de recursos. El alcance de esta Recomendación | Norma Internacional no incluye la visión activa del soporte lógico. Sin embargo, la relación entre objetos gestionados que representan recursos que utilizan el soporte lógico y los objetos gestionados que representan el soporte lógico que están utilizando (es decir, la visión latente del soporte lógico) sí queda dentro del alcance de la presente Recomendación | Norma Internacional.

El alcance de esta Recomendación | Norma Internacional incluye:

- la iniciación de la transferencia del soporte lógico;
- el control del soporte lógico posterior a la transferencia;
- la activación del soporte lógico (incluida la actualización y la corrección de la versión);
- la desactivación del soporte lógico;
- el cambio reconstitutivo del soporte lógico;
- la validación del soporte lógico;
- la indagación del soporte lógico;
- la copia de seguridad del soporte lógico;
- el restablecimiento del soporte lógico.

NOTA 1 – Copia de seguridad y restablecimiento se definen en modo genérico de modo que puedan aplicarse a recursos diferentes del soporte lógico.

El alcance de esta Recomendación | Norma Internacional no incluye:

- el mecanismo de transferencia del soporte lógico;
- el almacenamiento físico del soporte lógico (el vertido del soporte lógico a un almacenamiento físico de ficheros, por ejemplo un disco flexible, un disco duro, etc.);
- el formateo del soporte lógico;
- la denominación de los productos del soporte lógico;
- la secuenciación de las instrucciones de gestión del soporte lógico;
- la supervisión del soporte lógico;
- la gestión de la ejecución de procesos en un sistema.

NOTA 2 – El soporte de las actividades relacionadas con el cobro por licencia (es decir, la facturación, la contabilidad, etc.) y el seguimiento del soporte lógico (es decir, el seguimiento de la ejecución del soporte lógico para depuración) queda fuera del alcance de la presente Recomendación | Norma Internacional y está abierto a un ulterior estudio.

2 Referencias normativas

Las siguientes Recomendaciones | Normas Internacionales contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación | Norma Internacional. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones y Normas Internacionales son objeto de revisiones, por lo que se preconiza que los participantes en acuerdos basados en la presente Recomendación | Norma Internacional investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones y Normas Internacionales citadas a continuación. Los miembros de la CEI y de la ISO mantienen registros de las Normas Internacionales actualmente vigentes. La Oficina de Normalización de las Telecomunicaciones de la UIT mantiene una lista de las Recomendaciones UIT-T actualmente vigentes.

2.1 Recomendaciones | Normas Internacionales idénticas

- Recomendación UIT-T X.210 (1993) | ISO/CEI 10731:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Modelo de referencia básico – Convenios para la definición de servicios en la interconexión de sistemas abiertos.*
- Recomendación X.701 del CCITT (1992) | ISO/CEI 10040:1992, *Tecnología de la información – Interconexión de sistemas abiertos – Visión general de la gestión de sistemas*¹⁾.
- Recomendación X.720 del CCITT (1992) | ISO/CEI 10165-1:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Modelo de información de gestión.*
- Recomendación X.721 del CCITT (1992) | ISO/CEI 10165-2:1992, *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Definición de la información de gestión.*
- Recomendación X.722 del CCITT (1992) | ISO/CEI 10165-4:1992, *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Directrices para la definición de objetos gestionados.*
- Recomendación UIT-T X.723 (1993) | ISO/CEI 10165-5:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Información de gestión genérica.*
- Recomendación UIT-T X.724 (1993) | Norma ISO/CEI 10165-6:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la información de gestión: Requisitos y directrices para los formularios de declaración de conformidad de implementación asociados con la gestión de interconexión de sistemas abiertos.*
- Recomendación X.730 del CCITT (1992) | ISO/CEI 10164-1:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de gestión de objetos.*
- Recomendación X.731 del CCITT (1992) | ISO/CEI 10164-2:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de gestión de estados.*
- Recomendación X.732 del CCITT (1992) | ISO/CEI 10164-3:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Atributos para la representación de relaciones.*
- Recomendación UIT-T X.738 (1993) | ISO/CEI 10164-13:1995, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de sumario.*
- Recomendación UIT-T X.739 (1993) | ISO/CEI 10164-11:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Objetos métricos y atributos.*
- Recomendación X.740 del CCITT (1992) | ISO/CEI 10164-8:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de pista de auditoría de seguridad..*
- Recomendación UIT-T X.741 (1995) | ISO/CEI 10164-9:1995, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Objetos y atributos para el control de acceso.*
- Recomendación UIT-T X.742 (1995) | ISO/CEI 10164-10:1995, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de cómputo de utilización para contabilidad.*

¹⁾ Modificada por la Rec. UIT-T X.701/Corr.2 (1995) | ISO/CEI 10040/Corr.2:1995.

- Recomendación UIT-T X.745 (1993) | ISO/CEI 10164-12:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de gestión de prueba.*
- Recomendación UIT-T X.746 (1995) | ISO/CEI 10164-15:1995, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de planificación.*

2.2 Pares de Recomendaciones | Normas Internacionales de contenido técnico equivalente

- Recomendación X.208 del CCITT (1988), *Especificación de la notación de sintaxis abstracta uno.*
ISO/CEI 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1).*
- Recomendación X.209 del CCITT (1988), *Especificación de las reglas básicas de codificación de la notación de sintaxis abstracta uno.*
ISO/CEI 8825:1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*
- Recomendación UIT-T X.291 (1995), *Metodología y marco de las pruebas de conformidad de interconexión de sistemas abiertos de las Recomendaciones sobre los protocolos para aplicaciones del UIT-T – Especificación de sucesiones de pruebas abstractas.*
ISO/CEI 9646-2:1994, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 2: Abstract Test Suite specification.*
- Recomendación UIT-T X.296 (1995), *Metodología y marco de las pruebas de conformidad de interconexión de sistemas abiertos de las Recomendaciones sobre los protocolos para aplicaciones del UIT-T – Declaraciones de conformidad de implementación.*
ISO/CEI 9646-7:1995, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 7: Implementation Conformance Statements.*
- Recomendación X.700 del CCITT (1992), *Marco de gestión para la interconexión de sistemas abiertos para aplicaciones del CCITT.*
ISO/CEI 7498-4:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework.*
- Recomendación X.710 del CCITT (1991), *Definición del servicio común de información de gestión para aplicaciones del CCITT.*
ISO/CEI 9595:1991, *Information technology – Open Systems Interconnection – Common management information service definition.*
- Recomendación X.711 del CCITT (1991), *Especificación del protocolo común de información de gestión para aplicaciones del CCITT.*
ISO/CEI 9596-1:1991, *Information technology – Open Systems Interconnection – Common management information protocol – Part 1: Specification.*

2.3 Referencias adicionales

- Recomendación UIT-T M.3100 (1995), *Modelo genérico de información de red.*
- Recomendación UIT-T M.3101 (1995), *Declaraciones de conformidad de objeto gestionado para el modelo genérico de información de red.*
ISO/CEI 15068-2²⁾, *Information Technology – Portable Operating System Interface – POSIX System Administration – Part 2: Software Administration (IEEE P1387.2).*

3 Definiciones

A los efectos de la presente Recomendación | Norma Internacional se aplican las siguientes definiciones.

²⁾ Actualmente en estado de proyecto.

3.1 Definiciones de la estructura de gestión

La presente Recomendación | Norma Internacional utiliza los siguientes términos definidos en la Rec. X.700 del CCITT | ISO/CEI 7498-4:

- objeto gestionado.

3.2 Definiciones de la descripción general de gestión de sistemas

La presente Recomendación | Norma Internacional utilizan los siguientes términos definidos en la Rec. X.701 del CCITT | ISO/CEI 10040:

- a) clase de objeto gestionado;
- b) declaración de conformidad de información de gestión (MICS);
- c) declaración de conformidad de objeto gestionado (MOCS);
- d) formulario de MICS;
- e) formulario de MOCS;
- f) notificación.

3.3 Definiciones del servicio común de información de gestión

La presente Recomendación | Norma Internacional utiliza el siguiente término definido en la Rec. X.710 del CCITT e ISO/CEI 9595:

- atributo.

3.4 Definiciones del modelo de información de gestión

La presente Recomendación | Norma Internacional utiliza los siguientes términos definidos en la Rec. X.720 del CCITT | ISO/CEI 10165-1:

- a) acción;
- b) comportamiento;
- c) vinculación de nombre;
- d) lote;
- e) superclase.

3.5 Directrices para la definición de definiciones de objetos gestionados

La presente Recomendación | Norma Internacional utiliza los siguientes términos definidos en la Rec. X.722 del CCITT | ISO/CEI 10165-4:

- plantilla.

3.6 Definiciones del formulario de declaración de conformidad de implementación

La presente Recomendación | Norma Internacional utiliza los siguientes términos definidos en la Rec. UIT-T X.724 | ISO/CEI 10165-6:

- a) declaración de conformidad de relación gestionada (MRCS);
- b) sumario de conformidad de gestión (MCS);
- c) formulario de MCS;
- d) formulario de MRCS.

3.7 Definiciones adicionales

A los efectos de esta Recomendación | Norma Internacional se aplican las siguientes definiciones.

3.7.1 copia de seguridad: La operación **copia de seguridad**, aplicada al objeto gestionado de soporte lógico, hace que se produzca una copia de los recursos subyacentes. No tiene efecto directo sobre los recursos del soporte lógico original.

NOTA – La operación **copia de seguridad** puede aplicarse a otras clases de objeto gestionado; no obstante, por lo que se refiere a esta Recomendación | Norma Internacional, sólo se considera su utilización con soporte lógico.

3.7.2 entrega: La **entrega** de soporte lógico es el proceso de posicionamiento de los recursos asociados con un objeto gestionado de soporte lógico en el sistema junto con el objeto gestionado de soporte lógico.

NOTA – La manera de efectuar la entrega de soporte lógico, y en particular la elección del mecanismo de transferencia, quedan fuera del alcance de la presente Recomendación | Norma Internacional.

3.7.3 ejecutable: Un **soporte lógico ejecutable** representa el soporte lógico cuya ejecución puede ser iniciada. Quizás pueda provocarse la ejecución de soporte lógico ejecutable con una instrucción de gestión, sin embargo, sólo es posible ejecutar soporte lógico mediante una acción local.

3.7.4 instalación: La **instalación** del soporte lógico lo pone en forma adecuada para su utilización, en contraste con la forma en la que puede ser transferido entre sistemas (es decir, entregado). Una vez instalado el soporte lógico, puede ser **utilizado** por otros objetos gestionados.

NOTA – La instalación puede implicar también el establecimiento de relaciones entre objetos gestionados de soporte lógico.

3.7.5 restablecimiento: La operación **restablecimiento** puede ser utilizada para sustituir los recursos asociados a un objeto gestionado soporte lógico con una copia hecha mediante una operación copia de seguridad anterior.

NOTA – Al igual que la operación **copia de seguridad**, la operación **restablecimiento** puede aplicarse a otras clases de objetos gestionados; no obstante, por lo que se refiere a esta Recomendación | Norma Internacional, sólo se considera su utilización con soporte lógico.

3.7.6 reversión: La operación **reversión** invierte el efecto de una operación **instalación** anterior o provoca la eliminación de uno o más parches aplicados.

3.7.7 utilización: La **utilización** está representada por las relaciones entre algunos procesos activos y recursos de soporte lógico. Representa el hecho de que los procesos están utilizando el soporte lógico. El soporte lógico sólo puede ser utilizado cuando tiene un estado operativo de HABILITADO y un estado administrativo de DESBLOQUEADO. Puede haber constricciones respecto al número de cosas y las cosas que pueden utilizar un recurso de soporte lógico. Cuando se emplee soporte lógico, su estado de utilización será ACTIVO u OCUPADO.

NOTA – Desde el punto de vista de la gestión, estas relaciones quedan fuera del alcance de la presente Recomendación | Norma Internacional.

3.7.8 validación: La operación **validación** se utiliza para comprobar la integridad de los recursos de soporte lógico.

NOTA – La manera de efectuar esa comprobación es un asunto local.

4 Símbolos y abreviaturas

A los efectos de esta Recomendación | Norma Internacional se aplican las siguientes abreviaturas:

CMIP	Protocolo común de información de gestión (<i>common management information protocol</i>)
conf.	Confirmación (<i>confirmation</i>)
ICS	Declaración de conformidad de implementación (<i>implementation conformance statement</i>)
Id	Identificador (<i>identifier</i>)
ind.	Indicación (<i>indication</i>)
MCS	Sumario de conformidad de gestión (<i>management conformance summary</i>)
MICS	Declaración de conformidad de información de gestión (<i>management information conformance statement</i>)
MOCS	Declaración de conformidad de objeto gestionado (<i>managed object conformance statement</i>)
MRCS	Declaración de conformidad de relación gestionada (<i>managed relationship conformance statement</i>)
pet.	Petición (<i>request</i>)
PICS	Declaración de conformidad de implementación de protocolo (<i>protocol implementation conformance statement</i>)

5 Convenios

La presente especificación define servicios para el establecimiento y la terminación de relaciones de respaldo mediante copias de seguridad siguiendo los convenios descriptivos definidos en la Rec. UIT-T X.210 | ISO/CEI 10731, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*.

En los cuadros de parámetros de servicio de esta Recomendación | Norma internacional se utiliza la siguiente notación:

- M El parámetro es obligatorio (*mandatory*).
- (=) El valor del parámetro es igual al valor del parámetro de la columna de la izquierda.
- U La utilización del parámetro es una opción del usuario del servicio.
- El parámetro no está presente en la interacción descrita por la primitiva concernida.
- C El parámetro es condicional.
- P El parámetro está sujeto a las constricciones impuestas por la Rec. X.710 del CCITT | ISO/CEI 9595.

6 Requisitos para la gestión de soporte lógico

El soporte lógico ha de ser capaz de realizar las siguientes funciones, consideradas como requisitos, a reserva de la posible imposición de controles y condiciones:

- a) solicitar la entrega de soporte lógico a un sistema gestionado específico;
- b) controlar la instalación de soporte lógico en un sistema gestionado, incluyendo la instalación de correcciones (por ejemplo, mejoras) y de volver a una versión anterior del soporte lógico;
- c) iniciar la ejecución de un programa de soporte lógico;
- d) indagar respecto a los atributos de todo el soporte lógico retenido en un sistema gestionado;
- e) crear y suprimir soporte lógico retenido en un sistema gestionado;
- f) validar soporte lógico retenido en un sistema gestionado para comprobar su integridad y terminar la validación;
- g) restringir la utilización de recursos de soporte lógico en un sistema gestionado a efectos administrativos;
- h) copiar por seguridad un elemento de soporte lógico y restablecer un elemento de soporte lógico copiado anteriormente por el mismo motivo.

El modelo no impedirá que se efectúen registros cronológicos, contabilidad, auditorías y gestión de licencias en cada etapa de este proceso.

En todos los casos, el éxito o el fracaso de la operación debe ser comunicado al sistema de gestión.

Existe una relación entre objetos gestionados de soporte lógico que refleja la manera según la cual unos recursos de soporte lógico utilizan otros recursos de soporte lógico. Esta relación se conoce como «utilización» y se emplea para indicar qué versiones del soporte lógico han de utilizarse en cualquier momento. La gestión de esta relación queda en estudio.

7 Modelo para la función de gestión de soporte lógico

El objetivo principal de la función de gestión de soporte lógico es la gestión de piezas de soporte lógico. En este contexto, el soporte lógico incluye datos, información de control e instrucciones ejecutables. El aspecto gestión de soporte lógico puede representarse mediante la clase de objeto gestionado «**unidad de soporte lógico**». Puesto que la gestión del soporte lógico que representa instrucciones ejecutables tiene características adicionales, se define una segunda clase de objeto gestionado, la «**unidad de soporte lógico ejecutable**» (que es una subclase de la unidad de soporte lógico).

Otro aspecto de la gestión de soporte lógico es la gestión de la entrega de soporte lógico al sistema gestionado. El soporte lógico no necesariamente se gestiona en las mismas unidades en las que se entrega. Por ejemplo, es posible que un cierto número de unidades diferentes de soporte lógico se entreguen juntas a un sistema gestionado (por ejemplo, en un disco compacto ROM). De manera similar, si una unidad de soporte lógico grande ha de ser entregada por una red de comunicaciones, quizás sea necesario dividirla en partes más pequeñas para la entrega. Otro caso es aquel en el que todo lo que se entrega son los cambios necesarios de un objeto de soporte lógico existente, por ejemplo, un parche. La gestión de la entrega se hace en términos del objeto gestionado «**distribución de soporte lógico**».

7.1 Funciones de gestión de soporte lógico

7.1.1 Creación

En un sistema gestionado puede ser creada una unidad de soporte lógico utilizando la operación creación. Una unidad de soporte lógico puede ser creada en el estado creado o bien en el estado entregado. Una unidad de soporte lógico puede ser reproducida o copiada mediante una operación de gestión de creación con objeto de referencia, con referencia a ese objeto gestionado unidad de soporte lógico. También se puede crear una unidad de soporte lógico como consecuencia de la operación entrega en algún objeto gestionado distribución de soporte lógico o de alguna acción local. Como resultado de su creación, un objeto gestionado unidad de soporte lógico puede emitir una notificación de creación de objeto.

7.1.2 Supresión

Una unidad de soporte lógico puede ser suprimida de un sistema gestionado utilizando la operación supresión. Un efecto secundario de la operación supresión en una unidad de soporte lógico puede ser la eliminación de sus recursos subyacentes.

7.1.3 Entrega

Se puede solicitar la entrega de un conjunto coordinado de unidades de soporte lógico. El resultado de la entrega indica el éxito o el fracaso. La entrega se realiza enviando la acción entrega al objeto gestionado distribución de soporte lógico. El resultado de una acción de entrega consiste en la entrega de una copia de los elementos del soporte lógico objetivo al sistema objetivo en el estado entregado. Un efecto lateral puede ser la creación de uno o más objetos asociados (por ejemplo, unidades de soporte lógico).

La agrupación en lotes de las unidades de soporte lógico y la elección del mecanismo de transferencia es un asunto local y está fuera del ámbito de la presente Recomendación | Norma Internacional. Por ejemplo, esta información puede ser preconfigurada o especificada en la acción de entrega junto con cualquier otra información asociada.

7.1.4 Ejecución de programa

Un sistema gestor puede solicitar la ejecución de soporte lógico ejecutable mediante la operación **ejecución de programa**. La presente Recomendación | Norma Internacional no especifica cómo puede ser gestionada seguidamente esa ejecución del soporte lógico ejecutable; simplemente proporciona un mecanismo de iniciación de dicha ejecución.

7.1.5 Obtención

Es posible obtener información sobre soporte lógico (qué soporte lógico está presente, qué soporte lógico está disponible para su utilización, relaciones entre soportes lógicos, etc.) utilizando la operación obtención. Si la operación tiene éxito, el resultado de la misma contiene la información solicitada.

7.1.6 Instalación

La instalación adapta el soporte lógico para su utilización. Esto puede requerir, quizás, una cantidad notable de procesamiento y tiempo, e implicar la comprobación de que todas las partes de la versión del soporte lógico objetivo están presentes en el sistema gestionado (tanto si han sido entregadas como parte de una actualización o ya están presentes) y el ensamblado de las mismas para que estén listas para su utilización. La instalación se realiza mediante la acción de instalación dirigida al objeto gestionado unidad de soporte lógico, del que es una característica opcional.

Un parche es una modificación del soporte lógico y puede ser representado por un objeto gestionado soporte lógico. Un parche puede ser, por tanto, entregado, instalado, utilizado, copiado, etc. empleando la gestión de soporte lógico.

ISO/CEI 10164-18 : 1997 (S)

Un caso especial de instalación es el de un «parche» en la que lo que se entrega es el parche y la aplicación del parche es la instalación y produce una versión actualizada del soporte lógico listo para su utilización. La versión actualizada recibe un número de versión actualizada para identificación, entregado con el parche. El resultado de la instalación indica su éxito o fracaso.

Cuando la entrega del soporte lógico está completa, puede comenzar la instalación. La instalación quizás tenga que ser coordinada, pero la coordinación queda fuera del alcance de la presente Recomendación | Norma Internacional. Ejemplos de coordinación son:

- la compleción y la entrega de varios elementos de soporte lógico antes de que otro elemento de soporte lógico pueda ser instalado;
- la comprobación de que el soporte lógico vigente está fuera de uso antes de habilitar uno nuevo;
- la comprobación de que versiones particulares de otro soporte lógico están (o no están) instaladas;
- la sincronización de la instalación con otros sistemas abiertos, permitiendo la instalación simultánea pero no acoplada en más de un sistema abierto, y proporcionando una manera de que las instalaciones en más de un sistema abierto se agrupen en una sola unidad de trabajo, etc.

7.1.7 Reversión

La operación reversión se utiliza para hacer que un objeto gestionado unidad de soporte lógico instalado vuelva a la situación de no instalado o para provocar la eliminación de uno o más parches aplicados. Los parches que han de ser eliminados se identifican mediante el nombre del fichero o mediante el identificador (Id) del ejemplar de objeto gestionado unidad de soporte lógico. La operación reversión se realiza mediante una acción dirigida al objeto gestionado unidad de soporte lógico, del que es una característica opcional.

Quizás sea necesario mantener información adicional sobre cómo proceder a la reversión del objeto gestionado unidad de soporte lógico. Esto, no obstante, queda fuera del alcance de la presente Recomendación | Norma Internacional.

7.1.8 Fijación del estado administrativo

Una vez que el soporte lógico ha sido adaptado para su utilización puede ponerse a disposición para su uso mediante la operación fijación del estado administrativo a desbloqueado, dirigida al objeto gestionado unidad de soporte lógico. Es posible retirar soporte lógico especificado de un estado de disponibilidad fijando el estado administrativo del objeto gestionado unidad de soporte lógico a cierre o desbloqueando.

7.1.9 Terminación de validación

La validación del soporte lógico puede terminarse utilizando la operación terminación de validación. Esta operación se realiza mediante una acción dirigida al objeto gestionado unidad de soporte lógico, de la que es una característica opcional. La validación puede terminarse prematuramente porque quizá pudiera durar mucho tiempo.

7.1.10 Validación

La integridad del soporte lógico puede comprobarse utilizando la operación validación. También es posible validar que el soporte lógico entregado anteriormente está en condiciones de utilización. El resultado de la validación indica si el soporte lógico fue validado. La operación validación se realiza mediante una acción dirigida al objeto gestionado unidad de soporte lógico, del que es una característica opcional.

7.1.11 Notificaciones

Todas las operaciones aplicables a una unidad de soporte lógico pueden requerir que se envíen los resultados al sistema gestor junto con la confirmación de que la operación se ha completado. También puede ser necesario notificar a otro sistema gestor la compleción de estas operaciones cuando la petición de la operación no se hubiera originado en ese sistema gestor.

7.2 Otras funciones

Las funciones copia de seguridad y restablecimiento son genéricas y están concebidas para ser utilizadas con otros objetos gestionados, además de los objetos de soporte lógico de esta Recomendación | Norma Internacional.

7.2.1 Copia de seguridad

La operación copia de seguridad puede ser utilizada por un sistema gestor para solicitar la realización de una copia de un objeto fijado como objetivo. Cuando se aplica la operación copia de seguridad a un objeto gestionado de soporte lógico se provoca la realización de una copia de los recursos subyacentes. No tiene efecto directo en los recursos del soporte lógico original.

La operación copia de seguridad también puede ser aplicable a otras clases de objetos gestionados, no obstante, por lo que se refiere a esta Recomendación | Norma internacional, sólo se considera su utilización con soporte lógico.

7.2.2 Restablecimiento

La operación restablecimiento puede ser utilizada por un sistema gestor para solicitar la realización del restablecimiento de un objeto pretendido copiado por seguridad previamente.

La operación restablecimiento puede aplicarse también a otras clases de objetos gestionados; sin embargo, en lo que respecta a la presente Recomendación | Norma Internacional, sólo se considera su utilización con soporte lógico.

7.3 Objeto gestionado unidad de soporte lógico

7.3.1 Ciclo vital

Cuando un objeto gestionado unidad de soporte lógico es creado por primera vez, representa el recurso que está preparado para aceptar la entrega de una pieza de soporte lógico.

Hay cuatro mecanismos u operaciones mediante las cuales puede ser creado un objeto gestionado unidad de soporte lógico:

- 1) Reproducción del soporte lógico en un sistema gestionado objetivo sin ninguna operación (por ejemplo, por medios locales).
- 2) Utilización de la operación creación para crear una unidad de soporte lógico en el estado creado o entregado.
- 3) Utilización de la operación creación con objeto de referencia, que hace referencia a un objeto gestionado unidad de soporte lógico fuente, y que está dirigida a un destino objetivo para copiar o reproducir el soporte lógico de origen.
- 4) Utilización de la operación entrega dirigida al objeto gestionado distribución de soporte lógico en el sistema fuente. Esto puede dar lugar a la creación de objetos gestionados unidad de soporte lógico en el destino objetivo en el estado entregado. La operación entrega aplicada a un objeto gestionado distribución de soporte lógico hace que el soporte lógico se entregue a un destino objetivo. Cuando todo el soporte lógico requerido ha sido entregado de manera satisfactoria, el objeto gestionado distribución de soporte lógico emite una notificación de resultado de entrega para señalar que la entrega se ha completado.

Las operaciones creación y creación con objeto de referencia provocan la creación de un objeto gestionado unidad de soporte lógico en el destino objetivo. El éxito de la creación vendría indicado por la existencia de un objeto gestionado unidad de soporte lógico en el destino objetivo (por medio de una respuesta de creación o una notificación de creación de objeto).

Si el objeto gestionado unidad de soporte lógico fuera previamente creado pero no entregado puede emitir una notificación de cambio de estado cuando la entrega se ha completado, indicando que su estado ha cambiado de «creado» a «entregado». El objeto gestionado unidad de soporte lógico también puede ser ejemplificado en el estado entregado.

Una vez que el soporte lógico está en el estado entregado, la siguiente etapa consiste en que el soporte lógico pase a estar instalado. La instalación implica la adaptación del soporte lógico para su utilización. Por ejemplo, la instalación puede implicar la preparación y adaptación de la unidad de soporte lógico para una mejora, tomando quizás una copia de los recursos del soporte lógico vigente y aplicando los cambios a la copia. La instalación puede implicar el establecimiento de relaciones de configuración y dependencia entre la unidad de soporte lógico y otros objetos de soporte lógico.

Una vez instalado el soporte lógico, puede ser utilizado por otros objetos gestionados. Quizás haga falta ponerlo en disposición de uso fijando el estado administrativo a desbloqueado, si se halla en estado bloqueado. Esta operación se lleva a cabo utilizando la operación sustitución en el atributo estado administrativo definido en la Rec. X.731 del CCITT | ISO/CEI 10164-2.

La indisponibilidad para utilización del soporte lógico se consigue bloqueándolo administrativamente, (es decir, fijando el estado administrativo a bloqueado o cierre). De esta manera se evita que algún proceso nuevo utilice el soporte lógico. Si el estado administrativo se fija a bloqueado, todos los procesos en curso en ese momento son abortados. Si el estado administrativo se fija a cierre, se deja que sigan ejecutándose los procesos en curso en ese momento pero no se permiten

ISO/CEI 10164-18 : 1997 (S)

nuevos procesos. Cuando todos los procesos en curso han concluido su ejecución, el estado administrativo se fija automáticamente a bloqueado. Esta operación se lleva a cabo utilizando la operación sustitución en el atributo estado administrativo definido en la Rec. X.731 del CCITT | ISO/CEI 10164-2.

La operación reversión puede utilizarse para invertir el efecto de una operación instalación anterior, o provocar la eliminación de un parche aplicado.

La operación validación hace que se compruebe la integridad del soporte lógico, invocando quizás un algoritmo que genere una suma de control, verificando la existencia o no de virus, etc. También es posible provocar la terminación prematura del proceso de validación, utilizando la operación terminación de validación.

El respaldo del soporte lógico utilizando la operación copia de seguridad se considera equivalente a efectuar una copia del soporte lógico en curso para su utilización en caso de que fallara la versión existente. El soporte lógico copiado por seguridad puede restablecerse utilizando la operación restablecimiento.

7.3.2 Estados del objeto gestionado unidad de soporte lógico

Un objeto gestionado unidad de soporte lógico puede estar en uno de los diversos estados del ciclo vital, dependiendo de la última operación que se realizó en él. Los estados posibles (y su significado) son como sigue:

- Creado – La entrega del producto de soporte lógico no se ha completado, no obstante, algunos recursos arbitrarios del sistema gestionado han sido adjudicados a la unidad de soporte lógico.
- Entregado – La unidad de soporte lógico ha sido entregada de manera satisfactoria al objeto gestionado.
- Instalado – El soporte lógico ha sido instalado de manera satisfactoria en el sistema gestionado.

En el Cuadro 1 se establece la correspondencia entre los estados de un objeto gestionado unidad de soporte lógico y los valores de estado y situación definidos en la Rec. X.731 del CCITT | ISO/CEI 10164-2:

Cuadro 1 – Estados del ciclo vital de la unidad de soporte lógico

Estado de la unidad de soporte lógico	Valor {inicialización requerida} de situación de procedimiento	Valor {no instalado} de situación de disponibilidad
Creado	Presente	Presente
Entregado	Ausente	Presente
Instalado	Ausente	Ausente

Para el objeto gestionado unidad de soporte lógico, el estado administrativo, el estado operativo, la situación de procedimiento y la situación de disponibilidad son atributos de estado obligatorios mientras que el estado de utilización es un atributo opcional.

Los estados creado, entregado e instalado se excluyen mutuamente, es decir, una unidad de soporte lógico debe estar solamente en uno de esos estados en un momento determinado. Con independencia de los estados creado, entregado e instalado existen los estados en validación y en fallo.

Los estados en validación y en fallo son estados secundarios que pueden estar presentes junto con los estados del ciclo vital. El estado en validación se corresponde con un valor de situación de disponibilidad que incluye el valor {en prueba}. Un objeto gestionado unidad de soporte lógico puede pasar al estado en validación, o salir del mismo, independientemente de los valores de sus otros estados. Por ejemplo, una unidad de soporte lógico puede estar en el estado en validación mientras que está en fallo (véase el Cuadro 2).

Cuadro 2 – Estado en validación de la unidad de soporte lógico

Estado de la unidad de soporte lógico	Valor {en prueba} de situación de disponibilidad
En validación	Presente

Una unidad de soporte lógico que esté en el estado creado, entregado o instalado, puede también estar en el estado en fallo. El estado en fallo se corresponde con un valor de situación de disponibilidad que incluye el valor {en fallo}. Las causas específicas para pasar al estado de fallo y salir del mismo, es un asunto local. Un objeto gestionado unidad de soporte lógico puede pasar al estado en fallo, o salir del mismo, independientemente de los valores de sus otros estados. Por ejemplo, el efecto secundario de una operación VALIDACIÓN puede ser una transición al estado en fallo mientras se está en el estado en validación o bien una unidad de soporte lógico puede existir en el estado en fallo como resultado de una operación reversión (véase el Cuadro 3).

Cuadro 3 – Estado en fallo de la unidad de soporte lógico

Estado de la unidad de soporte lógico	Valor {en fallo} de situación de disponibilidad
En fallo	Presente

7.3.3 Operaciones en el objeto gestionado unidad de soporte lógico

Asociadas con un objeto gestionado unidad de soporte lógico están varias operaciones que se utilizan para hacer que cambie de estado.

Dichas operaciones son:

- Copia de seguridad – Hace que el soporte lógico se copie por seguridad en el destino objetivo.
- Creación – Provoca la generación de un nuevo objeto gestionado unidad de soporte lógico en el sistema gestionado.
- Supresión – Hace que el objeto gestionado unidad de soporte lógico sea suprimido del sistema gestionado y puede tener además el efecto de supresión de recursos asociados.
- Instalación – Prepara el soporte lógico para su utilización.
- Restablecimiento – Hace que el soporte lógico copiado por seguridad se restablezca a partir del destino objetivo.
- Reversión – Provoca la reversión de la aplicación de un parche o una instalación.
- Terminación de validación – Hace que se termine el proceso de validación del soporte lógico.
- Validación – Comprueba la integridad del soporte lógico.

Las operaciones creación y supresión se corresponden con las operaciones normalizadas creación y supresión definidas en la Rec. X.720 del CCITT | ISO/CEI 10165-1, mientras que las demás operaciones se corresponden con acciones. Además, pueden ser indagados y modificados valores de atributo utilizando las operaciones obtención y sustitución definidas en la Rec. X.720 del CCITT | ISO/CEI 10165-1.

Sólo se permiten las transiciones de estado del Cuadro 4, en el que únicamente se tienen en cuenta las operaciones que pueden afectar a los estados; se excluyen las operaciones que no provocan un cambio de estado.

7.4 Objeto gestionado soporte lógico ejecutable

La clase de objeto gestionado soporte lógico ejecutable es una subclase de la clase de objeto gestionado unidad de soporte lógico con características adicionales, para describir su funcionalidad como ejecutable. El soporte lógico ejecutable representa soporte lógico que puede ser ejecutado de alguna manera, por medios locales o a distancia, mediante la operación ejecución. Quizás pueda provocarse la ejecución de soporte lógico ejecutable con una instrucción de gestión, sin embargo, sólo es posible ejecutar soporte lógico mediante una acción local.

Aunque quede fuera del alcance de la presente Recomendación | Norma Internacional, el modelo no impide que subclases de soporte ejecutable incluyan información, como la relativa a si el soporte lógico puede tener uno o varios usuarios, en qué condiciones el soporte lógico está activo u ocupado o si se permite un número máximo de usuarios.

7.4.1 Estados adicionales para el objeto gestionado soporte lógico ejecutable

El estado de utilización, definido en la Rec. X.731 del CCITT | ISO/CEI 10164-2, indica si el soporte lógico ejecutable se está utilizando en ese momento.

Cuadro 4 – Matriz de transiciones de estado de la unidad de soporte lógico

Estados	(No existentes)	Creado	Entregado	Instalado
(No existente)	N/A	Creación ^{a)} , Creación con referencia ^{a)} o medios locales	Creación ^{a)} , Creación con referencia ^{a)} o soporte lógico entregado	N/A
Creado	Supresión	–	Soporte lógico entregado	N/A
Entregado	Supresión	N/A	–	Instalación o medios locales ^{a)}
Instalado	Supresión	N/A	Reversión ^{a)}	Reversión ^{a)}
^{a)} Dependiendo del comportamiento del objeto o de los medios locales. N/A No es posible el cambio de estado. – Puede ser aplicable cualquier operación que no provoque un cambio de estado (por ejemplo, copia de seguridad, validación, terminación de validación, indagación, restablecimiento).				

Para la clase de objeto gestionado soporte lógico ejecutable, el estado de utilización es un atributo de estado obligatorio, además de los obligatorios en el objeto gestionado unidad de soporte lógico. En la clase de objeto gestionado soporte lógico ejecutable se permiten los valores de estado de utilización de reposo, activo y ocupado.

7.4.2 Operaciones adicionales para el objeto gestionado soporte lógico ejecutable

Las operaciones que siguen son operaciones en el objeto gestionado soporte lógico ejecutable, además de las definidas para el objeto gestionado unidad de soporte lógico:

- ejecución de programa – Esta operación hace que se ejecute el objeto gestionado de soporte lógico.

La operación ejecución de programa se utiliza para iniciar la ejecución del programa de soporte lógico representado por el soporte lógico ejecutable. El soporte lógico ejecutable debe estar instalado para que pueda ser ejecutado.

El comportamiento del ejemplar específico de soporte lógico ejecutable o los medios locales determinarán el estado del soporte lógico ejecutable cuando haya sido ejecutado.

7.5 Objeto gestionado distribución de soporte lógico

Un objeto gestionado distribución de soporte lógico es un objeto estático que representa el mecanismo o los mecanismos de entrega de un sistema gestionado. Se trata de un objeto gestionado que distribuye soporte lógico al sistema gestionado objetivo cuando recibe una operación entrega procedente del sistema gestor. Los parámetros de la operación entrega pueden ser utilizados para indicar el conjunto de soporte lógico que ha de entregarse, el destino objetivo para la entrega y el mecanismo de transferencia elegido. Aunque esta clase de objeto gestionado puede ser utilizada para iniciar la entrega mediante diversos mecanismos de transferencia, no modela ninguno de esos mecanismos de transferencia. Los modelos se dejan para las especializaciones.

Este objeto gestionado emite una notificación con los resultados de la distribución cuando se ha completado la misma.

7.5.1 Operaciones en el objeto gestionado distribución de soporte lógico

Las siguientes operaciones, asociadas con un objeto gestionado distribución de soporte lógico, son un cierto número de operaciones:

- Creación – Provoca la generación de un nuevo objeto gestionado distribución de soporte lógico.
- Entrega – Hace que el objeto gestionado distribución de soporte lógico efectúe la creación del soporte lógico especificado (por un método que queda fuera del alcance de esta Recomendación | Norma internacional) en el sistema gestionado objetivo y que cualesquiera recursos asociados se asocien con los objetos gestionados unidad de soporte lógico que han de ser creados en el sistema gestionado objetivo como efecto secundario.
- Supresión – Hace que se suprima el objeto gestionado distribución de soporte lógico en el sistema gestionado.

7.5.2 Estados del objeto gestionado distribución de soporte lógico

Un objeto gestionado distribución de soporte lógico puede estar en uno de los estados siguientes, dependiendo de la última operación que se efectuó en él: creado y no existente (véase el Cuadro 5).

Cuadro 5 – Transiciones de estado del objeto gestionado distribución de soporte lógico

Estado de entrada/estado de salida	No existente	Creado
No existente	–	Creación
Creado	Supresión	Entrega

7.6 Relaciones

Se han identificado varias relaciones entre objetos gestionados de soporte lógico y también entre objetos gestionados de soporte lógico y otros objetos gestionados. Son las siguientes:

- Dependencia – Esta relación puede ser utilizada para modelar el hecho de que un objeto gestionado de soporte lógico depende de alguna manera de la presencia de otro objeto gestionado de soporte lógico. Esa relación puede utilizarse para modelar la aplicación de parches.
- Configuración – Esta relación puede ser utilizada para modelar el hecho de que un objeto gestionado unidad de soporte lógico puede afectar al comportamiento de otro objeto gestionado unidad de soporte lógico. Por ejemplo, un tipo de caracteres adicional podría ser modelado como una relación de configuración. Esta relación puede utilizarse también para modelar mejoras y la aplicación de parches.
- Utilización – Esta relación puede ser utilizada para mostrar qué otros objetos gestionados utilizan objetos gestionados de soporte lógico. Esos otros objetos gestionados probablemente sean los que representan procesos en curso en el sistema gestionado.

El fallo de un objeto gestionado puede hacer que fallen también cualesquiera objetos gestionados que dependan de él. Sin embargo, no provocará el fallo de objetos gestionados con una relación de configuración (aunque su comportamiento puede cambiar). La detección de soporte lógico averiado puede dar como resultado la exigencia de que se entregue e instale una nueva copia del soporte lógico o la entrega e instalación de una versión actualizada.

La especificación de cualquiera de estas relaciones, no obstante, queda fuera del alcance de la presente Recomendación | Norma Internacional, ya que las relaciones dependen de la aplicación.

8 Definiciones genéricas

8.1 Objetos gestionados

8.1.1 Soporte lógico

La clase de objeto soporte lógico es una clase de objeto gestionado que representa información lógica almacenada en un elemento gestionado, incluyendo programas y cuadros de datos.

Esta clase de objeto gestionado tiene las siguientes características obligatorias:

- Id de soporte lógico (softwareID).

Esta clase de objeto gestionado tiene los siguientes lotes condicionales (con las características que se indican a continuación):

- Notificaciones de creación y supresión (createDeleteNotificationsPackage) [notificaciones creación de objeto (objectCreation) y supresión de objeto (objectDeletion)].
- Notificación de cambio de valor de atributo (attributeValueChangeNotificationPackage) [notificación cambio de valor de atributo (attributeValueChange)].
- Notificación de cambio de estado (stateChangeNotificationPackage) [notificación cambio de estado (stateChange)].
- Estados operativos y administrativos (administrativeOperationalStatesPackage) [atributos estado administrativo (administrativeState) y estado operativo (operationalState)].

ISO/CEI 10164-18 : 1997 (S)

- Lista de objetos afectados (affectedObjectListPackage) [atributo lista de objetos afectados (affectedObjectList)].
- Alarmas de error de procesamiento de soporte lógico (softwareProcessingErrorAlarmPackage) [atributo situación de alarma (alarmStatus) (sólo lectura) y notificación alarma de error de procesamiento (processingErrorAlarm)].
- Etiqueta de usuario (userLabelPackage) [atributo etiqueta de usuario (userLabel)].
- Nombre de vendedor (vendorNamePackage) [atributo nombre de vendedor (vendorName)].
- Versión (versionPackage) (atributo versión).
- Lista de problemas actuales (currentProblemListPackage) [atributo lista de problemas actuales (currentProblemList)].

Esta clase de objeto gestionado se define en la Recomendación M.3100.

8.1.2 Unidad de soporte lógico

La clase de objeto unidad de soporte lógico es una clase de objeto gestionado que proporciona información administrable asociada con el soporte lógico (ya sea en forma de un fichero ejecutable, como por ejemplo un soporte lógico de programa, o un fichero no ejecutable, como por ejemplo un cuadro de correspondencias de datos o transconexiones), que reside físicamente en el medio de memoria.

Esta clase de objeto gestionado tiene las siguientes características obligatorias:

- Atributo estado administrativo (administrativeState).
- Atributo situación de disponibilidad (availabilityStatus).
- Atributo estado operativo (operationalState).
- Situación de procedimiento (proceduralStatus).
- Notificación de alarma de error de procesamiento (processingErrorAlarm) con parámetro de efecto de alarma en servicio (alarmEffectOnServiceParameter).

Esta clase de objeto gestionado tiene los siguientes lotes condicionales (con las características que se indican a continuación):

- Correcciones aplicadas (appliedPatchPackage) [atributo correcciones aplicadas (appliedPatches)].
- Suma de control (checksumPackage) [atributo suma de control (checksum)].
- Información de fichero (fileInformationPackage) [atributos fecha de creación (dateOfCreation), identidad de creador (identityOfCreator), fecha de última modificación (dateOfLastModification), identidad de último modificador (identityOfLastModifier), fecha de entregado (dateDelivered) y fecha de instalado (dateInstalled)].
- Fichero (filePackage) [atributos tipo de fichero (fileType), ubicación de fichero (fileLocation) y tamaño de fichero (fileSize)].
- Copia de seguridad automática de información (informationAutoBackupPackage) [notificación de informe de copia de seguridad automática (autoBackupReport), atributos umbral futuro de activación de copia de seguridad automática (futureAutoBackupTriggerThreshold) y destino futuro de copia de seguridad automática (futureAutoBackupDestination)].
- Restablecimiento automático de información (informationAutoRestorePackage) [notificación informe de restablecimiento automático (autoRestoreReport), atributos fuente restablecimiento automático futuro (futureAutoRestoreSource) y permitido restablecimiento futuro (futureAutoRestoreAllowed)].
- Copia de seguridad de información (informationBackupPackage) [acción de copia de seguridad, atributos tiempo de última copia de seguridad (lastBackupTime) y destino de última copia de seguridad (lastBackupDestination)].
- Restablecimiento de información (informationRestorePackage) [acción de restablecimiento, atributos tiempo de último restablecimiento (lastRestoreTime) y fuente de último restablecimiento (lastRestoreSource)].
- Instalación (installPackage) (acción de instalación).
- Campo de nota (noteFieldPackage) [atributo campo de nota (noteField)].
- Reversión (revertPackage) (acción de reversión).
- Terminación validación (terminateValidationPackage) (acción terminación validación).

- Estado de utilización (usageStatePackage) [atributo estado de utilización (usageState)].
- Validación (validationPackage) (acción validación).

8.1.3 Soporte lógico ejecutable

La clase de objeto soporte lógico ejecutable es una clase de objeto gestionado que proporciona información administrable asociada con un programa ejecutable en el sistema gestionado. Esta clase de objeto es una subclase del objeto gestionado unidad de soporte lógico. El programa ejecutable real (que puede constar de segmentos de códigos con o sin segmentos de datos, etc.) puede tener un formato no normalizado, dependiente de la máquina, que por lo general no es legible por el sistema gestor.

Una acción llamada ejecución de programa (executeProgram) (definida más abajo) puede utilizarse para ejecutar el programa representado por el ejemplar de objeto soporte lógico ejecutable. El atributo estado de utilización se utiliza para indicar si hay algunas ejecuciones del programa activas. El valor de reposo significa que no hay ejecuciones activas. El significado de los valores activo y ocupado es específico de la realización y quedan fuera del alcance de la presente Recomendación | Norma Internacional. El valor activo por ejemplo, podría significar que existe alguna ejecución en curso, mientras que el valor ocupado significa que el programa ha alcanzado su máxima capacidad y que cualquier otra petición de ejecución (utilizando la acción de ejecución de programa o por algún otro medio local) puede ser rechazada o puesta en fila de espera para posterior ejecución. En las especializaciones quizás se desee incluir un umbral de número de usuarios necesarios para alcanzar la capacidad máxima.

Esta clase de objeto gestionado tiene las siguientes características obligatorias adicionales:

- atributo estado de utilización (usageState).

Esta clase de objeto gestionado tiene los siguientes lotes condicionales adicionales (con las características que se indican a continuación):

- Lote ejecución de programa (executeProgramPackage) [acción de ejecución de programa (executeProgram)].

8.1.4 Distribución de soporte lógico

Un objeto gestionado distribución de soporte lógico es un objeto estático que representa el mecanismo o los mecanismos de entrega de un sistema gestionado. Se trata de un objeto gestionado que distribuye soporte lógico especificado a un sistema gestionado objetivo especificado cuando recibe una operación entrega procedente del sistema gestor. Los parámetros de la operación entrega se utilizan para indicar el conjunto de soporte lógico que ha de entregarse, el destino objetivo para la entrega y el mecanismo de transferencia elegido. Aunque esta clase de objeto puede ser utilizada para iniciar la entrega mediante diversos mecanismos de transferencia, no modela ninguno de esos mecanismos de transferencia. Los modelos se dejan para las especializaciones.

Este objeto gestionado emite una notificación con los resultados de la distribución cuando se ha completado la misma. La notificación de cambio de estado (stateChangeNotification) definida en la Rec. X.721 del CCITT | ISO/CEI 10165-2 se emitirá si cambia el valor del estado administrativo o del estado operativo. La clase de objeto gestionado distribución de soporte lógico es una subclase de la clase de objeto gestionado tope (top) definida en la Rec. X.721 del CCITT | ISO/CEI 10165-2.

Esta clase de objeto gestionado tiene las siguientes características obligatorias:

- Atributo estado administrativo (AdministrativeState).
- Atributo estado operativo (OperationalState).
- Id de distribución de soporte lógico (softwareDistributorId).
- Acción de entrega.
- Notificación de resultado de entrega (deliverResultNotification).
- Notificación de creación de objeto (objectCreation).
- Notificación de supresión de objeto (objectDeletion).
- Notificación de cambio de estado (stateChange).

8.2 Atributos

8.2.1 Parches aplicados (appliedPatches)

Este atributo identifica las correcciones o parches (patches) que han sido aplicados, y que todavía existen, a la unidad de soporte lógico representada por el ejemplar de objeto unidad de soporte lógico. Los parches son actualizaciones del soporte lógico. El valor de este atributo es de lectura solamente y se actualiza automáticamente cuando se efectúa una

ISO/CEI 10164-18 : 1997 (S)

corrección en el soporte lógico. Su sintaxis es una secuencia de identificadores de parches, siendo un identificador de parche un ejemplar de objeto (si el parche está representado por un objeto gestionado de unidad de soporte lógico) o una cadena gráfica (si la corrección no está representada por un objeto gestionado unidad de soporte lógico).

8.2.2 Suma de control (checksum)

Este atributo identifica la suma de control de la información de soporte lógico representada por el ejemplar de objeto Unidad de soporte lógico.

8.2.3 Fecha de entregado (dataDelivered)

Este atributo identifica la hora en que se entregó al sistema gestionado la información representada por el ejemplar de objeto unidad de soporte lógico. Son valores válidos para este atributo GeneralizedTime (tiempo generalizado) de ASN.1 o NULL (nulo) si la información no ha sido entregada.

8.2.4 Fecha de instalado (dateInstalled)

Este atributo identifica la hora en que se instaló la información representada por el ejemplar de objeto unidad de soporte lógico. Son valores válidos para este atributo GeneralizedTime de ASN.1 o NULL si la información no ha sido instalada.

8.2.5 Fecha de creación (dateOfCreation)

Este atributo indica la hora de la creación del objeto gestionado unidad de soporte lógico. La sintaxis de este atributo es del tipo GeneralizedTime de ASN.1. El valor inicial obligatorio de este atributo es la hora real (disponible por medios locales) de la creación del ejemplar de objeto.

8.2.6 Fecha de la última modificación (dateOfLastModification)

Este atributo indica la hora de la última modificación, o de la más reciente, (por ejemplo, aplicación de parche, reversión, instalación, entrega) de la información representada por el soporte lógico. Son valores válidos para este atributo GeneralizedTime de ASN.1 o NULL si la información no ha sido modificada.

NOTA – Cuando la última modificación es una reversión, el tiempo de reversión indica la fecha de la última modificación.

8.2.7 Ubicación de fichero (fileLocation)

Este atributo identifica la dirección completa (lógica o física) del fichero o ficheros de soporte lógico representados por un objeto unidad de soporte lógico. La sintaxis de este atributo es un conjunto de cadenas gráficas o bien es nula. El formato de una dirección depende de la implementación, de conformidad con los convenios relativos al direccionamiento de ficheros del sistema gestionado de que se trate, por lo que la dirección viene representada por una cadena gráfica en la sintaxis. Un valor nulo de este atributo indica que la información a la que se aplica el objeto gestionado unidad de soporte lógico todavía no ha sido entregada al sistema gestionado.

8.2.8 Tamaño de fichero (fileSize)

Este atributo indica el tamaño del objeto gestionado unidad de soporte lógico.

8.2.9 Tipo de fichero (fileType)

Este atributo indica el tipo de unidad de soporte lógico. Los tipos posibles de unidad de soporte lógico son el fichero binario no estructurado (por ejemplo, fichero ejecutable), el fichero de texto no estructurado (por ejemplo, fichero no ejecutable), el fichero especial en bloque, etc.

8.2.10 Destino futuro de copia de seguridad automática (futureAutoBackupDestination)

Este atributo especifica el destino en el que se copiará la información representada por el ejemplar de objeto. El criterio de copia de seguridad se define en el atributo umbral futuro de activación de copia de seguridad automática (futureAutoBackupTriggerThreshold) del ejemplar de objeto. El destino puede ser otro ejemplar de objeto de la misma clase de objeto que existe en el mismo sistema gestionado, un sistema abierto distante (utilizando un protocolo de transferencia de fichero particular, por ejemplo FTAM), o el sistema gestor (utilizando la notificación informe de copia de seguridad automática (autoBackupReport) en línea).

8.2.11 Umbral futuro de activación de copia de seguridad automática (futureAutoBackupTriggerThreshold)

Este atributo especifica el umbral que activará una copia de seguridad automática de la información representada por el ejemplar de objeto. El umbral viene definido por el número de veces que ha sido modificada la información. Una vez que la información ha sido modificada ese número de veces, se efectúa una copia de seguridad automática. El destino de la copia se especifica en el atributo destino de copia de seguridad automática futura (futureAutomaticBackupDestination). Las copias de seguridad se hacen además de otras copias periódicas planificadas. Al completarse la copia de seguridad automática, se emitirá desde el objeto una notificación de informe de copia de seguridad automática (autoBackupReport).

8.2.12 Permitido restablecimiento automático futuro (futureAutoRestoreAllowed)

Este atributo especifica si se permite el restablecimiento automático de la información representada por el objeto gestionado. La sintaxis de este atributo es del tipo BOOLEAN de ASN.1, con el valor TRUE (verdadero) para permitido y FALSE (falso) para no permitido. Los criterios de activación del restablecimiento automático de información son específicos del sistema.

8.2.13 Fuente de restablecimiento automático futuro (futureAutoRestoreSource)

Este atributo especifica la fuente de la información que ha de restablecerse a la información representada por el ejemplar de objeto gestionado. La fuente es un objeto gestionado local o un sistema distante. Los criterios de activación del restablecimiento automático de información son específicos del sistema.

8.2.14 Identidad de creador (identityOfCreator)

Este atributo identifica la entidad que crea el objeto gestionado.

8.2.15 Identidad de último modificador (identityOfLastModifier)

Este atributo identifica el último, o más reciente, modificador de la información representada por el ejemplar de objeto gestionado unidad de soporte lógico.

8.2.16 Destino de última copia de seguridad (lastBackupDestination)

Este atributo identifica el destino, si existe, en el que se copia la información representada por el objeto gestionado.

8.2.17 Tiempo de última copia de seguridad (lastBackupTime)

Este atributo identifica el tiempo de la última copia de seguridad de la información representada por el ejemplar de objeto gestionado. Son valores válidos para este atributo GeneralizedTime de ASN.1 o NULL si no se ha efectuado ninguna copia de seguridad de la información.

8.2.18 Fuente de último restablecimiento (lastRestoreSource)

Este atributo identifica la fuente, si existe, desde la que se restablece la información representada por el objeto gestionado.

8.2.19 Tiempo de último restablecimiento (lastRestoreTime)

Este atributo identifica el tiempo del último restablecimiento en la información representada por el ejemplar de objeto gestionado. Son valores válidos para este atributo GeneralizedTime de ASN.1 o NULL si no se ha efectuado ningún restablecimiento en la información.

8.2.20 Campo de nota (noteField)

Este atributo contiene cualquier información o comentario asociado con el objeto gestionado, incluyendo cualesquiera instrucciones de instalación específicas, parámetros y valores de arranque, información necesaria para activar características del objeto gestionado, etc.

8.2.21 Id de distribución de soporte lógico (softwareDistributorId)

El Id (identificador) de distribución de soporte lógico se utiliza para identificar los ejemplares de la clase de objeto gestionado distribución de soporte lógico.

8.3 Acciones

El conjunto de parámetros de acciones genéricas y las semánticas definidos por la presente Recomendación | Norma Internacional proporcionan los detalles de los siguientes parámetros generales del servicio M-ACCIÓN definido por la Rec. X.710 del CCITT | ISO/CEI 9595:

- tipo de acción;
- información de acción;
- réplica de acción.

8.3.1 Copia de seguridad

El servicio copia de seguridad es utilizado por un sistema gestor para pedir que se efectúe una copia de seguridad de la información representada por el ejemplar de objeto objetivo, o de la información almacenada en el medio de memoria representado por otro ejemplar de clase de objeto.

Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará copia de seguridad. El parámetro información de acción indicará el destino en el que la información debe copiarse. Son posibles destinos los siguientes:

- Un objeto gestionado local – En este caso, la operación copia de seguridad se efectuará internamente en el sistema gestionado.
- El sistema gestor desde el que se envía esta acción – En este caso, se enviará una copia de la información copiada por seguridad en línea en el parámetro réplica de acción.
- Un sistema distante – En este caso, la información copiada por seguridad será transferida fuera de línea al sistema distante utilizando algún medio local.

El parámetro acción de respuesta indicará el resultado de la copia de seguridad. En caso de copia local o fuera de línea, un valor NULL (nulo) indica que la copia de seguridad se ha hecho de manera satisfactoria. En caso de copia en línea, la información copiada se incluirá en el parámetro réplica de acción.

8.3.2 Entrega

El servicio entrega es utilizado por un sistema gestor que pide la distribución de un soporte lógico o de un conjunto de soportes lógicos. La información de la acción de entrega identifica el soporte lógico que ha de distribuirse. Puede entregarse un cierto número de unidades de soporte lógico junto a un sistema gestionado. El resultado de la compleción satisfactoria consiste en que el soporte lógico que ha de ser distribuido es copiado en el sistema objetivo, lo que puede tener el efecto de una creación de objetos unidad de soporte lógico.

Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará entrega.

8.3.3 Ejecución de programa

El servicio ejecución de programa es utilizado por un sistema gestor para iniciar la ejecución de un programa representado por un objeto de soporte lógico ejecutable.

Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará ejecución de programa.

8.3.4 Instalación

El servicio instalación es utilizado por un sistema de gestión para instalar un objeto unidad de soporte lógico entregado. Si la unidad de soporte lógico no puede ser instalada debido a una condición de estado no válida, se devuelve un error específico discordancia de estados de operación (operationStateMismatch).

Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará instalación.

8.3.5 Restablecimiento

El servicio restablecimiento es utilizado por un sistema gestor para pedir la realización de un restablecimiento en la información representada por el ejemplar de objeto fijado como objetivo. Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará

restablecimiento. El parámetro información de acción indicará la fuente a partir de la cual se restablecerá la información. Son posibles fuentes las siguientes:

- Un objeto gestionado local – En este caso, la operación restablecimiento se efectuará internamente en el sistema gestionado.
- El sistema gestor desde el que se envía esta acción – En este caso, se enviará una copia de la información de restablecimiento en línea en el parámetro de información de acción.
- Un sistema distante – En este caso, la información de restablecimiento será transferida fuera de línea desde el sistema distante utilizando algún medio local.

Esta acción es siempre confirmada.

8.3.6 Reversión

El servicio reversión es utilizado por un sistema gestor (por ejemplo, un sistema operativo) para indicar a un sistema gestionado que vuelva a su estado original un parche o un conjunto de parches aplicados al soporte lógico representado por el objeto gestionado unidad de soporte lógico.

El servicio reversión devolverá automáticamente el valor del atributo parches aplicados (appliedPatches) del ejemplar de objeto unidad de soporte lógico al que está dirigido el servicio. Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará reversión. El parámetro información de acción identificará el parche o los parches que han de ser revertidos del ejemplar de objeto unidad de soporte lógico. Cada identificador de parche es un identificador específico de sistema (Printable String, cadena imprimible) o un ejemplar de objeto (Object Instance) unidad de soporte lógico.

8.3.7 Terminación de validación

El servicio terminación de validación es utilizado por un sistema gestor para abortar una validación que en esos momentos está activa (en curso). Una condición previa para esta operación es que el objeto receptor de la petición esté validándose en esos momentos; de no ser así, el parámetro réplica de acción indica la inexistencia de un proceso de validación activa que hubiera que terminar.

Se proporcionan dos modos de terminación para terminar una validación, a saber, el modo cancelación y el modo truncado. En caso de modo truncado, se terminará la validación y el resultado de la validación completada parcialmente se devolverá en el parámetro réplica de acción del M-ACCIÓN terminación de validación. En caso de modo cancelación, se terminará la validación y se descartará el resultado de la validación completada parcialmente.

Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará terminación de validación. El parámetro información de acción indicará el modo de la terminación, es decir, modo cancelación o modo truncado. El parámetro réplica de acción indicará el resultado de la terminación, es decir, validación terminada (para modo cancelación), validación terminada con resultado parcial (para el modo truncado) o inexistencia de validación activa que haya que abortar.

8.3.8 Validación

El servicio validación es utilizado por un sistema gestor para pedir la realización de una validación en la información representada por el ejemplar objeto unidad de soporte lógico. Este servicio utiliza el servicio M-ACCIÓN y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de acción indicará validación. El parámetro información de acción indicará el tipo de validación que ha de efectuarse (utilizando un conjunto de extensiones de gestión). El parámetro réplica de acción indicará el resultado de la validación. Si el proceso de validación se termina (utilizando el M-ACCIÓN terminación de validación) antes de la compleción normal de la validación, se devolverá el valor «terminado» (con NULL (nulo) tipo ASN.1) en la réplica de acción del M-ACCIÓN validación. Si el modo de terminación del M-ACCIÓN terminación de validación es «truncado» el resultado de la validación truncada (es decir la parte de la validación que ya se ha completado) debe devolverse en el parámetro réplica de acción del M-ACCIÓN terminación de validación.

8.4 Notificaciones

El conjunto de notificaciones genéricas, parámetros y semánticas definidos por la presente Recomendación | Norma Internacional proporciona los detalles de los siguientes parámetros generales del servicio M-INFORME-DE-EVENTO definido por la Rec. X.710 del CCITT | ISO/CEI 9595:

- tipo de evento;
- información de evento;
- respuesta de evento.

8.4.1 Informe de copia de seguridad automático (autoBackupReport)

La notificación autoBackupReport se emite para informar de una copia de seguridad automática de la información representada por este objeto.

Los criterios relativos a la copia de seguridad automática y al destino de la copia de seguridad se especifican en los atributos umbral activador de copia de seguridad automática futura (futureAutoBackupTriggerThreshold) y destino de copia de seguridad automática futura (futureAutoBackupDestination) del objeto, respectivamente.

El destino de la copia de seguridad puede ser local (por ejemplo, copia a otro objeto de la misma clase dentro del sistema gestionado local) el sistema gestor o fuera de línea a un sistema distante utilizando un protocolo de transferencia de ficheros particular (por ejemplo, FTAM). En caso de copia de seguridad local y fuera de línea, el resultado de la copia, es decir, el éxito o el fracaso, se comunicará en esta notificación. En caso de copia al sistema gestor, la información de la copia será en línea incluida en la notificación como un componente del parámetro información de evento.

Este servicio utiliza el servicio M-INFORME-DE-EVENTO y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de evento indicará informe de copia de seguridad automática.

8.4.2 Informe de restablecimiento automático (autoRestoreReport)

La notificación autoRestoreReport es emitida desde el objeto gestionado cuando se produce un restablecimiento automático en la información representada por este ejemplar de objeto. Los criterios que provocan el restablecimiento automático son específicos del sistema. La fuente de la información restablecida y el resultado de la operación restablecimiento (es decir, su éxito o su fracaso) se comunicarán en la notificación.

Este servicio utiliza el servicio M-INFORME-DE-EVENTO y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de evento indicará informe de restablecimiento automático.

8.4.3 Notificación de resultado de entrega (deliverResultNotification)

Este tipo de notificación se utiliza para informar de los resultados de la entrega y es emitido desde el objeto gestionado cuando se completa la entrega. El parámetro resultado de entrega puede tomar uno de los valores siguientes: paso, fallo o desconocido.

Este servicio utiliza el servicio M-INFORME-DE-EVENTO y los procedimientos definidos en la Rec. X.710 del CCITT | ISO/CEI 9595. El parámetro tipo de evento indicará resultado de entrega.

8.5 Parámetros

8.5.1 Parámetro de efecto de alarma en servicio (alarmEffectOnServiceParameter)

El parámetro alarmEffectOnServiceParameter es un parámetro que ha de incluirse en el parámetro ManagementExtension (extensión de gestión) del parámetro AdditionalInformation (información adicional) del parámetro AlarmInfo (información de alarma) en un servicio señalador de alarmas. Este parámetro indica si la condición que produce la alarma afectará al servicio.

El parámetro ManagementExtension es de la forma (véase la Rec. X.721 del CCITT | ISO/CEI 10165-2):

```
ManagementExtension ::= SEQUENCE {  
    identifier OBJECT IDENTIFIER,  
    significance [1] BOOLEAN DEFAULT FALSE,  
    information [2] ANY DEFINED BY identifier}
```

El OBJECT IDENTIFIER (identificador de objeto) llevado en identifier (identificador) será el valor bajo el cual se registra esta definición de parámetro. El tipo llevado en information (información) será el tipo identificado por la construcción WITH SYNTAX de esta definición de parámetro.

8.5.2 Parámetro fallo de procesamiento de soporte lógico (softwareProcessingFailureParameter)

El parámetro softwareProcessingFailureParameter define la sintaxis de datos que se ha de devolver en una respuesta error de fallo de procesamiento del CMIP (CMIP ProcessingFailure) a M-ACCIÓN si se rechaza una petición de operación en soporte lógico debido a errores distintos de los ya definidos en CMIP-1. Los atributos en la sintaxis de datos devuelta incluyen los atributos de estado del objeto emisor. Se podrán incluir otros atributos aplicables pero esto será un asunto local.

8.6 Vinculaciones de nombre

8.6.1 Distribución de soporte lógico – Subsistema

Esta vinculación de nombre se utiliza para denominar un objeto distribución de soporte lógico con respecto a un objeto subsistema.

8.6.2 Distribución de soporte lógico – Sistema

Esta vinculación de nombre se utiliza para denominar un objeto distribución de soporte lógico con respecto a un objeto sistema.

8.6.3 Unidad de soporte lógico – Subsistema

Esta vinculación de nombre se utiliza para denominar un objeto unidad de soporte lógico con respecto a un objeto subsistema.

8.6.4 Unidad de soporte lógico – Sistema

Esta vinculación de nombre se utiliza para denominar un objeto unidad de soporte lógico con respecto a un objeto sistema.

9 Definiciones de servicios

9.1 Introducción

En esta cláusula se definen servicios para la entrega de soporte lógico, ejecución de un programa, instalación de soporte lógico, reversión de soporte lógico, terminación de validación de soporte lógico y validación de soporte lógico. Se definen también servicios con los que se informa de copias de seguridad y restablecimientos automáticos de soporte lógico. Además, se definen servicios genéricos para copia de seguridad y restablecimiento.

9.2 Servicio copia de seguridad (backup)

El servicio copia de seguridad permite a un gestor pedir que otro sistema abierto (el sistema gestionado) respalde una entidad mediante una copia de seguridad. El Cuadro 6 contiene la lista de parámetros de este servicio.

En el servicio copia de seguridad se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

El parámetro respuesta de copia de seguridad estará si la respuesta es positiva, de no ser así, el parámetro presente será errores.

9.3 Servicio de entrega (deliver)

El servicio entrega permite a un gestor pedir que otro sistema abierto (el sistema gestionado) inicie la entrega de soporte lógico. El Cuadro 7 contiene la lista de parámetros de este servicio.

En el servicio entrega se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

El parámetro información de resultado de entrega estará presente si la respuesta es positiva, de no ser así, el parámetro presente será errores.

9.4 Servicio ejecución de programa (execute program)

El servicio ejecución de programa permite a un gestor pedir que otro sistema abierto (el sistema gestionado) ejecute el soporte lógico del programa. El Cuadro 8 contiene la lista de parámetros de este servicio.

El servicio ejecución de programa utiliza los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

El parámetro respuesta de ejecución de programa estará presente si la respuesta es positiva, de no ser así, estará presente el parámetro errores.

Cuadro 6 – Parámetros de copia de seguridad

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de copia de seguridad	M	C(=)
Argumento de copia de seguridad	M	–
Destino de copia de seguridad	M	–
Información adicional	U	–
Tiempo actual	–	P
Respuesta de copia de seguridad	–	C
Respuesta	–	C
Información adicional	–	U
Errores	–	C

Cuadro 7 – Parámetros de entrega

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de entrega	M	C(=)
Información de entrega	M	–
Id de entrega	U	–
Soporte lógico objetivo	M	–
Sistema objetivo	U	–
Información de transferencia	U	–
Información adicional	U	–
Tiempo actual	–	P
Información de resultado de entrega	–	C
Id de entrega	–	U
Resultado de entrega	–	C
Información adicional	–	U
Errores	–	C

Cuadro 8 – Parámetros de ejecución de programa

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de ejecución de programa	M	C(=)
Información de ejecución de programa	M	–
Tiempo actual	–	P
Respuesta de ejecución de programa	–	C
Id de proceso	–	C
Propietario de proceso	–	C
Tiempo de arranque	–	C
Información adicional	–	U
Errores	–	C

9.5 Servicio instalación (install)

El servicio instalación permite a un gestor pedir que otro sistema abierto (el sistema gestionado) instale soporte lógico. El Cuadro 9 contiene la lista de parámetros de este servicio.

En el servicio instalación se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

Si no hay una respuesta positiva, el parámetro errores estará presente en la respuesta.

Cuadro 9 – Parámetros de instalación

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de instalación	M	C(=)
Información de instalación	M	–
Soporte lógico objetivo	M	–
Información de instalación	M	–
Tiempo actual	–	P
Errores	–	C

9.6 Servicio restablecimiento (restore)

El servicio restablecimiento permite a un gestor pedir que otro sistema abierto (el sistema gestionado) restablezca una entidad. El Cuadro 10 contiene la lista de parámetros de este servicio.

En el servicio restablecimiento se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

Si no hay una respuesta positiva, el parámetro errores estará presente en la respuesta.

Cuadro 10 – Parámetros de restablecimiento

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de restablecimiento	M	C(=)
Argumento de restablecimiento	M	–
Fuente de restablecimiento	M	–
Información adicional	U	–
Tiempo actual	–	P
Errores	–	C

9.7 Servicio reversión (revert)

El servicio reversión permite a un gestor pedir que otro sistema abierto (el sistema gestionado) revierta soporte lógico. El Cuadro 11 contiene la lista de parámetros de este servicio.

En el servicio reversión se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

El parámetro respuesta de reversión estará presente si la respuesta es positiva, de no ser así, estará presente el parámetro errores.

9.8 Servicio terminación de validación (terminate validation)

El servicio terminación de validación permite a un gestor pedir que otro sistema abierto (el sistema gestionado) termine la validación de un soporte lógico. El Cuadro 12 contiene la lista de parámetros de este servicio.

En el servicio terminación de validación se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

El parámetro respuesta de terminación de validación estará presente si la respuesta es positiva, de no ser así, estará presente el parámetro errores.

Cuadro 11 – Parámetros de reversión

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de reversión	M	C(=)
Información de reversión	M	–
Id de corrección	M	–
Información adicional	U	–
Tiempo actual	–	P
Respuesta de reversión	–	C
Parches aplicados	–	C
Información adicional	–	U
Errores	–	C

Cuadro 12 – Parámetros de terminación de validación

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de terminación de validación	M	C(=)
Argumento de terminación de validación	M	–
Información de terminación de validación	M	–
Información adicional	U	–
Tiempo actual	–	P
Respuesta de terminación de validación	–	C
Errores	–	C

9.9 Servicio validación (validate)

El servicio validación permite a un gestor pedir que otro sistema abierto (el sistema gestionado) valide soporte lógico. El Cuadro 13 contiene la lista de parámetros de este servicio.

En el servicio validación se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-ACCIÓN definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

El parámetro respuesta de validación estará presente si la respuesta es positiva, de no ser así, estará presente el parámetro errores.

9.10 Servicio informe de copia de seguridad automática (Auto backup report)

El servicio informe de copia de seguridad automática permite a un sistema abierto (el sistema gestionado) informar a propósito de una copia de seguridad automática. El Cuadro 14 contiene la lista de parámetros de este servicio.

En el servicio informe de copia de seguridad automática se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-INFORME-DE-EVENTO definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

9.11 Servicio informe de restablecimiento automático (Auto restore report)

El servicio informe de restablecimiento automático permite a un sistema abierto (el sistema gestionado) informar de un restablecimiento automático. El Cuadro 15 contiene la lista de parámetros de este servicio.

En el servicio informe de restablecimiento automático se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-INFORME-DE-EVENTO definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

Cuadro 13 – Parámetros de validación

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Identificador vinculado	–	P
Modo	P	–
Clase de objeto de base	P	–
Ejemplar de objeto de base	P	–
Alcance	P	–
Filtro	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Control de acceso	P	–
Sincronización	P	–
Tipo de validación	M	C(=)
Información de validación	M	–
Tiempo actual	–	P
Respuesta de validación	–	C
Errores	–	C

Cuadro 14 – Parámetros de informe de copia de seguridad automática

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Modo	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Tipo de informe de copia de seguridad automática	M	C(=)
Tiempo de evento	P	–
Información de informe de copia de seguridad automática	M	–
Resultado de copia de seguridad	M	–
Información adicional	U	–
Tiempo actual	–	P
Respuesta de evento	–	–
Errores	–	C

Cuadro 15 – Parámetros de informe de restablecimiento automático

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Modo	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Tipo de informe de restablecimiento automático	M	C(=)
Tiempo de evento	P	–
Información de informe de restablecimiento automático	M	–
Fuente	M	–
Éxito	M	–
Información adicional	U	–
Tiempo actual	–	P
Respuesta de evento	–	–
Errores	–	C

9.12 Servicio notificación de resultado de entrega (Deliver result notification)

El servicio notificación de resultado de entrega permite a un sistema abierto (el sistema gestionado) informar del resultado de la entrega del soporte lógico. El Cuadro 16 contiene la lista de parámetros de este servicio.

En el servicio notificación de resultado de entrega se utilizan los parámetros definidos en la cláusula 8, además de los parámetros generales del servicio M-INFORME-DE-EVENTO definidos en la Rec. X.710 del CCITT e ISO/CEI 9595.

Cuadro 16 – Parámetros de notificación de resultado de entrega

Nombre de parámetro	pet./ind.	resp./conf.
Identificador de invocación	P	P
Modo	P	–
Clase de objeto gestionado	–	P
Ejemplar de objeto gestionado	–	P
Tipo de notificación de resultado de entrega	M	C(=)
Tiempo de evento	P	–
Información de resultado de entrega	M	–
Id de entrega	U	–
Resultado de entrega	M	–
Información adicional	U	–
Tiempo actual	–	P
Respuesta de evento	–	–
Errores	–	C

10 Unidades funcionales

En la presente Recomendación | Norma Internacional se definen tres unidades funcionales para la gestión del soporte lógico y otras entidades:

- a) unidad funcional control de soporte lógico;
- b) unidad funcional entrega de soporte lógico;
- c) unidad funcional restablecimiento de copia de seguridad.

La unidad funcional control de soporte lógico requiere el soporte de los servicios ejecución de programa, instalación, reversión, terminación de validación y validación. La unidad funcional entrega de soporte lógico requiere el soporte de los servicios entrega y notificación de resultado de entrega. La unidad funcional restablecimiento de copia de seguridad requiere el soporte de los servicios acción de copia de seguridad, acción de restablecimiento, informe de copia de seguridad automática, notificación y notificación de informe de restablecimiento automático.

11 Protocolo

11.1 Elementos de procedimiento

11.1.1 Procedimiento de copia de seguridad

11.1.1.1 Cometido del gestor

11.1.1.1.1 Invocación

Los procedimientos de copia de seguridad se inician por medio de la primitiva de copia de seguridad. Al recibir una primitiva de copia de seguridad la máquina de protocolo de aplicación de gestión de sistemas (SMAPM) construirá una unidad de datos de protocolo de aplicación de gestión (MAPDU) y emitirá una primitiva del servicio común de información de gestión (CMIS) petición M-ACCIÓN con parámetros derivados de la primitiva de copia de seguridad. Se utilizará el modo confirmado.

11.1.1.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación copia de seguridad, la SMAPM emitirá una primitiva confirmación de copia de seguridad al usuario del servicio copia de seguridad con parámetros derivados de la primitiva de servicio CMIS confirmación M-ACCIÓN, concluyendo así el procedimiento de copia de seguridad.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio copia de seguridad puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.1.2 Cometido de agente**11.1.1.2.1 Recepción de petición**

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio copia de seguridad, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de copia de seguridad al usuario del servicio copia de seguridad con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.1.2.2 Respuesta

La SMAPM aceptará una primitiva respuesta de copia de seguridad y construirá una MAPDU que confirme la operación, y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva respuesta de copia de seguridad.

11.1.2 Procedimiento de entrega**11.1.2.1 Cometido de gestor****11.1.2.1.1 Invocación**

Los procedimientos de entrega se inician por medio de la primitiva de entrega. Al recibir una primitiva de entrega, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de entrega. Se utilizará el modo confirmado.

11.1.2.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación entrega, la SMAPM emitirá una primitiva confirmación de entrega al usuario del servicio entrega con parámetros derivados de la primitiva del servicio CMIS, confirmación M-ACCIÓN concluyendo así el procedimiento de entrega.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio entrega puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.2.2 Cometido de agente**11.1.2.2.1 Recepción de petición**

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio entrega, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de entrega al usuario del servicio entrega con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.2.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de entrega y construirá una MAPDU que confirme la operación y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de entrega.

11.1.3 Procedimiento de ejecución de programa**11.1.3.1 Cometido de gestor****11.1.3.1.1 Invocación**

Los procedimientos de ejecución de programa se inician por medio de la primitiva de ejecución de programa. Al recibir una primitiva de ejecución de programa, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de ejecución de programa. Se utilizará el modo confirmado.

11.1.3.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación ejecución de programa, la SMAPM emitirá una primitiva confirmación de ejecución de programa al usuario del servicio ejecución de programa con parámetros derivados de la primitiva del servicio CMIS confirmación M-ACCIÓN, concluyendo así el procedimiento de ejecución de programa.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio ejecución de programa puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.3.2 Cometido de agente

11.1.3.2.1 Recepción de petición

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio ejecución de programa, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de ejecución de programa al usuario del servicio ejecución de programa con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.3.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de ejecución de programa y construirá una MAPDU que confirme la operación, y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de ejecución de programa.

11.1.4 Procedimiento de instalación

11.1.4.1 Cometido de gestor

11.1.4.1.1 Invocación

Los procedimientos de instalación se inician por medio de la primitiva de instalación. Al recibir una primitiva de instalación, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de iniciación. Se utilizará el modo confirmado.

11.1.4.1.2 Recepción de respuesta

Al recibir una primitiva de servicio confirmación M-ACCIÓN de CMIS que contiene una MAPDU que responde a una operación instalación, la SMAPM emitirá una primitiva confirmación de instalación al usuario del servicio instalación con parámetros derivados de la primitiva del servicio CMIS confirmación M-ACCIÓN, concluyendo así el procedimiento de instalación.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio instalación puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.4.2 Cometido de agente

11.1.4.2.1 Recepción de petición

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio de instalación, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de instalación al usuario del servicio instalación con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.4.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de instalación y construirá una MAPDU que confirme la operación y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de instalación.

11.1.5 Procedimiento de restablecimiento**11.1.5.1 Cometido de gestor****11.1.5.1.1 Invocación**

Los procedimientos de restablecimiento se inician por medio de la primitiva de restablecimiento. Al recibir una primitiva de restablecimiento, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de restablecimiento. Se utilizará el modo confirmado.

11.1.5.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación restablecimiento, la SMAPM emitirá una primitiva confirmación de restablecimiento al usuario del servicio restablecimiento con parámetros derivados de la primitiva del servicio CMIS confirmación M-ACCIÓN, completando así el procedimiento de restablecimiento.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio restablecimiento puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.5.2 Cometido de agente**11.1.5.2.1 Recepción de petición**

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio restablecimiento, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de restablecimiento al usuario del servicio restablecimiento con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada indicando el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.5.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de restablecimiento y construirá una MAPDU que confirme la operación, y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de restablecimiento.

11.1.6 Procedimiento de reversión**11.1.6.1 Cometido de gestor****11.1.6.1.1 Invocación**

Los procedimientos de reversión se inician por medio de la primitiva de reversión. Al recibir una primitiva de reversión, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de reversión. Se utilizará el modo confirmado.

11.1.6.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación reversión, la SMAPM emitirá una primitiva de confirmación de reversión al usuario del servicio reversión con parámetros derivados de la primitiva del servicio CMIS confirmación M-ACCIÓN, concluyendo así el procedimiento de reversión.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio reversión puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.6.2.1 Recepción de petición

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio reversión, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de reversión al usuario del servicio reversión con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.6.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de reversión y construirá una MAPDU que confirme la operación, y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de reversión.

11.1.7 Procedimiento de terminación de validación

11.1.7.1 Cometido de gestor

11.1.7.1.1 Invocación

Los procedimientos de terminación de validación se inician por medio de la primitiva de terminación de validación. Al recibir una primitiva de terminación de validación, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de terminación de validación. Se utilizará el modo confirmado.

11.1.7.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación terminación de validación, la SMAPM emitirá una primitiva confirmación de terminación de validación al usuario del servicio terminación de validación con parámetros derivados de la primitiva del servicio CMIS confirmación M-ACCIÓN, concluyendo así el procedimiento de terminación de validación.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio terminación de validación puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.7.2 Cometido de agente

11.1.7.2.1 Recepción de petición

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU que pide el servicio terminación de validación, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva de indicación de terminación de validación al usuario del servicio terminación de validación con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetros errores presente.

11.1.7.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de terminación de validación y construirá una MAPDU que confirme la operación, y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de terminación de validación.

11.1.8 Procedimiento de validación

11.1.8.1 Cometido de gestor

11.1.8.1.1 Invocación

Los procedimientos de validación se inician por medio de la primitiva de validación. Al recibir una primitiva de validación, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-ACCIÓN con parámetros derivados de la primitiva de validación. Se utilizará el modo confirmado.

11.1.8.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-ACCIÓN que contiene una MAPDU que responde a una operación validación, la SMAPM emitirá una primitiva confirmación de validación al usuario del servicio validación con parámetros derivados de la primitiva del servicio CMIS confirmación M-ACCIÓN, concluyendo así el procedimiento de validación.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio validación puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.8.2 Cometido de agente**11.1.8.2.1 Recepción de petición**

Al recibir una primitiva del servicio CMIS indicación M-ACCIÓN que contiene una MAPDU pidiendo el servicio validación, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de validación al usuario del servicio validación con parámetros derivados de la primitiva del servicio CMIS indicación M-ACCIÓN. De otro modo, la SMAPM construirá una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con un parámetro errores presente.

11.1.8.2.2 Respuesta

La SMAPM aceptará una primitiva de respuesta de validación y construirá una MAPDU que confirme la operación y emitirá una primitiva del servicio CMIS respuesta M-ACCIÓN con parámetros derivados de la primitiva de respuesta de validación.

11.1.9 Procedimiento de informe de copia de seguridad automática**11.1.9.1 Cometido de agente****11.1.9.1.1 Invocación**

Los procedimientos de informe de copia de seguridad automática se inician por medio de la primitiva petición de informe de copia de seguridad automática. Al recibir una primitiva petición de informe de copia de seguridad automática, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-INFORME-DE-EVENTO con parámetros derivados de la primitiva de petición de informe de copia de seguridad automática. En el modo no confirmado no es aplicable el procedimiento de 11.1.9.1.2.

11.1.9.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-INFORME-DE-EVENTO que contiene una MAPDU que responde a una notificación de informe de copia de seguridad automática, la SMAPM emitirá una primitiva confirmación de informe de copia de seguridad automática al usuario del servicio informe de copia de seguridad automática con parámetros derivados de la primitiva del servicio CMIS confirmación M-INFORME-DE-EVENTO, concluyendo así el procedimiento de informe de copia de seguridad automática.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio informe de copia de seguridad automática puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.9.2 Cometido de gestor**11.1.9.2.1 Recepción de petición**

Al recibir una primitiva del servicio CMIS indicación M-INFORME-DE-EVENTO que contiene una MAPDU que pide el servicio informe de copia de seguridad automática, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de informe de copia de seguridad automática al usuario del servicio informe de copia de seguridad automática con parámetros derivados de la primitiva del servicio CMIS indicación M-INFORME-DE-EVENTO. De otro modo, la SMAPM construirá, en el modo confirmado, una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-INFORME-DE-EVENTO con un parámetro de errores presente. En el modo no confirmado no es aplicable el procedimiento de 11.1.9.2.2.

11.1.9.2.2 Respuesta

En el modo confirmado, la SMAPM aceptará una primitiva respuesta de informe de copia de seguridad automática y construirá una MAPDU que confirme a la notificación, y emitirá una primitiva del servicio CMIS respuesta M-INFORME-DE-EVENTO con parámetros derivados de la primitiva de respuesta de informe de copia de seguridad automática.

11.1.10 Procedimiento de informe de restablecimiento automático

11.1.10.1 Cometido de agente

11.1.10.1.1 Invocación

Los procedimientos de informe de restablecimiento automático se inician por medio de la primitiva petición de informe de restablecimiento automático. Al recibir una primitiva petición de informe de restablecimiento automático, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-INFORME-DE-EVENTO con parámetros derivados de la primitiva petición de informe de restablecimiento automático. En el modo no confirmado no es aplicable el procedimiento de 11.1.10.1.2.

11.1.10.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-INFORME-DE-EVENTO que contiene una MAPDU que responde a una notificación de informe de restablecimiento automático, la SMAPM emitirá una primitiva confirmación de informe de restablecimiento automático al usuario del servicio informe de restablecimiento automático con parámetros derivados de la primitiva del servicio CMIS confirmación M-INFORME-DE-EVENTO, concluyendo así el procedimiento de informe de restablecimiento automático.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio informe de restablecimiento automático puede hacer caso omiso de tales errores o abortar la asociación como consecuencia de los mismos.

11.1.10.2 Cometido de gestor

11.1.10.2.1 Recepción de petición

Al recibir una primitiva del servicio CMIS indicación M-INFORME-DE-EVENTO que contiene una MAPDU que pide el servicio informe de restablecimiento automático, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de informe de restablecimiento automático al usuario del servicio informe de restablecimiento automático con parámetros derivados de la primitiva del servicio CMIS indicación M-INFORME-DE-EVENTO. De otro modo, la SMAPM construirá, en el modo confirmado, una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-INFORME-DE-EVENTO con un parámetro de errores presente. En el modo no confirmado no es aplicable el procedimiento de 11.1.10.2.2.

11.1.10.2.2 Respuesta

En el modo confirmado, la SMAPM aceptará una primitiva respuesta de informe de restablecimiento automático y construirá una MAPDU que confirme la notificación, y emitirá una primitiva del servicio CMIS respuesta M-INFORME-DE-EVENTO con parámetros derivados de la primitiva de respuesta informe de restablecimiento automático.

11.1.11 Procedimiento de notificación de resultado de entrega

11.1.11.1 Cometido de agente

11.1.11.1.1 Invocación

Los procedimientos de notificación de resultado de entrega se inician por medio de la primitiva petición de notificación de resultado de entrega. Al recibir una primitiva petición de notificación de resultado de entrega, la SMAPM construirá una MAPDU y emitirá una primitiva del servicio CMIS petición M-INFORME-DE-EVENTO con parámetros derivados de la primitiva petición de notificación de resultado de entrega. En el modo no confirmado no es aplicable el procedimiento de 11.1.11.1.2.

11.1.11.1.2 Recepción de respuesta

Al recibir una primitiva del servicio CMIS confirmación M-INFORME-DE-EVENTO que contiene una MAPDU que responde a una notificación de resultado de entrega, la SMAPM emitirá una primitiva confirmación de notificación de resultado de entrega al usuario del servicio notificación de resultado de entrega con parámetros derivados de la primitiva del servicio CMIS confirmación M-INFORME-DE-EVENTO, concluyendo así el procedimiento de notificación de resultado de entrega.

NOTA – La SMAPM pasará por alto todos los errores de la MAPDU recibida. El usuario del servicio de notificación de resultado de entrega puede ignorar tales errores o abortar la asociación como consecuencia de los mismos.

11.1.11.2 Cometido de gestor**11.1.11.2.1 Recepción de petición**

Al recibir una primitiva del servicio CMIS indicación M-INFORME-DE-EVENTO que contiene una MAPDU que pide el servicio notificación de resultado de entrega, la SMAPM emitirá, si la MAPDU está bien formada, una primitiva indicación de notificación de resultado de entrega al usuario del servicio notificación de resultado de entrega con parámetros derivados de la primitiva del servicio CMIS indicación M-INFORME-DE-EVENTO. De otro modo, la SMAPM construirá, en el modo confirmado, una MAPDU apropiada que indique el error y emitirá una primitiva del servicio CMIS respuesta M-INFORME-DE-EVENTO con un parámetro de errores presente. En el modo no confirmado no es aplicable el procedimiento de 11.1.11.2.2.

11.1.11.2.2 Respuesta

En el modo confirmado, la SMAPM aceptará una primitiva respuesta de notificación de resultado de entrega y construirá una MAPDU que confirme la notificación, y emitirá una primitiva del servicio CMIS respuesta M-INFORME-DE-EVENTO con parámetros derivados de la primitiva de respuesta de notificación de resultado de entrega.

11.2 Sintaxis abstracta**11.2.1 Objetos**

En la presente Recomendación | Norma Internacional se hace referencia a los siguientes objetos de soporte, cuya notación en valor ASN.1 se especifica en el Anexo A:

- a) unidad de soporte lógico (softwareUnit);
- b) soporte lógico ejecutable (executableSoftware);
- c) distribución de soporte lógico (softwareDistributor).

11.2.2 Lotes

En la presente Recomendación | Norma Internacional se hace referencia a las siguientes definiciones de lotes cuya notación en valor de ASN.1 se especifica en el Anexo A:

- a) corrección aplicada (appliedPatchPackage);
- b) suma de control (checksumPackage);
- c) ejecución de programa (executeProgramPackage);
- d) información de fichero (fileInformationPackage);
- e) lote fichero (filePackage);
- f) copia de seguridad automática de información (informationAutoBackupPackage);
- g) restablecimiento automático de información (informationAutoRestorePackage);
- h) copia de seguridad de información (informationBackupPackage);
- i) restablecimiento de información (informationRestorePackage);
- j) instalación (installPackage);
- k) campo de nota (noteFieldPackage);
- l) alarma de error de procesamiento en servicio (processingErrorAlarmOnServicePackage);

ISO/CEI 10164-18 : 1997 (S)

- m) reversión (revertPackage);
- n) terminación de validación (terminateValidationPackage);
- o) estado de utilización (usageStatePackage);
- p) validación (validationPackage).

11.2.3 Atributos

En la presente Recomendación | Norma Internacional se hace referencia a los siguientes atributos de gestión específicos cuya sintaxis abstracta se indica en el Anexo A:

- a) correcciones aplicadas (appliedPatches);
- b) suma de control (checksum);
- c) fecha de entregado (dateDelivered);
- d) fecha de instalado (dateInstalled);
- e) fecha de creación (dateOfCreation);
- f) fecha de última modificación (dateOfLastModification);
- g) ubicación de fichero (fileLocation);
- h) tamaño de fichero (fileSize);
- i) tipo de fichero (fileType);
- j) destino de futuro de copia de seguridad automática (futureAutoBackupDestination);
- k) umbral futuro de activación de copia de seguridad automática (futureAutoBackupTriggerThreshold);
- l) permitido restablecimiento automático futuro (futureAutoRestoreAllowed);
- m) fuente de restablecimiento automático futuro (futureAutoRestoreSource);
- n) identidad de creador (identityOfCreator);
- o) identidad de último modificador (identityOfLastModifier);
- p) destino de última copia de seguridad (lastBackupDestination);
- q) tiempo de última copia de seguridad (lastBackupTime);
- r) fuente de último restablecimiento (lastRestoreSource);
- s) tiempo de último restablecimiento (lastRestoreTime);
- t) campo de nota (noteField);
- u) Id de distribución de soporte lógico (softwareDistributorId).

11.2.4 Notificaciones

En la presente Recomendación | Norma Internacional se hace referencia a los siguientes tipos de notificaciones específicas, cuya sintaxis abstracta se indica en el Anexo A:

- a) informe de copia de seguridad automática (autoBackupReport);
- b) informe de restablecimiento automático (autoRestoreReport);
- c) notificación de resultado de entrega (deliverResultNotification).

11.2.5 Acciones

En la presente Recomendación | Norma Internacional se hace referencia a los siguientes tipos de acciones específicas, cuya sintaxis abstracta se indica en el Anexo A:

- a) copia de seguridad (backup);
- b) entrega (deliver);
- c) ejecución de programa (executeProgram);
- d) instalación (install);

- e) restablecimiento (restore);
- f) reversión (revert);
- g) terminación de validación (terminateValidation);
- h) validación (validate).

11.2.6 Vinculaciones de nombre

En la presente Recomendación | Norma Internacional se hace referencia a las siguientes vinculaciones de nombre específicas, cuya sintaxis abstracta se indica en el Anexo A:

- a) distribución de soporte lógico-subsistema (softwareDistributor-subsystem);
- b) distribución de soporte lógico-sistema (softwareDistributor-system);
- c) unidad de soporte lógico-subsistema (softwareUnit-subsystem);
- d) unidad de soporte lógico-sistema (softwareUnit-system).

11.3 Negociación de unidades funcionales

En la presente Recomendación | Norma Internacional se asigna el siguiente identificador de objeto:

{joint-iso-ccitt ms(9) function(2) part18(18) functionalUnitPackage(1)}

como un valor del tipo FunctionalUnitPackageId (Id de lote unidad funcional) de ASN.1 definido en la Rec. X.701 del CCITT | ISO/CEI 10040 para negociar las siguientes unidades funcionales:

- 0 unidad funcional control de soporte lógico
- 1 unidad funcional entrega del soporte lógico
- 2 unidad funcional restablecimiento de copia de seguridad

donde el número identifica la posición de bit asignada a la unidad funcional y el nombre hace referencia a la unidad funcional definida en la cláusula 10.

Dentro del contexto de aplicación de gestión de sistemas, el mecanismo para la negociación de la unidad funcional señaladora de alarmas se describe en la Rec. X.701 del CCITT | ISO/CEI 10040.

NOTA – El requisito de negociar unidades funcionales lo especifica el contexto de aplicación.

12 Relación con otras funciones

Las siguientes funciones son proporcionadas por otras funciones de gestión de sistemas:

- calidad de funcionamiento del soporte lógico, cubierta por la función de recapitulación (véase la Rec. UIT-T X.738 | ISO/CEI 10164-13) y los objetos métricos y la función de atributos (véase la Rec. UIT-T X.739 | ISO/CEI 10164-11);
- rastreo de auditoría del soporte lógico, cubierta por la función de rastreo de auditoría de la seguridad (véase la Rec. UIT-T X.740 | ISO/CEI 10164-8);
- soporte de la seguridad del soporte lógico, cubierta por los objetos y atributos de control de acceso (véase la Rec. UIT-T X.741 | ISO/CEI 10164-9);
- contabilidad de la utilización de soporte lógico, cubierta por la función de cómputo de la utilización para fines de contabilidad (véase la Rec. UIT-T X.742 | ISO/CEI 10164-10);
- prueba del soporte lógico (que incluye la instalación del entorno de prueba, ejecución de la prueba del soporte lógico, fijación de puntos de corte y suspensión y reanudación del entorno de prueba del soporte lógico), cubierta por la función de gestión de pruebas (véase la Rec. UIT-T X.745 | ISO/CEI 10164-12);
- planificación de las funciones y operaciones del soporte lógico, cubierta por la función de planificación (véase la Rec. UIT-T X.746 | ISO/CEI 10164-15).

Las implementaciones que aleguen conformidad con la presente Recomendación | Norma Internacional deberán cumplir los requisitos de conformidad definidos en las subcláusulas que siguen.

13.1 Conformidad estática

La implementación deberá ser conforme a los requisitos de esta Recomendación | Norma Internacional en el cometido de gestor, en el cometido de agente o en ambos cometidos. Se hará una alegación de conformidad con al menos uno de los cometidos del Cuadro B.1.

Si se hace una alegación de conformidad para sustentar en el cometido de gestor, la implementación deberá soportar por lo menos una operación de gestión o notificación del objeto gestionado fichero registro cronológico o por lo menos una operación de gestión de una subclase del objeto gestionado registro de fichero registro cronológico especificado en esta Recomendación | Norma Internacional. Los requisitos de conformidad en el cometido de gestor para esas operaciones de gestión y notificaciones se especifican en el Anexo B y en otros cuadros referenciados por Anexo B.

Si se hace una alegación de conformidad para sustentar el cometido de agente, la implementación deberá soportar uno o más ejemplares de la clase de objeto gestionado fichero registro cronológico identificado en el Cuadro B.4. En el cometido de agente, también puede alegarse conformidad con subclases del objeto registro de fichero registro cronológico.

La implementación deberá soportar la sintaxis de transferencia derivada de las reglas de codificación especificadas en la Rec. X.209 del CCITT | ISO/CEI 8825 indicadas por {joint-iso-ccitt asn1(1) basicEncoding(1)} para los tipos de datos abstractos referenciados por las definiciones cuyo soporte se alega.

13.2 Conformidad dinámica

Las implementaciones que aleguen conformidad con esta Recomendación | Norma Internacional deberán soportar los elementos de procedimiento y las definiciones de semántica correspondientes a las definiciones cuyo soporte se alega.

13.3 Requisitos de la declaración de conformidad de implementación de gestión

Cualquier formulario de MCS, MOCS, MRCS que sea conforme a esta Recomendación | Norma Internacional será técnicamente idéntico a los formularios especificados en los Anexos A, B, C, D y E, preservando la numeración de los cuadros y los números de índice de los ítems y difiriendo solamente en la paginación y en los encabezamientos de las páginas.

El suministrador de una implementación que alegue conformidad con esta Recomendación | Norma Internacional deberá completar una copia del sumario de conformidad de gestión (MCS) proporcionado en el Anexo B como parte de los requisitos de conformidad, junto con cualesquiera otros formularios de declaración de conformidad de implementación (ICS) a los que se hace referencia como aplicables en ese MCS. Un formulario ICS que sea conforme a esta Recomendación | Norma Internacional deberá:

- describir una implementación que sea conforme a esta Recomendación | Norma Internacional;
- haber sido completado de acuerdo con las instrucciones que se dan a tal fin en la Rec. UIT-T X.724 | ISO/CEI 10165-6;
- incluir la información necesaria para identificar inequívocamente tanto al suministrador como a la implementación.

Las alegaciones de conformidad a la información de gestión definidas en esta Recomendación | Norma Internacional en las clases de objeto gestionado definidas en otros lugares incluirán los requisitos del formulario de MIDS, como se especifica en el Anexo F, en el MOCS para la clase de objeto gestionado.

Anexo A

Definición de información de gestión

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

--A.1 Managed Object Classes

--A.1.1 Software

-- *This object class is defined in Rec. M.3100.*

-- *(See Annex C.)*

--A.1.2 softwareUnit

softwareUnit **MANAGED OBJECT CLASS**

DERIVED FROM "Rec. M.3100:1995":software;

CHARACTERIZED BY

"Rec. M.3100:1995":administrativeOperationalStatesPackage,
 softwareUnitPackage **PACKAGE**
 BEHAVIOUR softwareUnitBehaviour;
 ATTRIBUTES
 "Rec. X.721 | ISO/IEC 10165-2:1992":availabilityStatus GET,
 "Rec. X.721 | ISO/IEC 10165-2:1992":proceduralStatus GET;;,
 processingErrorAlarmOnServicePackage;

CONDITIONAL PACKAGES

appliedPatchPackage **PRESENT IF** "an instance supports software patching",
 checksumPackage **PRESENT IF** "an instance supports check sum validation",
 fileInformationPackage **PRESENT IF** "an instance supports file information",
 filePackage **PRESENT IF** "an instance supports representation of a file",
 informationAutoBackupPackage **PRESENT IF** "an instance supports automatic backup",
 informationAutoRestorePackage **PRESENT IF** "an instance supports automatic restore",
 informationBackupPackage **PRESENT IF** "an instance supports the backup operation",
 informationRestorePackage **PRESENT IF** "an instance supports the restore operation",
 installPackage **PRESENT IF** "an instance supports the install operation",
 noteFieldPackage **PRESENT IF** "an instance supports it",
 revertPackage **PRESENT IF** "an instance supports it",
 terminateValidationPackage **PRESENT IF** "the validationPackage is present and an instance supports it",
 usageStatePackage **PRESENT IF** "an instance supports it",
 validationPackage **PRESENT IF** "an instance supports it";

REGISTERED AS {SWMF.softwareManagement managedObjectClass(3) softwareUnit(1)};

softwareUnitBehaviour **BEHAVIOUR**

DEFINED AS

"The softwareUnit object class is a class of managed objects that provide administrable information associated with software (whether it be in the form of an executable file, such as program software, or a non-executable file, such as a data or cross-connect mapping table). The file type, file location, and file size are among the attributes identified in this object class. When the fileInformationPackage is present, the mandatory initial value of the dateOfCreation attribute is the time that the managed object is created.

When the attribute value change notification package (inherited from the superclass software) is present, the attributeValueChange notification defined in Recommendation X.721 shall be emitted when the value of one of the following attribute changes:

- futureAutoBackupTriggerThreshold;
- futureAutoBackupDestination;
- futureAutoRestoreSource; and
- futureAutoRestoreAllowed.

Because some of the above attributes are in conditional packages, the behaviour for emitting the attributeValueChange notification applies only when the corresponding conditional packages are present in the managed object."

--A.1.3 executableSoftware**executableSoftware** MANAGED OBJECT CLASS

DERIVED FROM softwareUnit;

CHARACTERIZED BY

executableSoftwarePackage PACKAGE

BEHAVIOUR executableSoftwareBehaviour;;

usageStatePackage;

CONDITIONAL PACKAGES

executeProgramPackage

PRESENT IF "an instance supports it";

REGISTERED AS { SWMF.softwareManagement managedObjectClass(3) executableSoftware(2)};

executableSoftwareBehaviour BEHAVIOUR

DEFINED AS

"The executableSoftware object class is a class of managed objects that provide administrable information associated with an executable program in the managed system. The actual executable program (that may consist of code segments with or without data segments, etc.) may be in a non-standard, machine-dependent format that is generally unreadable by managing system and the rest of the outside world. An action called executeProgram (conditionally) can be used to execute the program represented by the executableSoftware object instance. The usageState attribute is used to indicate if there are any active executions of the program.";

--A.1.4 softwareDistributor**softwareDistributor** MANAGED OBJECT CLASS

DERIVED FROM "Rec. X.721 | ISO/IEC 10165-2:1992":top;

CHARACTERIZED BY

softwareDistributorPackage PACKAGE

BEHAVIOUR softwareDistributorBehaviour;

ATTRIBUTES

"Rec. X.721 | ISO/IEC 10165-2:1992":administrativeState GET-REPLACE,

"Rec. X.721 | ISO/IEC 10165-2:1992":operationalState GET,

softwareDistributorId GET;

ACTIONS

deliver;

NOTIFICATIONS

deliverResultNotification,

"Rec. X.721 | ISO/IEC 10165-2:1992":objectCreation,

"Rec. X.721 | ISO/IEC 10165-2:1992":objectDeletion,

"Rec. X.721 | ISO/IEC 10165-2:1992":stateChange ;;;

REGISTERED AS {SWMF.softwareManagement managedObjectClass(3) softwareDistributor(3)};

softwareDistributorBehaviour BEHAVIOUR

DEFINED AS

"A Software distributor managed object is a managed object which distributes software to the target managed system when it receives a deliver operation from the managing system. This managed object notifies the result of the distribution when the distribution is terminated. The stateChangeNotification defined in Rec. X.721 | ISO/IEC 10165 2:1992 shall be emitted if the value of the administrative state or operational state changes. The Software distributor managed object class is a subclass of the top managed object class defined in Rec. X.721 | ISO/IEC 10165-2:1992.";

--A.2 Packages**--A.2.1 appliedPatchPackage****appliedPatchPackage** PACKAGE

ATTRIBUTES

appliedPatches GET;

REGISTERED AS {SWMF.softwareManagement package(4) appliedPatchPackage(1)};

--A.2.2 checkSumPackage

checkSumPackage PACKAGE

ATTRIBUTES

**checkSum
GET;**

REGISTERED AS {SWMF.softwareManagement package(4) checkSumPackage(2)};

--A.2.3 executeProgramPackage

executeProgramPackage PACKAGE

ACTIONS

executeProgram;

REGISTERED AS {SWMF.softwareManagement package(4) executeProgramPackage(3)};

--A.2.4 fileInformationPackage

fileInformationPackage PACKAGE

ATTRIBUTES

**dateOfCreation GET,
identityOfCreator GET,
dateOfLastModification GET,
identityOfLastModifier GET,
dateDelivered GET,
dateInstalled GET;**

REGISTERED AS {SWMF.softwareManagement package(4) fileInformationPackage(4)};

--A.2.5 filePackage

filePackage PACKAGE

ATTRIBUTES

**fileLocation GET,
fileSize GET,
fileType GET;**

REGISTERED AS {SWMF.softwareManagement package(4) filePackage(5)};

--A.2.6 informationAutoBackupPackage

informationAutoBackupPackage PACKAGE

ATTRIBUTES

**futureAutoBackupTriggerThreshold GET-REPLACE,
futureAutoBackupDestination GET-REPLACE;**

NOTIFICATIONS

autoBackupReport;

REGISTERED AS {SWMF.softwareManagement package(4) informationAutoBackupPackage(6)};

--A.2.7 informationAutoRestorePackage

informationAutoRestorePackage PACKAGE

ATTRIBUTES

**futureAutoRestoreSource GET-REPLACE,
futureAutoRestoreAllowed GET-REPLACE;**

NOTIFICATIONS

autoRestoreReport;

REGISTERED AS {SWMF.softwareManagement package(4) informationAutoRestorePackage(7)};

--A.2.8 informationBackupPackage

informationBackupPackage PACKAGE

ATTRIBUTES

lastBackupTime GET,
lastBackupDestination GET;

ACTIONS

backup softwareProcessingFailureParameter;

REGISTERED AS {SWMF.softwareManagement package(4) informationBackupPackage(8)};

--A.2.9 informationRestorePackage

informationRestorePackage PACKAGE

ATTRIBUTES

lastRestoreTime GET,
lastRestoreSource GET;

ACTIONS

restore;

REGISTERED AS {SWMF.softwareManagement package(4) informationRestorePackage(9)};

--A.2.10 installPackage

installPackage PACKAGE

ACTIONS

install;

REGISTERED AS {SWMF.softwareManagement package(4) installPackage(10)};

--A.2.11 noteFieldPackage

noteFieldPackage PACKAGE

ATTRIBUTES

noteField GET-REPLACE;

REGISTERED AS {SWMF.softwareManagement package(4) noteFieldPackage(11)};

--A.2.12 processingErrorAlarmOnServicePackage

processingErrorAlarmOnServicePackage PACKAGE

NOTIFICATIONS

"Rec. X.721 | ISO/IEC 10165-2:1992":processingErrorAlarm
alarmEffectOnServiceParameter;

REGISTERED AS {SWMF.softwareManagement package(4) processingErrorAlarmOnServicePackage(12)};

--A.2.13 revertPackage

revertPackage PACKAGE

ACTIONS

revert softwareProcessingFailureParameter ;

REGISTERED AS {SWMF.softwareManagement package(4) revertPackage(13)};

--A.2.14 terminateValidationPackage

terminateValidationPackage PACKAGE

ACTIONS

terminateValidation;

REGISTERED AS {SWMF.softwareManagement package(4) terminateValidationPackage(14)};

--A.2.15 usageStatePackage

usageStatePackage PACKAGE

ATTRIBUTES

"Rec. X.721 | ISO/IEC 10165-2:1992":usageState GET;

REGISTERED AS {SWMF.softwareManagement package(4) usageStatePackage(15)};

--A.2.16 validationPackage

validationPackage PACKAGE

ACTIONS

validate;

REGISTERED AS {SWMF.softwareManagement package(4) validatePackage(16)};

--A.3 Attributes**--A.3.1 appliedPatches**

appliedPatches ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.AppliedPatches;

MATCHES FOR EQUALITY;

BEHAVIOUR appliedPatchesBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) appliedPatches(1)};

appliedPatchesBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the patches that have been applied to and still exist in the software which is represented by the software unit object instance. Patches are updates to executable programs. The value of this attribute is read-only and automatically updated when a patch is applied to the software.";

--A.3.2 checksum

checksum ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.CheckSum;

MATCHES FOR EQUALITY;

BEHAVIOUR checksumBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) checksum(2)};

checksumBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the checksum of the software information represented by the softwareUnit object instance.";

--A.3.3 dateDelivered

dateDelivered ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.Date;

MATCHES FOR EQUALITY;

BEHAVIOUR dateDeliveredBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) dateDelivered(3)};

dateDeliveredBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the time that the information represented by the softwareUnit object instance was delivered to the managed system. Valid values for this attribute are ASN.1 GeneralizedTime or NULL if the information has not been delivered.";

--A.3.4 dateInstalled

dateInstalled **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.Date;

MATCHES FOR **EQUALITY**;

BEHAVIOUR dateInstalledBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) dateInstalled(4)};

dateInstalledBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute identifies the time that the information represented by the softwareUnit object instance was installed. Valid values for this attribute are ASN.1 GeneralizedTime or NULL if the information has not been installed.";

--A.3.5 dateOfCreation

dateOfCreation **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.GlobalTime;

MATCHES FOR **EQUALITY**;

BEHAVIOUR dateOfCreationBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) dateOfCreation(5)};

dateOfCreationBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute indicates the time of creation of the managed object. The syntax of this attribute is of ASN.1 GeneralizedTime type.";

--A.3.6 dateOfLastModification

dateOfLastModification **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.Date;

MATCHES FOR **EQUALITY**;

BEHAVIOUR dateOfLastModificationBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) dateOfLastModification(6)};

dateOfLastModificationBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute identifies the time of the last, or most recent modification to the information represented by the softwareUnit object instance. Valid values for this attribute are ASN.1 GeneralizedTime or NULL if the information has not been modified.";

--A.3.7 fileLocation

fileLocation **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.FileLocation;

MATCHES FOR **EQUALITY**;

BEHAVIOUR fileLocationBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) fileLocation(7)};

fileLocationBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute specifies the full address(es) (either logical or physical) of the softwareUnit object. The format of the address is implementation-dependent, conforming to the file-addressing conventions of the particular managed system in question. A null value of this attribute indicates that the information to which the softwareUnit managed object applies has not yet been installed in the managed system.";

--A.3.8 fileSize

fileSize ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.InformationSize;

MATCHES FOR EQUALITY;

BEHAVIOUR fileSizeBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) fileSize(8)};

fileSizeBehaviour BEHAVIOUR

DEFINED AS

"This attribute indicates the size of the softwareUnit managed object.";

--A.3.9 fileType

fileType ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.FileType;

MATCHES FOR EQUALITY;

BEHAVIOUR fileTypeBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) fileType(9)};

fileTypeBehaviour BEHAVIOUR

DEFINED AS

"This attribute indicates the type of the softwareUnit. Possible softwareUnit types are unstructured binary file (e.g. executable file), unstructured text file (e.g. non-executable file), and block special file, etc.";

--A.3.10 futureAutoBackupDestination

futureAutoBackupDestination ATTRIBUTE

WITH ATTRIBUTE SYNTAX BackupRestoreASN1Module.BackupDestination;

MATCHES FOR EQUALITY;

BEHAVIOUR futureAutoBackupDestinationBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) futureAutoBackupDestination(10)};

futureAutoBackupDestinationBehaviour BEHAVIOUR

DEFINED AS

"This attribute specifies the destination to which the information represented by this object instance will be backup. The backup criteria is defined in the futureAutoBackupTriggerThreshold attribute of the object instance. The destination can be another object instance of the same object class exists in the same local managed system, a remote open system (by using a particular file transfer protocol, e.g. FTAM), or the managing system (by in-line using the autoBackupReport notification).";

--A.3.11 futureAutoBackupTriggerThreshold

futureAutoBackupTriggerThreshold ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.Integer;

MATCHES FOR EQUALITY, ORDERING;

BEHAVIOUR futureAutoBackupTriggerThresholdBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) futureAutoBackupTriggerThreshold(11)};

futureAutoBackupTriggerThresholdBehaviour BEHAVIOUR

DEFINED AS

ISO/CEI 10164-18 : 1997 (S)

"This attribute specifies the threshold that will trigger an automatic backup for the information represented by the object instance. The threshold is defined as the number of time that the information has been modified. Once the information has been modified for that number of times, an automatic backup will be performed. The backup destination is specified in the attribute futureAutomaticBackupDestination. Such backups are carried out in addition to other scheduled periodic backup. At the completion of the automatic backup, an autoBackupReport notification shall be emitted from the object.";

--A.3.12 futureAutoRestoreAllowed

futureAutoRestoreAllowed **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.Boolean;

MATCHES FOR **EQUALITY**;

BEHAVIOUR futureAutoRestoreAllowedBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) futureAutoRestoreAllowed(12)};

futureAutoRestoreAllowedBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute specifies whether automatic restore of the information represented by this manage object instance is allowed. The syntax of this attribute is of ASN.1 BOOLEAN type with the value TRUE meaning allowed, and FALSE meaning not allowed. The criteria that triggers automatic information restore is system specific.";

--A.3.13 futureAutoRestoreSource

futureAutoRestoreSource **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.AutoRestoreSource;

MATCHES FOR **EQUALITY**;

BEHAVIOUR futureAutoRestoreSourceBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) futureAutoRestoreSource(13)};

futureAutoRestoreSourceBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute specifies the source of the information to be restored to the information represented by the managed object instance. The source is either a local managed object or a remote system. The criteria of triggering an automatic restore of information is system specific.";

--A.3.14 identityOfCreator

identityOfCreator **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.Identity;

MATCHES FOR **EQUALITY**;

BEHAVIOUR identityOfCreatorBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) identityOfCreator(14)};

identityOfCreatorBehaviour **BEHAVIOUR**

DEFINED AS

"This attribute identifies the entity that creates the managed object.";

--A.3.15 identityOfLastModifier

identityOfLastModifier **ATTRIBUTE**

WITH ATTRIBUTE SYNTAX SWMF.Identity;

MATCHES FOR **EQUALITY**;

BEHAVIOUR identityOfLastModifierBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) identityOfLastModifier(15)};

identityOfLastModifierBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the last, or most recent, modifier of the information represented by the softwareUnit object instance.";

--A.3.16 lastBackupDestination

lastBackupDestination ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.LastBackupDestination;

MATCHES FOR EQUALITY;

BEHAVIOUR lastBackupDestinationBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) lastBackupDestination(16)};

lastBackupDestinationBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the destination, if exists, to which the information represented by the managed object is backed up.";

--A.3.17 lastBackupTime

lastBackupTime ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.Date;

MATCHES FOR EQUALITY;

BEHAVIOUR lastBackupTimeBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) lastBackupTime(17)};

lastBackupTimeBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the time of the last backup on the information represented by the managed object instance. Valid values for this attribute are ASN.1 GeneralizedTime or NULL (if no backup has been performed on the information).";

--A.3.18 lastRestoreSource

lastRestoreSource ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.LastRestoreSource;

MATCHES FOR EQUALITY;

BEHAVIOUR lastRestoreSourceBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) lastRestoreSource(18)};

lastRestoreSourceBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the source, if exists, from which the information represented by the managed object is restored.";

--A.3.19 lastRestoreTime

lastRestoreTime ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.Date;

MATCHES FOR EQUALITY;

BEHAVIOUR lastRestoreTimeBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) lastRestoreTime(19)};

lastRestoreTimeBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies the time of the last restore on the information represented by the managed object instance. Valid values for this attribute are ASN.1 GeneralizedTime or NULL (if no restore has been performed on the information).";

--A.3.20 noteField

noteField ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.NoteField;

MATCHES FOR EQUALITY;

BEHAVIOUR noteFieldBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) noteField(20)};

noteFieldBehaviour BEHAVIOUR

DEFINED AS

"This attribute contains any information or comments associated with the managed object, including any specific installation instructions, startup parameters and values, information necessary to activate features of the managed object, etc.";

--A.3.21 softwareDistributorId

softwareDistributorId ATTRIBUTE

WITH ATTRIBUTE SYNTAX SWMF.SimpleNameType;

MATCHES FOR EQUALITY;

BEHAVIOUR softwareDistributorIdBehaviour;

REGISTERED AS {SWMF.softwareManagement attribute(7) softwareDistributorId(21)};

softwareDistributorIdBehaviour BEHAVIOUR

DEFINED AS

"This attribute identifies an instance of the software distributor managed object class.";

--A.4 Name Bindings

--A.4.1 softwareDistributor-subsystem

softwareDistributor-subsystem NAME BINDING

SUBORDINATE OBJECT CLASS softwareDistributor AND SUBCLASSES;

NAMED BY SUPERIOR OBJECT CLASS

"Rec. X.723 | ISO/IEC 10165-5":subsystem AND SUBCLASSES;

WITH ATTRIBUTE softwareDistributorId;

CREATE WITH-AUTOMATIC-INSTANCE-NAMING;

DELETE DELETES-CONTAINED-OBJECTS;

REGISTERED AS {SWMF.softwareManagement nameBinding(6) softwareDistributor-subsystem(1)};

--A.4.2 softwareDistributor-system

softwareDistributor-system NAME BINDING

SUBORDINATE OBJECT CLASS softwareDistributor AND SUBCLASSES;

NAMED BY SUPERIOR OBJECT CLASS

"Rec. X.721 | ISO/IEC 10165-2:1992":system AND SUBCLASSES;

WITH ATTRIBUTE softwareDistributorId;

CREATE WITH-AUTOMATIC-INSTANCE-NAMING;

DELETE DELETES-CONTAINED-OBJECTS;

REGISTERED AS {SWMF.softwareManagement nameBinding(6) softwareDistributor-system(2)};

--A.4.3 softwareUnit-subsystem

softwareUnit-subsystem NAME BINDING

SUBORDINATE OBJECT CLASS softwareUnit AND SUBCLASSES;

NAMED BY SUPERIOR OBJECT CLASS

"Rec. X.723 | ISO/IEC 10165-5":subsystem AND SUBCLASSES;

WITH ATTRIBUTE "Rec. M.3100:1995":softwareId;

CREATE WITH-AUTOMATIC-INSTANCE-NAMING;

DELETE DELETES-CONTAINED-OBJECTS;

REGISTERED AS {SWMF.softwareManagement nameBinding(6) softwareUnit-subsystem(3)};

--A.4.4 softwareUnit-system

softwareUnit-system NAME BINDING

SUBORDINATE OBJECT CLASS softwareUnit AND SUBCLASSES;

NAMED BY SUPERIOR OBJECT CLASS

"Rec. X.721 | ISO/IEC 10165-2:1992":system AND SUBCLASSES;

WITH ATTRIBUTE "Rec. M.3100:1995":softwareId;

CREATE WITH-AUTOMATIC-INSTANCE-NAMING;

DELETE DELETES-CONTAINED-OBJECTS;

REGISTERED AS {SWMF.softwareManagement nameBinding(6) softwareUnit-system(4)};

--A.5 Actions

--A.5.1 deliver

deliver ACTION

BEHAVIOUR deliverBehaviour;

MODE CONFIRMED;

PARAMETERS softwareProcessingFailureParameter;

WITH INFORMATION SYNTAX SWMF.DeliverInfo;

REGISTERED AS {SWMF.softwareManagement action(9) deliver(1)};

deliverBehaviour BEHAVIOUR

DEFINED AS

"The deliver service is used by a managing system to request distribution of a software or a set of software. A number of software units may be delivered together to a managed system. If the Deliver Info does not indicate a target destination, the system uses local means to determine the target destination. This service used the CMIS M-ACTION service and procedures defined in ISO/IEC 9595. The Action Type parameter shall indicate deliver.";

--A.5.2 executeProgram

executeProgram ACTION

BEHAVIOUR executeProgramBehaviour;

MODE CONFIRMED;

ISO/CEI 10164-18 : 1997 (S)

PARAMETERS softwareProcessingFailureParameter;

WITH INFORMATION SYNTAX SWMF.ExecuteProgramInfo;

WITH REPLY SYNTAX SWMF.ExecuteProgramReply;

REGISTERED AS {SWMF.softwareManagement action(9) executeProgram(2)};

executeProgramBehaviour BEHAVIOUR

DEFINED AS

"The executeProgram service is used by a managing system to request an execution of the program represented by the executableSoftware object instance. A success request will be confirmed with information including the process Id, process owner, and the starting time of the execution. When the executableSoftware is in the busy usageState, any executeProgram request will be denied, until the usageState changes to active or idle. This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC IS 9595. The Action Type parameter shall indicate executeProgram.";

--A.5.3 install

install ACTION

BEHAVIOUR installBehaviour;

MODE CONFIRMED;

PARAMETERS

softwareProcessingFailureParameter;

WITH INFORMATION SYNTAX

SWMF.InstallInfo;

REGISTERED AS {SWMF.softwareManagement action(9) install(3)};

installBehaviour BEHAVIOUR

DEFINED AS

"The install service is used by a managing system to instruct a managed system to install a softwareUnit object instance. If applicable, the install service will effect the automatic update the value of the appliedPatches attribute. This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC 9595. The Action Type parameter shall indicate install.";

--A.5.4 revert

revert ACTION

BEHAVIOUR revertBehaviour;

MODE CONFIRMED;

PARAMETERS

softwareProcessingFailureParameter;

WITH INFORMATION SYNTAX SWMF.RevertArgument;

WITH REPLY SYNTAX SWMF.RevertReply;

REGISTERED AS {SWMF.softwareManagement action(9) revert(4)};

revertBehaviour BEHAVIOUR

DEFINED AS

"The revert service is used by a managing system (e.g. OS) to instruct a managed system to revert an applied patch or set of patches of the software represented by the software unit managed object or the executable software managed object. The revert service will automatically return the value of the appliedPatches attribute of the executableSoftware object instance to which the service is directed.

This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC 9595. The Action Type parameter shall indicate revert patches. The Action Information parameter shall identify the patch or patches to be reverted of the executableSoftware object instance. Each patch identifier is a choice of a system specific identifier (Printable String) or a software unit object instance (Object Instance).";

--A.5.5 terminateValidation**terminateValidation ACTION****BEHAVIOUR terminateValidationBehaviour;****MODE CONFIRMED;****PARAMETERS****softwareProcessingFailureParameter;****WITH INFORMATION SYNTAX SWMF.TerminateValidationArgument;****WITH REPLY SYNTAX SWMF.TerminateValidationReply;****REGISTERED AS {SWMF.softwareManagement action(9) terminateValidation(5)};****terminateValidationBehaviour BEHAVIOUR****DEFINED AS**

"The terminateValidation service is used by a managing system to terminate a currently active (running) validation process on one or more objects. Two termination modes are provided for terminating a validation, namely, cancel-mode and truncated-mode. For the truncate-mode, the validation will be terminated and the result of the partially completed validation will be returned in the Action Reply parameter of the terminateValidation M-ACTION. For the cancel-mode, the validation will be terminated and the result of the partially completed validation will be discarded. This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC 9595. The Action Type parameter shall indicate terminateValidation. The Action Information parameter shall indicate the mode of the termination, i.e. cancel-mode or truncate-mode. The Action Reply parameter shall indicate the result of the termination, i.e. terminated (for cancel-mode), terminated with partial result (for truncate-mode), or no active validation to be terminated."

--A.5.6 validate**validate ACTION****BEHAVIOUR validateBehaviour;****MODE CONFIRMED;****PARAMETERS****softwareProcessingFailureParameter;****WITH INFORMATION SYNTAX SWMF.ValidateInfo;****WITH REPLY SYNTAX SWMF.ValidateReply;****REGISTERED AS {SWMF.softwareManagement action(9) validate(6)};****validateBehaviour BEHAVIOUR****DEFINED AS**

"The validate service is used by a managing system to request performing an validation on the information represented by the softwareUnit object instance. This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC 9595. The Action Type parameter shall indicate validate. If the validation process is terminated (by using the terminateValidation M-ACTION) before the normal completion of the validation, then the value 'terminated' (with ASN.1 type NULL) shall be returned in the Action Reply of the validate M-ACTION."

--A.6 Notifications**--A.6.1 autoBackupReport****autoBackupReport NOTIFICATION****BEHAVIOUR autoBackupReportBehaviour;****WITH INFORMATION SYNTAX SWMF.AutoBackupReportInfo;****REGISTERED AS {SWMF.softwareManagement notification(10) autoBackupReport(1)};****autoBackupReportBehaviour BEHAVIOUR****DEFINED AS**

ISO/CEI 10164-18 : 1997 (S)

"The autoBackupReport notification is emitted to report an automatic backup of the information represented by this object. The automatic backup criteria and backup destination are specified in the futureAutoBackupTriggerThreshold and futureAutoBackupDestination attributes of the object respectively. The backup destination may be local (i.e. backup to another object of the same class within the local managed system), the managing system, or off-line to a remote system by using a particular file transfer protocol (e.g. FTAM). For local and off-line backup, the result of the backup, i.e. success or failure, will be reported in this notification. For backup to the managing system, the backup information will be in-line included in the notification as a component of the Event Information parameter.";

--A.6.2 autoRestoreReport

autoRestoreReport NOTIFICATION

BEHAVIOUR autoRestoreReportBehaviour;

WITH INFORMATION SYNTAX SWMF.AutoRestoreReportInfo;

REGISTERED AS {SWMF.softwareManagement notification(10) autoRestoreReport(2)};

autoRestoreReportBehaviour BEHAVIOUR

DEFINED AS

"The autoRestoreReport notification is emitted from the managed object when an automatic restore is occurred on the information represented by this object instance. The criteria that triggers the automatic restore is system specific. The source of the restored information and the result of the restore operation (i.e. either success or failure) shall be reported in the notification.";

--A.6.3 deliverResultNotification

deliverResultNotification NOTIFICATION

BEHAVIOUR deliverResultNotificationBehaviour;

WITH INFORMATION SYNTAX SWMF.DeliverResultInfo;

REGISTERED AS {SWMF.softwareManagement notification(10) deliverResultNotification(3)};

deliverResultNotificationBehaviour BEHAVIOUR

DEFINED AS

"This notification type is used to report deliver results and is emitted from the managed object when the delivery is completed. The deliver result parameter may take one of the following values: pass, fail or unknown. This service uses the CMIS M-EVENT-REPORT service and procedures defined in ISO/IEC 9595. The Notification Type parameter shall indicate deliverResult.";

--A.7 Parameters

--A.7.1 alarmEffectOnServiceParameter

alarmEffectOnServiceParameter PARAMETER

CONTEXT EVENT-INFO;

WITH SYNTAX

SWMF.AlarmEffectOnServiceParameter;

BEHAVIOUR alarmEffectOnServiceParameterBehaviour;

REGISTERED AS {SWMF.softwareManagement parameter(5) alarmEffectOnServiceParameter(1)};

alarmEffectOnServiceParameterBehaviour BEHAVIOUR

DEFINED AS

"The alarmEffectOnServiceParameter is a parameter to be included in the ManagementExtension parameter of the AdditionalInformation parameter of the AlarmInfo parameter in an Alarm Reporting service. This parameter indicates whether the service is affected by the alarm. The ManagementExtension is of the form (see Rec. X.721 | ISO/IEC 10165-2):

ManagementExtension ::= SEQUENCE {

identifier OBJECT IDENTIFIER,

significance [1] BOOLEAN DEFAULT FALSE,

information [2] ANY DEFINED BY identifier}

The OBJECT IDENTIFIER carried in identifier shall be the value under which this parameter definition is registered. The type carried in information shall be the type identified by the WITH SYNTAX construct of this parameter definition.";

--A.7.2 softwareProcessingFailureParameter

softwareProcessingFailureParameter PARAMETER

CONTEXT SPECIFIC-ERROR;

WITH SYNTAX SWMF.SoftwareProcessingFailure;

BEHAVIOUR softwareProcessingFailureParameterBehaviour;

REGISTERED AS {SWMF.softwareManagement parameter(5) softwareProcessingFailureParameter(2)};

softwareProcessingFailureParameterBehaviour BEHAVIOUR

DEFINED AS

"The softwareProcessingFailureParameter defines the data syntax to be return in a CMIP ProcessingFailure error reply to a M-ACTION if a request for an operation on software is denied due to errors other than those already defined in CMIP-1. The attributes in the returned data syntax include the state attributes of the emitting object. Any other applicable attributes may be included but these are a local matter.";

--A.8 Supporting Productions

SWMF {joint-iso-ccitt ms(9) function(2) part18(18) asn1Module(2) 0}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- EXPORTS everything

IMPORTS

AE-title

FROM

ACSE-1 {joint-iso-itu-t(2) association-control(2) modules(0) apdus(0) version1(1)}

Attribute, AttributeId, ObjectInstance

FROM

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

Destination, ManagementExtension, SimpleNameType

FROM

Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1};

--object identifier values--

softwareManagement OBJECT IDENTIFIER ::= { joint-iso-ccitt ms(9) function(2) part18(18)}

-- supporting productions

AlarmEffectOnServiceParameter ::= BOOLEAN -- TRUE = affected --

AppliedPatches ::= SEQUENCE OF Patch

AutoBackupReportInfo ::= SEQUENCE {
 backupResult [0] BackupResult,
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

AutoRestoreReportInfo ::= SEQUENCE {
 source AutoRestoreSource,
 success BOOLEAN, -- TRUE for success
 additionalInfo SET OF ManagementExtension OPTIONAL }

AutoRestoreSource ::= CHOICE { localObject ObjectInstance,
 remoteSystem GraphicString -- off-line from remote system --}

BackupResult ::= CHOICE {

```

    inLine [0] CHOICE {
        success BIT STRING,
        fail-pduSizeLimitation [3] NULL,
        fail-securityLicensing [4] NULL,
        fail-unknown [5] NULL}

    local [1] SEQUENCE {
        destination ObjectInstance, -- in the managed system --
        success BOOLEAN -- TRUE for success --
    },

    offLine [2] SEQUENCE {
        destination GraphicString, --the remote system
        result CHOICE {
            success [6] NULL,
            fail-securityLicensing [7] NULL,
            fail-unknown [8] NULL}
        }}

```

Boolean ::= BOOLEAN

Checksum ::= BIT STRING

Date ::= CHOICE {
 time GeneralizedTime ,
 noSuchInformation NULL}

DeliverId ::= CHOICE {
 globalValue OBJECT IDENTIFIER,
 localValue INTEGER}

DeliverInfo ::= SEQUENCE {
 deliverId [0] DeliverId OPTIONAL,
 -- each delivery may have an id to correlate action and event --
 targetSoftware SET OF DistributedSoftware,
 targetSystem [1] Destination OPTIONAL,
 transferInfo [2] TransferInfo OPTIONAL,
 additionalInfo [3] SET OF ManagementExtension OPTIONAL }

DeliverResult ::= INTEGER {
 pass (0),
 communicationError (1),
 equipmentError (2),
 qosError (3),
 accessDenied (4),
 notFound (5),
 insufficientSpace (6),
 alreadyDelivered (7),
 inProgress (8),
 unknown (9) }

DeliverResultInfo ::= SEQUENCE {
 deliverId [0] DeliverId OPTIONAL,
 deliverResult [1] DeliverResult,
 additionalInfo SET OF ManagementExtension OPTIONAL}

DistributedSoftware ::= CHOICE {
 distributedSoftwareId GraphicString,
 distributedSoftwarePointer ObjectInstance }

ExecuteProgramInfo ::= SET OF ManagementExtension

ExecuteProgramReply ::= SEQUENCE {
 processId INTEGER,
 processOwner Identity,
 startTime GeneralizedTime,
 additionalInfo SET OF ManagementExtension OPTIONAL }

FileLocation ::= SET OF GraphicString -- Empty set means file is not installed --

FileType ::= INTEGER {
 unstructuredText (0), -- *FTAM-1*
 unstructuredBinary (1), -- *FTAM-3*
 blockSpecial (2)}

GlobalTime ::= GeneralizedTime

Identity ::= GraphicString

InformationSize ::= CHOICE {
 numberOfBits [0] INTEGER,
 numberOfBytes [1] INTEGER}

Integer ::= INTEGER

InstallInfo ::= SEQUENCE {
 targetSoftware SET OF DistributedSoftware,
 installInfo SET OF ManagementExtension }

LastBackupDestination ::= CHOICE {
 notBackedUp NULL,
 localObject ObjectInstance,
 managingSystem AE-title,
 remoteSystem GraphicString}

LastRestoreSource ::= CHOICE {
 notRestored NULL,
 localObject ObjectInstance,
 managingSystem AE-title,
 remoteSystem GraphicString}

NoteField ::= GraphicString

Null ::= NULL

Patch ::= CHOICE {
 patchId GraphicString, -- *system specific identifier* --
 patchPointer ObjectInstance } -- *of Software Unit object class* --

RevertArgument ::= SEQUENCE {
 revertInfo [0] RevertInfo,
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

RevertInfo ::= SEQUENCE OF CHOICE {
 patchId GraphicString, -- *system specific identifier* --
 patchPointer ObjectInstance } -- *Executable Software object class* --

RevertReply ::= SEQUENCE {
 revertedPatches [0] AppliedPatches,
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

SoftwareProcessingFailure ::= SET OF Attribute -- *state attributes* --

TerminateValidationArgument ::= SEQUENCE {
 terminateValidationInfo [0] TerminateValidationInfo,
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

TerminateValidationInfo ::= ENUMERATED {
 cancel (0), -- *discard the result of the partial audit* --
 truncate (1) } -- *report the result of the partially completed audit* --

TerminateValidationReply ::= CHOICE {
 noOutStandingValidation [0] NULL,
 validationCancelled [1] NULL,
 resultOfPartialValidation [2] ValidateReply}

TransferInfo ::= SEQUENCE {
 transferProtocol TransferProtocol,
 protocolSpecificInfo ANY DEFINED BY transferProtocol OPTIONAL}

TransferProtocol ::= OBJECT IDENTIFIER

```

ValidateInfo ::= CHOICE {
    instanceDefaultValidationType [0] NULL, -- local matter --
    registeredValidationType[1] OBJECT IDENTIFIER,
    systemSpecificValidationType [2] SET OF ManagementExtension }

ValidateReply ::= CHOICE {
    validationTerminated [0] NULL,
    passValidation [1] NULL,
    passValidationWithResult [2] SET OF ManagementExtension,
    failValidation [3] NULL,
    failValidationWithResult [4] SET OF ManagementExtension }

END -- end of SWMF supporting productions --

```

--A.9 Backup and Restore Actions--

-- The backup and restore actions are generic actions --
 -- and intended for reuse outside the scope of this --
 -- Recommendation | International Standard as appropriate --
 -- for objects in addition to software objects. --

--A.9.1 backup

```

backup ACTION
    BEHAVIOUR backupBehaviour;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX BackupRestoreASN1Module.BackupArgument;
    WITH REPLY SYNTAX BackupRestoreASN1Module.BackupReply;

```

REGISTERED AS {SWMF.softwareManagement action(9) backup(7)};

backupBehaviour BEHAVIOUR

DEFINED AS

"The backup service is used by a managing system to request performing a backup on the information represented by the target object instance. This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC 9595.

The Action Type parameter shall indicate backup.

The Action Information parameter shall indicate the destination to which the information will be backed up. Possible destinations are:

- A local managed object of the same class as the one this action is applied to. In this case, the backup operation will be performed internally in the managed system.
- The managing system from which this action is sent. In this case, a copy of the backup information will be sent in-line in the Action Reply.
- A remote system. In this case, the backup information will be transferred off-line to the remote system by using a locally chosen file transfer protocol.

The Action Reply parameter shall indicate the result of the backup. For local or off-line backup, a NULL value indicates the backup is success. For in-line backup, the backup information will be included in the Action Reply parameter.";

--A.9.2 restore--

```

restore ACTION
    BEHAVIOUR restoreBehaviour;
    MODE CONFIRMED;
    WITH INFORMATION SYNTAX BackupRestoreASN1Module.RestoreArgument;

```

REGISTERED AS {SWMF.softwareManagement action(9) restore(8)};

restoreBehaviour BEHAVIOUR

DEFINED AS

"The restore service is used by a managing system to request performing a restore on the information represented by the target object instance. This service uses the CMIS M-ACTION service and procedures defined in ISO/IEC 9595.

The Action Type parameter shall indicate restore.

The Action Information parameter shall indicate the source from which the information will be restored. Possible sources are:

- A local managed object of the same class as the one this action is applied to. In this case, the restore operation will be performed internally in the managed system.
- The managing system from which this action is sent. In this case, a copy of the restore information will be sent in-line in the Action Information parameter.
- A remote system. In this case, the restore information will be transferred off-line from the remote system by using a locally chosen file transfer protocol.

The Action Reply parameter shall indicate the result of the restore. A NULL value indicates the restore is success.";

--A.9.3 Backup Restore Supporting Productions--

BackupRestoreASN1Module {joint-iso-ccitt ms(9) function(2) part18(18) asn1Module(2) 1}

DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- EXPORTS everything

IMPORTS

ObjectInstance

FROM

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

ManagementExtension

FROM

Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module(2) 1};

--supporting productions--

BackupArgument ::= SEQUENCE {
 backupDestination [0] BackupDestination,
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

BackupDestination ::= CHOICE {
 localObject ObjectInstance,
 inLine NULL, -- in-line in the notification in additionalInfo --
 offLine GraphicString -- remote system by, e.g. FTAM --}

BackupReply ::= SEQUENCE {
 reply [0] CHOICE {
 success NULL, -- for local or off-line backup
 inLine BIT STRING },
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

RestoreArgument ::= SEQUENCE {
 restoreSource [0] RestoreSource,
 additionalInfo [1] SET OF ManagementExtension OPTIONAL }

RestoreSource ::= CHOICE {
 localObject ObjectInstance,
 inLine BIT STRING ,
 offLine GraphicString -- remote system via some other transfer protocol, e.g. FTAM --
 }

END -- of BackupRestoreASN1Module supporting productions --

Formulario de MCS

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

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 MCS

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 ITU-T Rec. X.291 and ISO/IEC 9646-2 and ITU-T Rec. X.296 and 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 "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 selectable options among a set of status values. Support of at least one of the choices (from the items with the same value of N) is required.

For all annexes of this Recommendation | International Standard, the following common notations, defined in ITU-T Rec. X.291 and ISO/IEC 9646-2 and ITU-T Rec. X.296 and 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.1.4 Table format

Some of the tables in this Recommendation | International Standard have been split because the information is too wide to fit on the page. Where this occurs, the index number of the first block of columns are the index numbers of the corresponding rows of the remaining blocks of columns. A complete table reconstructed from the constituent parts should have the following layout:

Index	First block of columns	Second block of columns	Etc.
-------	------------------------	-------------------------	------

3) Comunicado sobre derechos de autor del formulario de MCS

Los usuarios de esta Recomendación | Norma Internacional pueden reproducir libremente el formulario de MCS de este anexo a fin de que pueda ser utilizado para los fines previstos, y pueden además publicar el MCS cumplimentado. En la Rec. UIT-T X.724 | ISO/CEI 10165-6 se especifican las instrucciones para rellenar el formulario de MCS.

In this Recommendation | International Standard the constituent parts of the table appear consecutively, starting with the first block of columns.

When a table with subrows is too wide to fit on a page, the continuation table(s) have been constructed with index numbers identical to the index numbers in the corresponding rows of the first table, and with subindex numbers corresponding to the subrows within each indexed row. For example, if Table X.1 has 2 rows and the continuation of Table X.1 has 2 subrows for each row, the tables are presented as follows:

Table X.1 – Title

					Support		
Index	A	B	C	D	E	F	G
1	a	b	–				
2	a	b	–				

Table X.1 (continued) – Title

Index	Subindex	H	I	J	K	L
1	1.1	h	i	j		
	1.2	h	i	j		
2	2.1	h	i	j		
	2.1	h	i	j		

A complete table reconstructed from the constituent parts should have the following layout:

								Support					
Index	A	B	C	D	E	F	G	Subindex	H	I	J	K	L
1	a	b	–					1.1	h	i	j		
								1.2	h	i	j		
2	a	b	–					2.1	h	i	j		
								2.2	h	i	j		

References made to cells within tables shall be interpreted as references within reconstructed tables. In the example, above, the reference X.1/1d corresponds with the blank cell in column G for row with Index 1, and X.1/1.2b corresponds with the blank cell in column L for row with Subindex 1.2.

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

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 contents of the MCS or any referenced implementation conformance statement, in the box below.

B.3 Identification of the Recommendations | International Standards 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 Recommendations | International Standards which specify the management information to which conformance is claimed, in the box below.

Recommendations International Standards 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 Recommendations | International Standards, 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 the implementation shall state the capabilities and features supported and provide a 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 systems management functional units, in Table B.2.

Table B.2 – Systems management functional units

Index	Systems management functional unit name	Manager		Agent		Additional information
		Status	Support	Status	Support	
1	Software control functional unit	c1		c2		
2	Software deliver functional unit	c1		c2		
3	Backup restore functional unit	c1		c2		
c1: if B.1/1a then o else –.						
c2: if B.1/1a then o else –.						

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	Additional information
1	Operations on managed objects	o.2		
2	Backup action for software unit managed object	o.2		
3	Backup action for executable software managed object	o.2		
4	Deliver action for software distributor managed object	o.2		
5	Execute action for executable software managed object	o.2		
6	Install action for software unit managed object	o.2		
7	Install action for executable software managed object	o.2		
8	Restore action for software unit managed object	o.2		
9	Restore action for executable software managed object	o.2		
10	Revert action for software unit managed object	o.2		
11	Revert action for executable software managed object	o.2		
12	Terminate validation for software unit managed object	o.2		
13	Terminate validation for executable software managed object	o.2		
14	Validate action for software unit managed object	o.2		
15	Validate action for executable software managed object	o.2		
16	Auto backup report notification for software objects	o.2		
17	Auto restore report notification for software objects	o.2		
18	Deliver result notification for software distributor	o.2		
19	State change notification for software objects	o.2		

Table B.3 (concluded) – Manager role minimum conformance requirement

Index	Item	Status	Support	Additional information
20	Object creation notification for software objects	o.2		
21	Object deletion notification for software objects	o.2		
22	Attribute value change notification for software objects	o.2		
23	Processing error alarm notification for software objects	o.2		
c3: if B.2/1a or B.2/2a then m else (if B.1/1a then o.2 else –).				

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	Additional information
1	Software managed object class or executable Software managed object class	c5		
2	Software distributor managed object class	c6		
c5: if B.1/2a and B.2/1a then m else –.				
c6: if B.1/2a and B.2/2a then m else –.				

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		

The supplier of the implementation shall provide information on claims of conformance to any of the Recommendations | International Standards summarized in the following tables. 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, B.8 and B.9, 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	CCITT Rec. X.730 ISO/IEC 10164-1	Tables B.2 to B.8	objectCreation, objectDeletion and attributeValueChange	–	m			
2	CCITT Rec. X.730 ISO/IEC 10164-1	Annex E all tables	SM application context	OBJECT IDENTIFIER	m			
3	CCITT Rec. X.731 ISO/IEC 10164-2	Tables B.2 to B.4	stateChange	–	m			
4	CCITT Rec. X.733 ISO/IEC 10164-4	Tables B.2 to B.12	processingErrorAlarm	–	m			
5	CCITT Rec. X.735 ISO/IEC 10164-6	Annex B all tables	–	–	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.744 ISO/IEC 10164-18	D.4	softwareUnit		c8			
2	ITU-T Rec. X.744 ISO/IEC 10164-18	D.5	executableSoftware		c8			
3	ITU-T Rec. X.744 ISO/IEC 10164-18	D.6	softwareDistributor		c9			
c8: if B.4/1a then o.3 else –. c9: if B.4/2a then m else –.								

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	ITU-T Rec. X.744 ISO/IEC 10164-18	Annex E	softwareDistributor-subsystem	–	o			
2	ITU-T Rec. X.744 ISO/IEC 10164-18	Annex E	softwareDistributor-system	–	o			
3	ITU-T Rec. X.744 ISO/IEC 10164-18	Annex E	softwareUnit-subsystem	AND SUBCLASSES	o			
4	ITU-T Rec. X.744 ISO/IEC 10164-18	Annex E	softwareUnit-system	AND SUBCLASSES	o			

Table B.9 – MICS support summary

Index	Identification of the document that includes the MICS proforma	Table numbers of MICS proforma	Description	Constraints and values	Status	Support	Table numbers of MICS	Additional information
1	ITU-T Rec. X.744 ISO/IEC 10164-18	Tables C.1 and C.2	management operations		c10			
2	ITU-T Rec. X.744 ISO/IEC 10164-18	Table C.3	notifications		c11			
c10: if B.3/1a or B.3/2a or B.3/3a or B.3/4a or B.3/5a or B.3/6a or B.3/7a or B.3/8a or B.3/9a or B.3/10a or B.3/11a or B.3/12a or B.3/13a or B.3/14a or B.3/15a the m else –. c11: if B.3/16a or B.3/17a or B.3/18a or B.3/19a or B.3/20a or B.3/21a or B.3/22a or B.3/23a then m else –.								

Formular de MICS

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

C.1 Introduction

The purpose of this MICS proforma is to provide a mechanism for a supplier of an implementation which claims conformance, in the manager role, to management information specified in this Recommendation | International Standard, to provide conformance information in a standard form.

C.2 Instructions for completing the MICS proforma to produce an MICS

The MICS proforma contained in this annex is comprised of information in tabular form, in accordance with ITU-T Rec. X.724 | ISO/IEC 10165-6. In addition to the general guidance given in ITU-T Rec. X.724 | ISO/IEC 10165-6, the Additional information column shall be used to identify the object classes for which the management operations are supported. The supplier of the implementation shall state which items are supported in tables below and if necessary, provide additional information.

C.3 Symbols, abbreviations and terms

The following abbreviation is used throughout the MICS proforma:

dmi-att **joint-iso-ccitt ms(9) smi(3) part2(2) attribute(7)**

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

C.4 Statement of conformance to the management information**C.4.1 Attributes**

The specifier of a manager role implementation that claims to support management operations on the attributes specified in this Recommendation | International Standard shall import a copy of Table C.1 and complete it.

Table C.1 – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
1	administrativeState	{dmi-att 31}	ENUMERATED	o.1		o.1		o.1		-		-		o.1		
2	affectedObjectList	{m3100-att 2}	SET OF CHOICE	o.1		o.1		o.1		o.1		o.1		o.1		
3	alarmStatus	{m3100-att 6}	ENUMERATED	o.1		o.1		o.1		-		-		o.1		
4	allomorphs	{dmi-att 50}	SET OF CHOICE	o.1		o.1		-		-		-		-		
5	appliedPatches	{swmf-att 1}	SEQUENCE OF CHOICE	o.1		o.1		o.1		-		-		o.1		

⁴⁾ **Comunicado sobre derechos de autor del formulario de MICS**

Los usuarios de esta Recomendación | Norma Internacional pueden reproducir libremente el formulario de MICS de este anexo a fin de que pueda ser utilizado para los fines previstos, y pueden además publicar el MICS cumplimentado.

Table C.1 (continued) – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
6	availabilityStatus	{dmi-att 33}	SET OF INTEGER	o.1		o.1		o.1		o.1		o.1		o.1		
7	checksum	{swmf-att 2}	BIT STRING	o.1		o.1		o.1		-		-		o.1		
8	currentProblemList	{m3100-att 17}	SET OF SEQUENCE	o.1		o.1		o.1		o.1		o.1		o.1		
9	dateDelivered	{swmf-att 3}	CHOICE	o.1		o.1		o.1		-		-		o.1		
10	dateInstalled	{swmf-att 4}	CHOICE	o.1		o.1		o.1		-		-		o.1		
11	dateOfCreation	{swmf-att 5}	GeneralizedTime	o.1		o.1		o.1		-		-		o.1		
12	dateOfLastModification	{swmf-att 6}	CHOICE	o.1		o.1		o.1		-		-		o.1		
13	fileLocation	{swmf-att 7}	CHOICE	o.1		o.1		o.1		-		-		o.1		
14	fileSize	{swmf-att 8}	CHOICE	o.1		o.1		o.1		-		-		o.1		
15	fileType	{swmf-att 9}		o.1		o.1		-		-		-		-		
16	futureAutoBackupDestination	{swmf-att 10}	BackupDestination	o.1		o.1		o.1		-		-		o.1		
17	futureAutoBackupTriggerThreshold	{swmf-att 11}	INTEGER	o.1		o.1		o.1		-		-		o.1		
18	futureAutoRestoreAllowed	{swmf-att 12}	BOOLEAN	o.1		o.1		o.1		-		-		o.1		
19	futureAutoRestoreSource	{swmf-att 13}	CHOICE	o.1		o.1		o.1		-		-		o.1		
20	identityOfCreator	{swmf-att 14}	PrintableString	o.1		o.1		o.1		-		-		o.1		
21	identityOfLastModifier	{swmf-att 15}	PrintableString	o.1		o.1		o.1		-		-		o.1		
22	lastBackupDestination	{swmf-att 16}	CHOICE	o.1		o.1		o.1		-		-		o.1		
23	lastBackupTime	{swmf-att 17}	CHOICE	o.1		o.1		o.1		-		-		o.1		
24	lastRestoreSource	{swmf-att 18}	CHOICE	o.1		o.1		o.1		-		-		o.1		
25	lastRestoreTime	{swmf-att 19}	CHOICE	o.1		o.1		o.1		-		-		o.1		
26	nameBinding	{dmi-att 63}	OBJECT IDENTIFIER	o.1		o.1		-		-		-		-		
27	noteField	{swmf-att 20}	PrintableString	o.1		o.1		o.1		-		-		o.1		
28	objectClass	{dmi-att 65}	CHOICE	o.1		o.1		-		-		-		-		
29	operationalState	{dmi-att 35}	ENUMERATED	-		o.1		-		-		-		-		
30	packages	{dmi-att 66}	SET OF OBJECT IDENTIFIER	o.1		o.1		-		-		-		-		
31	proceduralStatus	{dmi-att 36}	SET OF INTEGER	o.1		o.1		-		-		-		-		
32	softwareDistributorId	{swmf-att 21}	SimpleNameType	o.1		o.1		-		-		-		-		
33	softwareId	{m3100-att 38}	CHOICE	o.1		o.1		-		-		-		-		

Table C.1 (concluded) – Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Status	Support	Status	Support	Status	Support	Status	Support	Status	Support	Status	Support	
34	usageState	{dmi-att 39}	ENUMERATED	-		o.1		-		-		-		-		
35	userLabel	{m3100-att 50}	GraphicString	o.1		o.1		o.1		-		-		-		
36	vendorName	{m3100-att 51}	GraphicString	o.1		o.1		o.1		-		-		-		
37	version	{m3100-att 52}	GraphicString	o.1		o.1		o.1		-		-		-		

C.4.2 Actions

The specifier of a manager role implementation that claims to support actions specified in this Recommendation | International Standard shall import a copy of Table C.2 and complete it.

Table C.2 – Action support summary

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	backup	{swmf-act 7}	softwareProcessingFailureParameter	o.1			1.1	BackupArgument	Information Syntax SEQUENCE	c:m		
							1.1.1	restoreSource	CHOICE	c:m		
							1.1.1.1	localObject	Object Instance	c:o.1		
							1.1.1.2	inLine	BIT STRING	c:o.1		
							1.1.1.3	offLine	GraphicString	c:o.1		
							1.1.2	additionalInfo	SET OF ANY	c:o		
							1.2	BackupReply	Reply Syntax SEQUENCE	c:m		
							1.2.1	reply	CHOICE	c:m		
							1.2.1.1	success	NULL	c:o.2		
							1.2.1.2	inLine	BIT STRING	c:o.2		
							1.2.2	additionalInfo	SET OF ANY	c:o		
2	executeProgram	{swmf-act 2}	softwareProcessingFailureParameter	o.1			2.1	executeProgramInfo	Information Syntax SET OF SEQUENCE	c:m		
							2.1.2	significance	BOOLEAN	c:o		
							2.1.3	information	ANY DEFINED BY identifier	c:m		
							2.2	ExecuteProgramReply	Reply Syntax SEQUENCE	c:m		
							2.2.1	processId	INTEGER	c:m		
							2.2.2	processOwner	GraphicString	c:m		

Table C.2 (continued) – Action support summary

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							2.2.3	startTime	GeneralizedTime	c:m		
							2.2.4	additionalInfo	SET OF SEQUENCE	c:o		
							2.2.4.1	identifier	OBJECT IDENTIFIER	c:m		
							2.2.4.2	significance	BOOLEAN	c:o		
							2.2.4.3	information	ANY DEFINED BY identifier	c:m		
3	install	{swmf-act 3}	softwareProcessingFailureParameter	o.1			3.1	InstallInfo	Information Syntax SEQUENCE	c:m		
							3.1.1	targetSoftware	SET OF CHOICE	c:m		
							3.1.1.1	distributedSoftwareId	GraphicString	c:o.1		
							3.1.1.2	distributedSoftwarePointer	CHOICE	c:o.1		
							3.1.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.2		
							3.1.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							3.1.1.2.1.2	AttributeValue	ANY	c:m		
							3.1.1.2.2	nonSpecificForm	OCTET STRING	c:o.2		
							3.1.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.2		
							3.1.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							3.1.1.2.3.2	AttributeValue	ANY	c:m		
							3.1.2	installInfo	SET OF SEQUENCE	c:m		
							3.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							3.1.2.2	significance	BOOLEAN	c:o		
							3.1.2.3	information	ANY DEFINED BY identifier	c:m		
4	restore	{swmf-act 8}		o.1			4.1	RestoreArgument	Information Syntax SEQUENCE	c:m		
							4.1.1	restoreSource	CHOICE	c:m		
							4.1.1.1	localObject	Object Instance	c:o.1		
							4.1.1.2	inLine	BIT STRING	c:o.1		
							4.1.1.3	offLine	Printable String	c:o.1		

Table C.2 (continued) – Action support summary

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							4.1.2	additionalInfo	SET OF ANY	c:o		
5	revert	{swmf-act 4}	softwareProcessingFailureParameter	o.1			5.1	RevertArgument	Information Syntax SEQUENCE	c:m		
							5.1.1	revertInfo	SEQUENCE OF CHOICE	c:m		
							5.1.1.1	patchId	GraphicString	c:o.3		
							5.1.1.2	patchPointer	CHOICE	c:o.3		
							5.1.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.4		
							5.1.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							5.1.1.2.1.2	AttributeValue	ANY	c:m		
							5.1.1.2.2	nonSpecificForm	OCTET STRING	c:o.4		
							5.1.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.4		
							5.1.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							5.1.1.2.3.2	AttributeValue	ANY	c:m		
							5.1.2	additionalInfo	SET OF SEQUENCE	c:o		
							5.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							5.1.2.2	significance	BOOLEAN	c:o		
							5.1.2.3	information	ANY DEFINED BY identifier	c:m		
							5.2	RevertReply	Reply Syntax SEQUENCE	c:m		
							5.2.1	appliedPatches	SEQUENCE OF CHOICE	c:m		
							5.2.1.1	patchId	GraphicString	c:o.5		
							5.2.1.2	patchPointer	CHOICE	c:o.5		
							5.2.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.6		
							5.2.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							5.2.1.2.1.2	AttributeValue	ANY	c:m		
							5.2.1.2.2	nonSpecificForm	OCTET STRING	c:o.6		
							5.2.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.6		
							5.2.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		

Table C.2 (continued) – Action support summary

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							5.2.1.2.3.2	AttributeValue	ANY	c:m		
							5.2.2	additionalInfo	SET OF SEQUENCE	c:o		
							5.2.2.1	identifier	OBJECT IDENTIFIER	c:m		
							5.2.2.2	significance	BOOLEAN	c:o		
							5.2.2.3	information	ANY DEFINED BY identifier	c:m		
6	terminateValidation	{swmf-act 5}	softwareProcessingFailureParameter	o.1			6.1	TerminateValidationArgument	Information Syntax SEQUENCE	c:m		
							6.1.1	terminateValidationInfo	ENUMERATED	c:m		
							6.1.2	additionalInfo	SET OF SEQUENCE	c:o		
							6.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							6.1.2.2	significance	BOOLEAN	c:o		
							6.1.2.3	information	ANY DEFINED BY identifier	c:m		
							6.2	TerminateValidationReply	Reply Syntax CHOICE	c:m		
							6.2.1	noOutStandingValidation	NULL	c:o.7		
							6.2.2	validationCancelled	NULL	c:o.7		
							6.2.3	resultOfPartialValidation	CHOICE	c:o.7		
							6.2.3.1	validationTerminated	NULL	c:o.8		
							6.2.3.2	passValidation	NULL	c:o.8		
							6.2.3.3	passValidationWithResult	SET OF SEQUENCE	c:o.8		
							6.2.3.3.1	identifier	OBJECT IDENTIFIER	c:m		
							6.2.3.3.2	significance	BOOLEAN	c:o		
							6.2.3.3.3	information	ANY DEFINED BY identifier	c:m		
							6.2.3.4	failValidation	NULL	c:o.8		
							6.2.3.5	failValidationWithResult	SET OF SEQUENCE	c:o.8		
							6.2.3.5.1	identifier	OBJECT IDENTIFIER	c:m		
							6.2.3.5.2	significance	BOOLEAN	c:o		

Table C.2 (continued) – Action support summary

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							6.2.3.5.3	information	ANY DEFINED BY identifier	c:m		
7	validate	{swmf-act 6}	softwareProcessingFailureParameter	o.1			7.1	ValidateInfo	Information Syntax CHOICE	c:m		
							7.1.1	instanceDefaultValidationType	NULL	c:o.9		
							7.1.2	registeredValidationType	OBJECT IDENTIFIER	c:o.9		
							7.1.3	systemSpecificValidationType	SET OF SEQUENCE	c:o.9		
							7.1.3.1	identifier	OBJECT IDENTIFIER	c:m		
							7.1.3.2	significance	BOOLEAN	c:o		
							7.1.3.3	information	ANY DEFINED BY identifier	c:m		
							7.2	ValidateReply	Reply Syntax CHOICE	c:m		
							7.2.1	validationTerminated	NULL	c:o.10		
							7.2.2	passValidation	NULL	c:o.10		
							7.2.3	passValidationWithResult	SET OF SEQUENCE	c:o.10		
							7.2.3.1	identifier	OBJECT IDENTIFIER	c:m		
							7.2.3.2	significance	BOOLEAN	c:o		
							7.2.3.3	information	ANY DEFINED BY identifier	c:m		
							7.2.4	failValidation	NULL	c:o.10		
							7.2.5	failValidationWithResult	SET OF SEQUENCE	c:o.10		
							7.2.5.1	identifier	OBJECT IDENTIFIER	c:m		
							7.2.5.2	significance	BOOLEAN	c:o		
							7.2.5.3	information	ANY DEFINED BY identifier	c:m		
8	deliver	{swmf-act 1}	softwareProcessingFailureParameter	o.1			8.1	DeliverInfo	Information Syntax SEQUENCE	c:m		
							8.1.1	deliverId	CHOICE	o		
							8.1.1.1	globalValue	OBJECT IDENTIFIER	c:o.1		
							8.1.1.2	localValue	INTEGER	c:o.1		
							8.1.2	targetSoftware	SET OF CHOICE	m		

Table C.2 (concluded) – Action support summary

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							8.1.2.1	distributedSoftwareId	GraphicString	o:2		
							8.1.2.2	distributedSoftwarePointer	CHOICE	o:2		
							8.1.2.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o:3		
							8.1.2.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							8.1.2.2.1.2	AttributeValue	ANY	c:m		
							8.1.2.2.2	nonSpecificForm	OCTET STRING	c:o:3		
							8.1.2.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o:3		
							8.1.2.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							8.1.2.2.3.2	AttributeValue	ANY	c:m		
							8.1.3	targetSystem	CHOICE	o		
							8.1.3.1	single	AE-title	c:o:4		
							8.1.3.2	multiple	SET OF AE-title	c:o:4		
							8.1.4	transferInfo	SEQUENCE	o		
							8.1.4.1	transferProtocol	CHOICE	c:m		
							8.1.4.1.1	globalValue	OBJECT IDENTIFIER	c:o:5		
							8.1.4.1.2	localValue	INTEGER	c:o:5		
							8.1.4.2	protocolSpecificInfo	ANY DEFINED BY transferProtocol	c:o		
							8.1.5	additionalInfo	SET OF SEQUENCE	o		
							8.1.5.1	identifier	OBJECT IDENTIFIER	c:m		
							8.1.5.2	significance	BOOLEAN	c:o		
							8.1.5.3	information	ANY DEFINED BY identifier	c:m		

C.4.3 Notifications

The specifier of a manager role implementation that claims to support notifications specified in this Recommendation | International Standard shall import a copy of Table C.3 and complete it.

Table C.3 – Notification support summary

						Support									
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information	
1	attributeValueChange	{dmi-not1}		o.1				1.1	AttributeValueChangeInfo		Information Syntax SEQUENCE	c:m			
								1.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o			
								1.1.2	attributeIdentifierList	{dmi-att 8}	SET OF CHOICE	c:o			
								1.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o.1			
								1.1.2.2	localForm	-	INTEGER	c:o.1			
								1.1.3	attributeValueChangeDefinition	{dmi-att 10}	SET OF SEQUENCE	c:m			
								1.1.3.1	attributeID	-	CHOICE	c:m			
								1.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.2			
								1.1.3.1.2	localForm	-	INTEGER	c:o.2			
								1.1.3.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o			
								1.1.3.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m			
								1.1.4	notificationIdentifier	{dmi-att 16}	INTEGER	c:o			
								1.1.5	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o			
								1.1.5.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m			
								1.1.5.2	sourceObjectInst	-	CHOICE	c:o			
								1.1.5.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.3			
								1.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								1.1.5.2.1.2	AttributeValue	-	ANY	c:m			
								1.1.5.2.2	nonSpecificForm	-	OCTET STRING	c:o.3			
								1.1.5.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.3			
								1.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								1.1.5.2.3.2	AttributeValue	-	ANY	c:m			

Table C.3 (continued) – Notification support summary

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								1.1.6	additionalText	{dmi-att 7}	GraphicString	c:o		
								1.1.7	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								1.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m		
								1.1.7.2	significance	-	BOOLEAN	c:o		
								1.1.7.3	information	-	ANY DEFINED BY identifier	c:m		
2	autoBackupReport	{2 9 2 18 10 1}		o.1				2.1	AutoBackupReportInfo		Information Syntax SEQUENCE	c:m		
								2.1.1	backupResult	-	CHOICE	c:m		
								2.1.1.1	inLine	-	CHOICE	c:o.4		
								2.1.1.2	local	-	SEQUENCE	c:o.4		
								2.1.1.2.1	destination	-	CHOICE	c:m		
								2.1.1.2.1.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.5		
								2.1.1.2.1.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								2.1.1.2.1.1.2	AttributeValue	-	ANY	c:m		
								2.1.1.2.1.2	nonSpecificForm	-	OCTET STRING	c:o.5		
								2.1.1.2.1.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.5		
								2.1.1.2.1.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								2.1.1.2.1.3.2	AttributeValue	-	ANY	c:m		
								2.1.1.2.2	success	-	BOOLEAN	c:m		
								2.1.1.3	offLine	-	SEQUENCE	c:o.4		
								2.1.1.3.1	destination	-	GraphicString	c:m		
								2.1.1.3.2	result	-	CHOICE	c:m		
3	autoRestoreReport	{2 9 2 18 10 2}		o.1				3.1	AutoRestoreReportInfo		Information Syntax SEQUENCE	c:m		
								3.1.1	source	-	CHOICE	c:m		
								3.1.1.1	localObject	-	CHOICE	c:o.6		
								3.1.1.1.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.7		

Table C.3 (continued) – Notification support summary

							Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information	
								3.1.1.1.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								3.1.1.1.1.2	AttributeValue	-	ANY	c:m			
								3.1.1.1.2	nonSpecificForm	-	OCTET STRING	c:o.7			
								3.1.1.1.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.7			
								3.1.1.1.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								3.1.1.1.3.2	AttributeValue	-	ANY	c:m			
								3.1.1.2	remoteSystem	-	GraphicString	c:o.6			
								3.1.2	success	-	BOOLEAN	c:m			
								3.1.3	additionalInfo	-	SET OF SEQUENCE	c:o			
								3.1.3.1	identifier	-	OBJECT IDENTIFIER	c:m			
								3.1.3.2	significance	-	BOOLEAN	c:o			
								3.1.3.3	information	-	ANY DEFINED BY identifier	c:m			
4	objectCreation	{dmi-not6}		o.1				4.1	ObjectInfo		Information Syntax SEQUENCE	c:m			
								4.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o			
								4.1.2	attributeList	{dmi-att 9}	SET OF SEQUENCE	c:o			
								4.1.2.1	attributeId	-	CHOICE	c:m			
								4.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.8			
								4.1.2.1.2	localForm	-	INTEGER	c:o.8			
								4.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m			
								4.1.3	notificationIdentifier	{dmi-att 16}	INTEGER	c:o			
								4.1.4	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o			
								4.1.4.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m			
								4.1.4.2	sourceObjectInst	-	CHOICE	c:o			

Table C.3 (continued) – Notification support summary

							Support								
Index	Notifi- cation type template label	Value of object identifier for notifica- tion type	Con- straints and values	Sta- tus	Con- firm- ed	Non- con- firm- ed	Additional infor- mation	Subindex	Notification field name label	Value of object identifier of attribute type associa- ted with field	Constraints and values	Sta- tus	Sup- port	Additional infor- mation	
								4.1.4.2.1	distinguish- edName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9			
								4.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								4.1.4.2.1.2	AttributeVal- ue	-	ANY	c:m			
								4.1.4.2.2	nonSpecificFo- rm	-	OCTET STRING	c:o.9			
								4.1.4.2.3	localDistingui- shedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9			
								4.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								4.1.4.2.3.2	AttributeVal- ue	-	ANY	c:m			
								4.1.5	additionalText	{dmi-att 7}	GraphicString	c:o			
								4.1.6	additionalInfo- rmation	{dmi-att 6}	SET OF SEQUENCE	c:o			
								4.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m			
								4.1.6.2	significance	-	BOOLEAN	c:o			
								4.1.6.3	information	-	ANY DEFINED BY identifier	c:m			
5	objectDe- letion	{dmi-not7}		o.1				5.1	ObjectInfo		Information Syntax SEQUENCE	c:m			
								5.1.1	sourceIndicat- or	{dmi-att 26}	ENUMERATED	c:o			
								5.1.2	attributeList	{dmi-att 9}	SET OF SEQUENCE	c:o			
								5.1.2.1	attributeId	-	CHOICE	c:m			
								5.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 10			
								5.1.2.1.2	localForm	-	INTEGER	c:o. 10			
								5.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m			
								5.1.3	notificationId- entifier	{dmi-att 16}	INTEGER	c:o			
								5.1.4	correlatedNoti- fications	{dmi-att 12}	SET OF SEQUENCE	c:o			
								5.1.4.1	correlatedNoti- fications	{dmi-att 12}	SET OF INTEGER	c:m			

Table C.3 (continued) – Notification support summary

							Support							
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
								5.1.4.2	sourceObjectInst	-	CHOICE	c:o		
								5.1.4.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.11		
								5.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								5.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								5.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o.11		
								5.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.11		
								5.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								5.1.4.2.3.2	AttributeValue	-	ANY	c:m		
								5.1.5	additionalText	{dmi-att 7}	GraphicString	c:o		
								5.1.6	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								5.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		
								5.1.6.2	significance	-	BOOLEAN	c:o		
								5.1.6.3	information	-	ANY DEFINED BY identifier	c:m		
6	processingErrorAlarm	{dmi-not10}		o.1			alarmEffectOnServiceParameter	6.1	AlarmInfo		Information Syntax SEQUENCE	c:m		
								6.1.1	probableCause	{dmi-att 18}	CHOICE	c:m		
								6.1.1.1	globalValue	-	OBJECT IDENTIFIER	o.12		
								6.1.1.2	localValue	-	INTEGER	o.12		
								6.1.2	specificProblems	{dmi-att 27}	SET OF CHOICE	o		
								6.1.2.1	OBJECT IDENTIFIER	-	OBJECT IDENTIFIER	c:o.13		
								6.1.2.2	INTEGER	-	INTEGER	c:o.13		
								6.1.3	perceivedSeverity	{dmi-att 17}	ENUMERATED	c:m		

Table C.3 (continued) – Notification support summary

Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Support		Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
					Confirmed	Non-confirmed								
								6.1.4	backedUpStatus	{dmi-att 11}	BOOLEAN	o		
								6.1.5	backUpObject	{dmi-att 40}	CHOICE	o		
								6.1.5.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 14		
								6.1.5.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.5.1.2	AttributeValue	-	ANY	c:m		
								6.1.5.2	nonSpecificForm	-	OCTET STRING	c:o. 14		
								6.1.5.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 14		
								6.1.5.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.5.3.2	AttributeValue	-	ANY	c:m		
								6.1.6	trendIndication	{dmi-att 30}	ENUMERATED	o		
								6.1.7	thresholdInfo	{dmi-att 29}	SEQUENCE	o		
								6.1.7.1	triggeredThreshold	-	CHOICE	c:m		
								6.1.7.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 15		
								6.1.7.1.2	localForm	-	INTEGER	c:o. 15		
								6.1.7.2	observedValue	-	CHOICE	c:m		
								6.1.7.2.1	integer	-	INTEGER	c:o. 16		
								6.1.7.2.2	real	-	REAL	c:o. 16		
								6.1.7.3	thresholdLevel	-	CHOICE	c:o		
								6.1.7.3.1	up	-	SEQUENCE	c:o. 17		
								6.1.7.3.1.1	high	-	CHOICE	c:m		
								6.1.7.3.1.1.1	integer	-	INTEGER	c:o. 18		
								6.1.7.3.1.1.2	real	-	REAL	c:o. 18		

Table C.3 (continued) – Notification support summary

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								6.1.7.3.1.2	low	-	CHOICE	c:o		
								6.1.7.3.1.2.1	integer	-	INTEGER	c:o.19		
								6.1.7.3.1.2.2	real	-	REAL	c:o.19		
								6.1.7.3.2	down	-	SEQUENCE	c:o.17		
								6.1.7.3.2.1	high	-	CHOICE	c:m		
								6.1.7.3.2.1.1	integer	-	INTEGER	c:o.20		
								6.1.7.3.2.1.2	real	-	REAL	c:o.20		
								6.1.7.3.2.2	low	-	CHOICE	c:m		
								6.1.7.3.2.2.1	integer	-	INTEGER	c:o.21		
								6.1.7.3.2.2.2	real	-	REAL	c:o.21		
								6.1.7.4	armTime	-	GeneralizedTime	c:o		
								6.1.8	notificationIdentifier	{dmi-att 16}	INTEGER	o		
								6.1.9	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	o		
								6.1.9.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								6.1.9.2	sourceObjectInst	-	CHOICE	c:o		
								6.1.9.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.22		
								6.1.9.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.9.2.1.2	AttributeValue	-	ANY	c:m		
								6.1.9.2.2	nonSpecificForm	-	OCTET STRING	c:o.22		
								6.1.9.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.22		
								6.1.9.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.9.2.3.2	AttributeValue	-	ANY	c:m		
								6.1.10	stateChangeDefinition	{dmi-att 28}	SET OF SEQUENCE	o		

Table C.3 (continued) – Notification support summary

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								6.1.10.1	attributeID	-	CHOICE	c:m		
								6.1.10.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.23		
								6.1.10.1.2	localForm	-	INTEGER	c:o.23		
								6.1.10.2	oldAttributeV alue	-	ANY DEFINED BY attributeID	c:o		
								6.1.10.3	newAttribut eValue	-	ANY DEFINED BY attributeID	c:m		
								6.1.11	monitoredAttr ibutes	{dmi-att 15}	SET OF SEQUENCE	o		
								6.1.11.1	attributeId	-	CHOICE	c:m		
								6.1.11.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.24		
								6.1.11.1.2	localForm	-	INTEGER	c:o.24		
								6.1.11.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								6.1.12	proposedRepa irActions	{dmi-att 19}	SET OF CHOICE	o		
								6.1.12.1	OBJECT IDENTIFIER	-	OBJECT IDENTIFIER	c:o.25		
								6.1.12.2	INTEGER	-	INTEGER	c:o.25		
								6.1.13	additionalText	{dmi-att 7}	GraphicString	o		
								6.1.14	additionalInfo rmation	{dmi-att 6}	SET OF SEQUENCE	o		
								6.1.14.1	identifier	-	OBJECT IDENTIFIER	c:m		
								6.1.14.2	significance	-	BOOLEAN	c:o		
								6.1.14.3	information	-	ANY DEFINED BY identifier	c:m		
7	stateCha nge	{dmi- not14}		o.1				7.1	StateChangeI nfo		Information Syntax SEQUENCE	c:m		
								7.1.1	sourceIndicat or	{dmi-att 26}	ENUMERATED	c:o		
								7.1.2	attributeIdenti fierList	{dmi-att 8}	SET OF CHOICE	c:o		
								7.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o.26		

Table C.3 (continued) – Notification support summary

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								7.1.2.2	localForm	-	INTEGER	c:o.26		
								7.1.3	stateChangedDefinition	{dmi-att 28}	SET OF SEQUENCE	c:m		
								7.1.3.1	attributeID	-	CHOICE	c:m		
								7.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.27		
								7.1.3.1.2	localForm	-	INTEGER	c:o.27		
								7.1.3.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o		
								7.1.3.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m		
								7.1.4	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								7.1.5	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								7.1.5.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								7.1.5.2	sourceObjectInst	-	CHOICE	c:o		
								7.1.5.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.28		
								7.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								7.1.5.2.1.2	AttributeValue	-	ANY	c:m		
								7.1.5.2.2	nonSpecificForm	-	OCTET STRING	c:o.28		
								7.1.5.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.28		
								7.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								7.1.5.2.3.2	AttributeValue	-	ANY	c:m		
								7.1.6	additionalText	{dmi-att 7}	GraphicString	c:o		
								7.1.7	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		

Table C.3 (concluded) – Notification support summary

					Support									
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
								7.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m		
								7.1.7.2	significance	-	BOOLEAN	c:o		
								7.1.7.3	information	-	ANY DEFINED BY identifier	c:m		
8	deliverResultNotification	{swmf-not 3}		o.1				8.1	DeliverResultInfo		Information Syntax SEQUENCE	c:m		
								8.1.1	deliverId	-	CHOICE	o		
								8.1.1.1	globalValue	-	OBJECT IDENTIFIER	c:o.1		
								8.1.1.2	localValue	-	INTEGER	c:o.1		
								8.1.2	deliverResult	-	ENUMERATED	m		
								8.1.3	additionalInfo	-	SET OF SEQUENCE	o		
								8.1.3.1	identifier	-	OBJECT IDENTIFIER	c:m		
								8.1.3.2	significance	-	BOOLEAN	c:o		
								8.1.3.3	information	-	ANY DEFINED BY identifier	c:m		

C.4.4 Create and delete management operations

The specifier of a manager role implementation that claims to support the create or delete management operations on the managed objects specified in this Recommendation | International Standard shall import a copy of this table and complete it.

Table C.4 – Create and delete support

Index	Operation	Constraints and values	Status	Support	Additional information
1	Create support	softwareUnit, executableSoftware or softwareDistributor	o.1		
1.1	Create with reference object	–	c:o		
2	Delete support	softwareUnit, executableSoftware or softwareDistributor	o.1		

Formulario de MOCS

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

D.1 Introduction

The purpose of this MOCS proforma is to provide a mechanism for a supplier of an implementation which claims conformance, in the agent role, to a managed object class, to provide conformance information in a standard form.

D.2 Instructions for completing the MOCS proforma to produce a MOCS

The MOCS proforma contained in this annex is comprised of information in tabular form, in accordance with ITU-T Rec. X.724 | ISO/IEC 10165-6. The supplier of the implementation shall state which items are supported in the tables below and if necessary, provide additional information.

D.3 Symbols, abbreviations and terms

The following abbreviations are used throughout the MOCS proforma:

dmi-att joint-iso-ccitt ms(9) smi(3) part2(2) attribute(7)
dmi-nb joint-iso-ccitt ms(9) smi(3) part2(2) nameBinding(6)
dmi-not joint-iso-ccitt ms(9) smi(3) part2(2) notification(10)
dmi-obj joint-iso-ccitt ms(9) smi(3) part2(2) managedObjectClass(3)
dmi-pkg joint-iso-ccitt ms(9) smi(3) part2(2) package(4)

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

D.4 Software unit managed object class

D.4.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the software unit managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed.

Table D.1 – Software Unit Managed object class support

Index	Managed object class template label	Value of object identifier for class	Support of all mandatory features? (Y/N)	Is the actual class the same as the managed object class to which conformance is claimed? (Y/N)
1	softwareUnit	{swmf-obj 1}		

If the answer to the actual class question in Table D.1 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.3.

Table D.2 – Software Unit Actual class support

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

⁵⁾ **Comunicado sobre derechos de autor del formulario de MOCS**

Los usuarios de esta Recomendación | Norma Internacional pueden reproducir libremente el formulario de MOCS de este anexo a fin de que pueda ser utilizado para los fines previstos, y pueden además publicar el MOCS cumplimentado. En la Rec. UIT-T X.724 | ISO/CEI 10165-6 se especifican las instrucciones para rellenar el formulario de MOCS.

D.4.2 Packages

Table D.3 – Software Unit Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	administrativeOperationalStatesPackage	{m3100-pkg 1}	Mandatory	m		
2	affectedObjectListPackage	{m3100-pkg 2}	“an instance supports it”	o		
3	allomorphicPackage	{dmi-pkg 17}	“if an object supports allomorphism”	c1		
4	appliedPatchPackage	{swmf-pkg 1}	“an instance supports software patching”	o		
5	attributeValueChangeNotificationPackage	{m3100-pkg 4}	“the attributeValueChange notification defined in Recommendation X.721 is supported by an instance of this class.”	o		
6	checkSumPackage	{swmf-pkg 2}	“an instance supports it”	o		
7	createDeleteNotificationsPackage	{m3100-pkg 10}	“the objectCreation and objectDeletion notifications defined in Recommendation X.721 are supported by an instance of this class.”	o		
8	currentProblemListPackage	{m3100-pkg 13}	“an instance supports it”	o		
9	fileInformationPackage	{swmf-pkg 4}	“an instance supports it”	o		
10	filePackage	{swmf-pkg 5}	“an instance supports it”	o		
11	informationAutoBackupPackage	{swmf-pkg 6}	“an instance supports it”	o		
12	informationAutoRestorePackage	{swmf-pkg 7}	“an instance supports it”	o		
13	informationBackupPackage	{swmf-pkg 8}	“an instance supports it”	o		
14	informationRestorePackage	{swmf-pkg 9}	“an instance supports it”	o		
15	installPackage	{swmf-pkg 10}	“an instance supports it”	o		
16	noteFieldPackage	{swmf-pkg 11}	“an instance supports it”	o		
17	packagesPackage	{dmi-pkg 16}	“any registered package, other than this package has been instantiated.”	m		
18	processingErrorAlarmOnServicePackage	{swmf-pkg 12}	Mandatory	m		
19	revertPackage	{swmf-pkg 13}	“an instance supports it”	o		
20	softwarePackage		Mandatory	m		
21	softwareProcessingErrorAlarmPackage	{m3100-pkg 26}	“an instance supports it”	o		
22	softwareUnitPackage		Mandatory	m		
23	stateChangeNotificationPackage	{m3100-pkg 28}	“the stateChange notification defined in Recommendation X.721 is supported by an instance of this class.”	o		
24	terminateValidationPackage	{swmf-pkg 14}	“the validationPackage is present and an instance supports it.”	c2		
25	topPackage		Mandatory	m		
26	usageStatePackage	{swmf-pkg 15}	“an instance supports it”	o		
27	userLabelPackage	{m3100-pkg 32}	“an instance supports it”	o		
28	validationPackage	{swmf-pkg 16}	“an instance supports it”	o		
29	vendorNamePackage	{m3100-pkg 33}	“an instance supports it”	o		
30	versionPackage	{m3100-pkg 34}	“an instance supports it”	o		
c1: if D.41/1b then – else m. c2: if D.4.3/28a then o else –.						

Table D.4 – Software Unit Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
1	administrativeState	{dmi-att 31}	ENUMERATED	m		m		m		-		-		c3		
2	affectedObjectList	{m3100-att 2}	SET OF CHOICE	c4		c4a		c4		c4		c4		c4		
3	alarmStatus	{m3100-att 6}	ENUMERATED	c5		c5a		c5		-		-		c5		
4	allomorphs	{dmi-att 50}	SET OF CHOICE	c6		c7		-		-		-		-		
5	appliedPatches	{swmf-att 1}	SEQUENCE OF CHOICE	c8		c9		c8		-		-		c8		
6	availabilityStatus	{dmi-att 33}	SET OF INTEGER	c3		m		c3		c3		c3		c3		
7	checkSum	{swmf-att 2}	BIT STRING	c10		c11		c10		-		-		c10		
8	currentProblemList	{m3100-att 17}	SET OF SEQUENCE	c12		c13		c12		c12		c12		c12		
9	dateDelivered	{swmf-att 3}	CHOICE	c14		c15		c14		-		-		c14		
10	dateInstalled	{swmf-att 4}	CHOICE	c16		c17		c16		-		-		c16		
11	dateOfCreation	{swmf-att 5}	GeneralizedTime	c18		c19		c18		-		-		c18		
12	dateOfLastModification	{swmf-att 6}	CHOICE	c20		c21		c20		-		-		c20		
13	fileLocation	{swmf-att 7}	CHOICE	c22		c23		c22		-		-		c22		
14	fileSize	{swmf-att 8}	CHOICE	c24		c25		c24		-		-		c24		
15	fileType	{swmf-att 9}		o		o		-		-		-		-		
16	futureAutoBackupDestination	{swmf-att 10}	BackupDestination	c26		c26		c26		-		-		c27		
17	futureAutoBackupTriggerThreshold	{swmf-att 11}	INTEGER	c26		c26		c26		-		-		c27		
18	futureAutoRestoreAllowed	{swmf-att 12}	BOOLEAN	c28		c28		c28		-		-		c29		
19	futureAutoRestoreSource	{swmf-att 13}	CHOICE	c28		c28		c28		-		-		c29		
20	identityOfCreator	{swmf-att 14}	GraphicString	c18		c19		c18		-		-		c18		
21	identityOfLastModifier	{swmf-att 15}	GraphicString	c18		c19		c18		-		-		c18		
22	lastBackupDestination	{swmf-att 16}	CHOICE	c30		c31		c30		-		-		c30		
23	lastBackupTime	{swmf-att 17}	CHOICE	c30		c31		c30		-		-		c30		
24	lastRestoreSource	{swmf-att 18}	CHOICE	c32		c33		c32		-		-		c32		
25	lastRestoreTime	{swmf-att 19}	CHOICE	c32		c33		c32		-		-		c32		
26	nameBinding	{dmi-att 63}	OBJECT IDENTIFIER	o		m		x		-		-		x		
27	noteField	{swmf-att 20}	GraphicString	c34		c34		c34		-		-		c35		
28	objectClass	{dmi-att 65}	CHOICE	m		m		x		-		-		x		

Table D.4 (concluded) – Software Unit Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
29	operationalState	{dmi-att 35}	ENUMERATED	x		m		x		-		-		x		
30	packages	{dmi-att 66}	SET OF OBJECT IDENTIFIER	o		m		x		x		x		x		
31	proceduralStatus	{dmi-att 36}	SET OF INTEGER	o		m		-		-		-		-		
32	softwareId	{m3100-att 38}	CHOICE	o		m		x		-		-		x		
33	usageState	{dmi-att 39}	ENUMERATED	-		o		-		-		-		-		
34	userLabel	{m3100-att 50}	GraphicString	o		o		o		-		-		-		
35	vendorName	{m3100-att 51}	GraphicString	o		o		o		-		-		-		
36	version	{m3100-att 52}	GraphicString	o		o		o		-		-		-		

c3: if D.4.1/1b then x else –.
 c4: if D.4.3/2a and D.4.1/1b then x else –.
 c4a: if D.4.3/2a then m else –.
 c5: if D.4.3/21a and D.4.1/1b then x else –.
 c5a: if D.4.3/21a then m else –.
 c6: if D.4.3/3a then o else –.
 c7: if D.4.3/3a then m else –.
 c8: if D.4.3/4a and D.4.1/1b then x else –.
 c9: if D.4.3/4a then m else –.
 c10: if D.4.3/6a and D.4.1/1b then x else –.
 c11: if D.4.3/6a then m else –.
 c12: if D.4.3/8a and D.4.1/1b then x else –.
 c13: if D.4.3/8a then m else –.
 c14: if D.4.3/9a and D.4.1/1b then x else –.
 c15: if D.4.3/9a then m else –.
 c16: if D.4.3/9a and D.4.1/1b then x else –.
 c17: if D.4.3/9a then m else –.
 c18: if D.4.3/9a and D.4.1/1b then x else –.
 c19: if D.4.3/9a then m else –.
 c20: if D.4.3/9a and D.4.1/1b then x else –.
 c21: if D.4.3/9a then m else –.
 c22: if D.4.3/10a and D.4.1/1b then x else –.
 c23: if D.4.3/10a then m else –.
 c24: if D.4.3/10a and D.4.1/1b then x else –.
 c25: if D.4.3/10a then m else –.
 c26: if D.4.3/11a then m else –.
 c27: if D.4.3/11a and D.4.1/1b then x else –.
 c28: if D.4.3/12a then m else –.
 c29: if D.4.3/12a and D.4.1/1b then x else –.
 c30: if D.4.3/13a and D.4.1/1b then x else –.
 c31: if D.4.3/13a then m else –.
 c32: if D.4.3/14a and D.4.1/1b then x else –.
 c33: if D.4.3/14a then m else –.
 c34: if D.4.3/16a then m else –.
 c35: if D.4.3/16a and D.4.1/1b then x else –.

D.4.4 Attribute groups

There are no attribute groups specified for this managed object class.

D.4.5 Actions**Table D.5 – Software Unit Action support**

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	backup	{swmf-act 7}	softwareProcessingFailureParameter	c1			1.1	BackupArgument	Information Syntax SEQUENCE	c1		
							1.1.1	backupDestination	CHOICE	c:m		
							1.1.1.1	localObject	Object Instance	c:o.1		
							1.1.1.2	inLine	NULL	c:o.1		
							1.1.1.3	offLine	GraphicString	c:o.1		
							1.1.2	additionalInfo	SET OF ANY	c:o		
							1.2	BackupReply	Reply Syntax SEQUENCE	c1		
							1.2.1	reply	CHOICE	c:m		
							1.2.1.1	success	NULL	c:o.2		
							1.2.1.2	inLine	BIT STRING	c:o.2		
							1.2.2	additionalInfo	SET OF ANY	c:o		
2	install	{swmf-act 3}	softwareProcessingFailureParameter	c2			2.1	InstallInfo	Information Syntax SEQUENCE	c2		
							2.1.1	targetSoftware	SET OF CHOICE	c:m		
							2.1.1.1	distributedSoftwareId	GraphicString	c:o.1		
							2.1.1.2	distributedSoftwarePointer	CHOICE	c:o.1		
							2.1.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.2		
							2.1.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							2.1.1.2.1.2	AttributeValue	ANY	c:m		
							2.1.1.2.2	nonSpecificForm	OCTET STRING	c:o.2		
							2.1.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.2		
							2.1.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							2.1.1.2.3.2	AttributeValue	ANY	c:m		
							2.1.2	installInfo	SET OF SEQUENCE	c:m		
							2.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							2.1.2.2	significance	BOOLEAN	c:o		
							2.1.2.3	information	ANY DEFINED BY identifier	c:m		

Table D.5 (continued) – Software Unit Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
3	restore	{swmf-act 8}		c3			3.1	RestoreArgument	Information Syntax SEQUENCE	c3		
							3.1.1	restoreSource	CHOICE	c:m		
							3.1.1.1	localObject	Object Instance	c:o.1		
							3.1.1.2	inLine	BIT STRING	c:o.1		
							3.1.1.3	offLine	Printable String	c:o.1		
							3.1.2	additionalInfo	SET OF ANY	c:o		
4	revert	{swmf-act 4}	softwareProcessingFailureParameter	c4			4.1	RevertArgument	Information Syntax SEQUENCE	c4		
							4.1.1	revertInfo	SEQUENCE OF CHOICE	c:m		
							4.1.1.1	patchId	GraphicString	c:o.3		
							4.1.1.2	patchPointer	CHOICE	c:o.3		
							4.1.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.4		
							4.1.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							4.1.1.2.1.2	AttributeValue	ANY	c:m		
							4.1.1.2.2	nonSpecificForm	OCTET STRING	c:o.4		
							4.1.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.4		
							4.1.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							4.1.1.2.3.2	AttributeValue	ANY	c:m		
							4.1.2	additionalInfo	SET OF SEQUENCE	c:o		
							4.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							4.1.2.2	significance	BOOLEAN	c:o		
							4.1.2.3	information	ANY DEFINED BY identifier	c:m		
							4.2	RevertReply	Reply Syntax SEQUENCE	c4		
							4.2.1	appliedPatches	SEQUENCE OF CHOICE	c:m		
							4.2.1.1	patchId	GraphicString	c:o.5		
							4.2.1.2	patchPointer	CHOICE	c:o.5		
							4.2.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.6		

Table D.5 (continued) – Software Unit Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							4.2.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							4.2.1.2.1.2	AttributeValue	ANY	c:m		
							4.2.1.2.2	nonSpecificForm	OCTET STRING	c:o.6		
							4.2.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.6		
							4.2.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							4.2.1.2.3.2	AttributeValue	ANY	c:m		
							4.2.2	additionalInfo	SET OF SEQUENCE	c:o		
							4.2.2.1	identifier	OBJECT IDENTIFIER	c:m		
							4.2.2.2	significance	BOOLEAN	c:o		
							4.2.2.3	information	ANY DEFINED BY identifier	c:m		
5	terminateValidation	{swmf-act 5}	softwareProcessingFailureParameter	c5			5.1	TerminateValidationArgument	Information Syntax SEQUENCE	c5		
							5.1.1	terminateValidationInfo	ENUMERATED	c:m		
							5.1.2	additionalInfo	SET OF SEQUENCE	c:o		
							5.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							5.1.2.2	significance	BOOLEAN	c:o		
							5.1.2.3	information	ANY DEFINED BY identifier	c:m		
							5.2	TerminateValidationReply	Reply Syntax CHOICE	c5		
							5.2.1	noOutStandingValidation	NULL	c:o.7		
							5.2.2	validationCancelled	NULL	c:o.7		
							5.2.3	resultOfPartialValidation	CHOICE	c:o.7		
							5.2.3.1	validationTerminated	NULL	c:o.8		
							5.2.3.2	passValidation	NULL	c:o.8		
							5.2.3.3	passValidationWithResult	SET OF SEQUENCE	c:o.8		
							5.2.3.3.1	identifier	OBJECT IDENTIFIER	c:m		
							5.2.3.3.2	significance	BOOLEAN	c:o		
							5.2.3.3.3	information	ANY DEFINED BY identifier	c:m		

Table D.5 (concluded) – Software Unit Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							5.2.3.4	failValidation	NULL	c:o.8		
							5.2.3.5	failValidationWithResult	SET OF SEQUENCE	c:o.8		
							5.2.3.5.1	identifier	OBJECT IDENTIFIER	c:m		
							5.2.3.5.2	significance	BOOLEAN	c:o		
							5.2.3.5.3	information	ANY DEFINED BY identifier	c:m		
6	validate	{swmf-act 6}	softwareProcessingFailureParameter	c6			9.1	ValidateInfo	Information Syntax CHOICE	c6		
							9.1.1	instanceDefaultValidationType	NULL	c:o.9		
							9.1.2	registeredValidationType	OBJECT IDENTIFIER	c:o.9		
							9.1.3	systemSpecificValidationType	SET OF SEQUENCE	c:o.9		
							9.1.3.1	identifier	OBJECT IDENTIFIER	c:m		
							9.1.3.2	significance	BOOLEAN	c:o		
							9.1.3.3	information	ANY DEFINED BY identifier	c:m		
							9.2	ValidateReply	Reply Syntax CHOICE	c6		
							9.2.1	validationTerminated	NULL	c:o.10		
							9.2.2	passValidation	NULL	c:o.10		
							9.2.3	passValidationWithResult	SET OF SEQUENCE	c:o.10		
							9.2.3.1	identifier	OBJECT IDENTIFIER	c:m		
							9.2.3.2	significance	BOOLEAN	c:o		
							9.2.3.3	information	ANY DEFINED BY identifier	c:m		
							9.2.4	failValidation	NULL	c:o.10		
							9.2.5	failValidationWithResult	SET OF SEQUENCE	c:o.10		
							9.2.5.1	identifier	OBJECT IDENTIFIER	c:m		
							9.2.5.2	significance	BOOLEAN	c:o		
							9.2.5.3	information	ANY DEFINED BY identifier	c:m		
c1: if D.4.3/13a then m else –. c2: if D.4.2/15a then m else –. c3: if D.4.2/14a then m else –. c4: if D.4.2/19a then m else –. c5: if D.4.2/24a then m else –. c6: if D.4.2/28a then m else –.												

Table D.6 – Software Unit Notification support

						Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	attributeValueChange	{dmi-not1}		c1				1.1	AttributeValueChangeInfo		Information Syntax SEQUENCE	c1		
								1.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o		
								1.1.2	attributeIdentifierList	{dmi-att 8}	SET OF CHOICE	c:o		
								1.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o.1		
								1.1.2.2	localForm	-	INTEGER	c:o.1		
								1.1.3	attributeValueChangeDefinition	{dmi-att 10}	SET OF SEQUENCE	c:m		
								1.1.3.1	attributeID	-	CHOICE	c:m		
								1.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.2		
								1.1.3.1.2	localForm	-	INTEGER	c:o.2		
								1.1.3.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o		
								1.1.3.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m		
								1.1.4	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								1.1.5	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								1.1.5.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								1.1.5.2	sourceObjectInst	-	CHOICE	c:o		
								1.1.5.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.3		
								1.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								1.1.5.2.1.2	AttributeValue	-	ANY	c:m		
								1.1.5.2.2	nonSpecificForm	-	OCTET STRING	c:o.3		
								1.1.5.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.3		
								1.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								1.1.5.2.3.2	AttributeValue	-	ANY	c:m		
								1.1.6	additionalText	{dmi-att 7}	GraphicString	c:o		
								1.1.7	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								1.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m		

Table D.6 (continued) – Software Unit Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
2	autoBackupReport	{2 9 2 18 10 1}		c2				1.1.7.2	significance	-	BOOLEAN	c:o		
								1.1.7.3	information	-	ANY DEFINED BY identifier	c:m		
								2.1	AutoBackupReportInfo		Information Syntax SEQUENCE	c2		
								2.1.1	backupResult	-	CHOICE	c:m		
								2.1.1.1	inLine	-	CHOICE	c:o.4		
								2.1.1.2	local	-	SEQUENCE	c:o.4		
								2.1.1.2.1	destination	-	CHOICE	c:m		
								2.1.1.2.1.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.5		
								2.1.1.2.1.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								2.1.1.2.1.1.2	AttributeValue	-	ANY	c:m		
								2.1.1.2.1.2	nonSpecificForm	-	OCTET STRING	c:o.5		
								2.1.1.2.1.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.5		
								2.1.1.2.1.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								2.1.1.2.1.3.2	AttributeValue	-	ANY	c:m		
								2.1.1.2.2	success	-	BOOLEAN	c:m		
								2.1.1.3	offLine	-	SEQUENCE	c:o.4		
								2.1.1.3.1	destination	-	GraphicString	c:m		
								2.1.1.3.2	result	-	CHOICE	c:m		
3	autoRestoreReport	{2 9 2 18 10 2}		c3				3.1	AutoRestoreReportInfo		Information Syntax SEQUENCE	c3		
								3.1.1	source	-	CHOICE	c:m		
								3.1.1.1	localObject	-	CHOICE	c:o.6		
								3.1.1.1.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.7		
								3.1.1.1.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								3.1.1.1.1.2	AttributeValue	-	ANY	c:m		
								3.1.1.1.2	nonSpecificForm	-	OCTET STRING	c:o.7		
								3.1.1.1.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.7		
								3.1.1.1.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								3.1.1.1.3.2	AttributeValue	-	ANY	c:m		

Table D.6 (continued) – Software Unit Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								3.1.1.2	remoteSystem	-	GraphicString	c:o.6		
								3.1.2	success	-	BOOLEAN	c:m		
								3.1.3	additionalInfo	-	SET OF SEQUENCE	c:o		
								3.1.3.1	identifier	-	OBJECT IDENTIFIER	c:m		
								3.1.3.2	significance	-	BOOLEAN	c:o		
								3.1.3.3	information	-	ANY DEFINED BY identifier	c:m		
4	objectCreation	{dmi-not6}		c4				4.1	ObjectInfo		Information Syntax SEQUENCE	c4		
								4.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o		
								4.1.2	attributeList	{dmi-att 9}	SET OF SEQUENCE	c:o		
								4.1.2.1	attributeId	-	CHOICE	c:m		
								4.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.8		
								4.1.2.1.2	localForm	-	INTEGER	c:o.8		
								4.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								4.1.3	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								4.1.4	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								4.1.4.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								4.1.4.2	sourceObjectInst	-	CHOICE	c:o		
								4.1.4.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9		
								4.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								4.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								4.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o.9		
								4.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9		
								4.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								4.1.4.2.3.2	AttributeValue	-	ANY	c:m		
								4.1.5	additionalText	{dmi-att 7}	GraphicString	c:o		
								4.1.6	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								4.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		

Table D.6 (continued) – Software Unit Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								4.1.6.2	significance	-	BOOLEAN	c:o		
								4.1.6.3	information	-	ANY DEFINED BY identifier	c:m		
5	objectDeletion	{dmi-not7}		c5				5.1	ObjectInfo		Information Syntax SEQUENCE	c5		
								5.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o		
								5.1.2	attributeList	{dmi-att 9}	SET OF SEQUENCE	c:o		
								5.1.2.1	attributeId	-	CHOICE	c:m		
								5.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 10		
								5.1.2.1.2	localForm	-	INTEGER	c:o. 10		
								5.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								5.1.3	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								5.1.4	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								5.1.4.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								5.1.4.2	sourceObjectInst	-	CHOICE	c:o		
								5.1.4.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 11		
								5.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								5.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								5.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o. 11		
								5.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 11		
								5.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								5.1.4.2.3.2	AttributeValue	-	ANY	c:m		
								5.1.5	additionalText	{dmi-att 7}	GraphicString	c:o		
								5.1.6	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								5.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		
								5.1.6.2	significance	-	BOOLEAN	c:o		
								5.1.6.3	information	-	ANY DEFINED BY identifier	c:m		

Table D.6 (continued) – Software Unit Notification support

						Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
6	processingErrorAlarm	{dmi-not10}		m			alarmEffectOnServiceParameter	6.1	AlarmInfo		Information Syntax SEQUENCE	m		
								6.1.1	probableCause	{dmi-att18}	CHOICE	m		
								6.1.1.1	globalValue	-	OBJECT IDENTIFIER	o.12		
								6.1.1.2	localValue	-	INTEGER	o.12		
								6.1.2	specificProblems	{dmi-att27}	SET OF CHOICE	o		
								6.1.2.1	OBJECT IDENTIFIER	-	OBJECT IDENTIFIER	c:o.13		
								6.1.2.2	INTEGER	-	INTEGER	c:o.13		
								6.1.3	perceivedSeverity	{dmi-att17}	ENUMERATED	m		
								6.1.4	backedUpStatus	{dmi-att11}	BOOLEAN	o		
								6.1.5	backUpObject	{dmi-att40}	CHOICE	o		
								6.1.5.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.14		
								6.1.5.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.5.1.2	AttributeValue	-	ANY	c:m		
								6.1.5.2	nonSpecificForm	-	OCTET STRING	c:o.14		
								6.1.5.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.14		
								6.1.5.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.5.3.2	AttributeValue	-	ANY	c:m		
								6.1.6	trendIndication	{dmi-att30}	ENUMERATED	o		
								6.1.7	thresholdInfo	{dmi-att29}	SEQUENCE	o		
								6.1.7.1	triggeredThreshold	-	CHOICE	c:m		
								6.1.7.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.15		
								6.1.7.1.2	localForm	-	INTEGER	c:o.15		
								6.1.7.2	observedValue	-	CHOICE	c:m		
								6.1.7.2.1	integer	-	INTEGER	c:o.16		

Table D.6 (continued) – Software Unit Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								6.1.7.2.2	real	-	REAL	c:o.16		
								6.1.7.3	thresholdLevel	-	CHOICE	c:o		
								6.1.7.3.1	up	-	SEQUENCE	c:o.17		
								6.1.7.3.1.1	high	-	CHOICE	c:m		
								6.1.7.3.1.1.1	integer	-	INTEGER	c:o.18		
								6.1.7.3.1.1.2	real	-	REAL	c:o.18		
								6.1.7.3.1.2	low	-	CHOICE	c:o		
								6.1.7.3.1.2.1	integer	-	INTEGER	c:o.19		
								6.1.7.3.1.2.2	real	-	REAL	c:o.19		
								6.1.7.3.2	down	-	SEQUENCE	c:o.17		
								6.1.7.3.2.1	high	-	CHOICE	c:m		
								6.1.7.3.2.1.1	integer	-	INTEGER	c:o.20		
								6.1.7.3.2.1.2	real	-	REAL	c:o.20		
								6.1.7.3.2.2	low	-	CHOICE	c:m		
								6.1.7.3.2.2.1	integer	-	INTEGER	c:o.21		
								6.1.7.3.2.2.2	real	-	REAL	c:o.21		
								6.1.7.4	armTime	-	GeneralizedTime	c:o		
								6.1.8	notificationIdentifier	{dmi-att 16}	INTEGER	o		
								6.1.9	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	o		
								6.1.9.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								6.1.9.2	sourceObjectInst	-	CHOICE	c:o		
								6.1.9.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.22		
								6.1.9.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.9.2.1.2	AttributeValue	-	ANY	c:m		
								6.1.9.2.2	nonSpecificForm	-	OCTET STRING	c:o.22		
								6.1.9.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.22		

Table D.6 (continued) – Software Unit Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								6.1.9.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.9.2.3.2	AttributeValue	-	ANY	c:m		
								6.1.10	stateChangeDefinition	{dmi-att 28}	SET OF SEQUENCE	o		
								6.1.10.1	attributeID	-	CHOICE	c:m		
								6.1.10.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 23		
								6.1.10.1.2	localForm	-	INTEGER	c:o. 23		
								6.1.10.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o		
								6.1.10.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m		
								6.1.11	monitoredAttributes	{dmi-att 15}	SET OF SEQUENCE	o		
								6.1.11.1	attributeId	-	CHOICE	c:m		
								6.1.11.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 24		
								6.1.11.1.2	localForm	-	INTEGER	c:o. 24		
								6.1.11.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								6.1.12	proposedRepairActions	{dmi-att 19}	SET OF CHOICE	o		
								6.1.12.1	OBJECT IDENTIFIER	-	OBJECT IDENTIFIER	c:o. 25		
								6.1.12.2	INTEGER	-	INTEGER	c:o. 25		
								6.1.13	additionalText	{dmi-att 7}	GraphicString	o		
								6.1.14	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	o		
								6.1.14.1	identifier	-	OBJECT IDENTIFIER	c:m		
								6.1.14.2	significance	-	BOOLEAN	c:o		
								6.1.14.3	information	-	ANY DEFINED BY identifier	c:m		
7	stateChange	{dmi-not14}		c6				7.1	StateChangeInfo		Information Syntax SEQUENCE	c6		
								7.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o		
								7.1.2	attributeIdentifierList	{dmi-att 8}	SET OF CHOICE	c:o		
								7.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o. 26		
								7.1.2.2	localForm	-	INTEGER	c:o. 26		

Table D.6 (concluded) – Software Unit Notification support

							Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information	
								7.1.3	stateChangeDefinition	{ dmi-att 28 }	SET OF SEQUENCE	c:m			
								7.1.3.1	attributeID	-	CHOICE	c:m			
								7.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 27			
								7.1.3.1.2	localForm	-	INTEGER	c:o. 27			
								7.1.3.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o			
								7.1.3.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m			
								7.1.4	notificationIdentifier	{ dmi-att 16 }	INTEGER	c:o			
								7.1.5	correlatedNotifications	{ dmi-att 12 }	SET OF SEQUENCE	c:o			
								7.1.5.1	correlatedNotifications	{ dmi-att 12 }	SET OF INTEGER	c:m			
								7.1.5.2	sourceObjectInst	-	CHOICE	c:o			
								7.1.5.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 28			
								7.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								7.1.5.2.1.2	AttributeValue	-	ANY	c:m			
								7.1.5.2.2	nonSpecificForm	-	OCTET STRING	c:o. 28			
								7.1.5.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 28			
								7.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m			
								7.1.5.2.3.2	AttributeValue	-	ANY	c:m			
								7.1.6	additionalText	{ dmi-att 7 }	GraphicString	c:o			
								7.1.7	additionalInformation	{ dmi-att 6 }	SET OF SEQUENCE	c:o			
								7.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m			
								7.1.7.2	significance	-	BOOLEAN	c:o			
								7.1.7.3	information	-	ANY DEFINED BY identifier	c:m			
c1: if D.4.3/5a then m else –. c2: if D.4.3/11a then m else –. c3: if D.4.3/12a then m else –. c4: if D.4.3/7a then m else –. c5: if D.4.3/7a then m else –. c6: if D.4.3/23a then m else –.															

Table D.7 – Software Unit Parameter support

Index	Parameter template label	Value of object identifier for parameter	Constraints and values	Status	Support	Additional information
1	alarmEffectOnServiceParameter	{2 9 2 18 5 1}	EVENT-INFO processingErrorAlarm	m		
2	softwareProcessingFailureParameter	{2 9 2 18 5 2}	SPECIFIC-ERROR	c1		
c1: if D.4.3/13a or D.4.3/15a or D.4.3/19a or D.4.3/24a or D.4.3/28a then m else –.						

D.5 Executable software managed object class

D.5.1 Statement of conformance to the managed object class

The supplier of the implementation shall state whether or not all mandatory features of the executable software managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed.

Table D.8 – Executable Software Managed object class support

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

If the answer to the actual class question in Table D.8 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.10.

Table D.9 – Executable Software Actual class support

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

D.5.2 Packages

Table D.10 — Executable Software Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	administrativeOperationalStatesPackage	{m3100-pkg 1}	Mandatory	m		
2	affectedObjectListPackage	{m3100-pkg 2}	“an instance supports it”	o		
3	allomorphicPackage	{dmi-pkg 17}	“if an object supports allomorphy”	c1		
4	appliedPatchPackage	{swmf-pkg 1}	“an instance supports software patching”	o		
5	attributeValueChangeNotificationPackage	{m3100-pkg 4}	“the attributeValueChange notification defined in Recommendation X.721 is supported by an instance of this class.”	o		
6	checksumPackage	{swmf-pkg 2}	“an instance supports it”	o		
7	createDeleteNotificationsPackage	{m3100-pkg 10}	“the objectCreation and objectDeletion notifications defined in Recommendation X.721 are supported by an instance of this class.”	o		
8	currentProblemListPackage	{m3100-pkg 13}	“an instance supports it”	o		
9	fileInformationPackage	{swmf-pkg 4}	“an instance supports it”	o		
10	filePackage	{swmf-pkg 5}	“an instance supports it”	o		
11	informationAutoBackupPackage	{swmf-pkg 6}	“an instance supports it”	o		
12	informationAutoRestorePackage	{swmf-pkg 7}	“an instance supports it”	o		
13	informationBackupPackage	{swmf-pkg 8}	“an instance supports it”	o		
14	informationRestorePackage	{swmf-pkg 9}	“an instance supports it”	o		
15	installPackage	{swmf-pkg 10}	“an instance supports it”	o		
16	noteFieldPackage	{swmf-pkg 11}	“an instance supports it”	o		
17	packagesPackage	{dmi-pkg 16}	“any registered package, other than this package has been instantiated”	m		
18	processingErrorAlarmOnServicePackage	{swmf-pkg 12}	Mandatory	m		
19	revertPackage	{swmf-pkg 13}	“an instance supports it”	o		
20	softwarePackage		Mandatory	m		
21	softwareProcessingErrorAlarmPackage	{m3100-pkg 26}	“an instance supports it”	o		
22	softwareUnitPackage		Mandatory	m		
23	stateChangeNotificationPackage	{m3100-pkg 28}	“the stateChange notification defined in Recommendation X.721 is supported by an instance of this class.”	o		

Table D.10 (concluded) – Executable Software Package support

Index	Parameter template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
24	terminateValidationPackage	{swmf-pkg 14}	“the validationPackage is present and an instance supports it”	c2		
25	topPackage		Mandatory	m		
26	usageStatePackage	{swmf-pkg 15}	“an instance supports it”	o		
27	userLabelPackage	{m3100-pkg 32}	“an instance supports it”	o		
28	validationPackage	{swmf-pkg 16}	“an instance supports it”	o		
29	vendorNamePackage	{m3100-pkg 33}	“an instance supports it”	o		
30	versionPackage	{m3100-pkg 34}	“an instance supports it”	o		
31	executableSoftwarePackage	(Not registered)	Mandatory	m		
32	executeProgramPackage	{swmf-pkg 3}	“an instance supports it”	o		
c1: if D.5.1/1b then – else m. c2: if D.5.3/28a then o else –.						

D.5.3 Attributes

Table D.11 – Executable Software Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
1	administrativeState	{dmi-att 31}	ENUMERATED	m		m		m		-		-		c3		
2	affectedObjectList	{m3100-att 2}	SET OF CHOICE	c4		c4a		c4		c4		c4		c4		
3	alarmStatus	{m3100-att 6}	ENUMERATED	c5		c5a		c5		-		-		c5		
4	allomorphs	{dmi-att 50}	SET OF CHOICE	c6		c7		-		-		-		-		
5	appliedPatches	{swmf-att 1}	SEQUENCE OF CHOICE	c8		c9		c8		-		-		c8		
6	availabilityStatus	{dmi-att 33}	SET OF INTEGER	c3		m		c3		c3		c3		c3		
7	checksum	{swmf-att 2}	BIT STRING	c10		c11		c10		-		-		c10		
8	currentProblemList	{m3100-att 17}	SET OF SEQUENCE	c12		c13		c12		c12		c12		c12		

Table D.11 (continued) – Executable Software Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
9	dateDelivered	{swmf-att 3}	CHOICE	c14		c15		c14		-		-		c14		
10	dateInstalled	{swmf-att 4}	CHOICE	c16		c17		c16		-		-		c16		
11	dateOfCreation	{swmf-att 5}	GeneralizedTime	c18		c19		c18		-		-		c18		
12	dateOfLastModification	{swmf-att 6}	CHOICE	c20		c21		c20		-		-		c20		
13	fileLocation	{swmf-att 7}	CHOICE	c22		c23		c22		-		-		c22		
14	fileSize	{swmf-att 8}	CHOICE	c24		c25		c24		-		-		c24		
15	fileType	{swmf-att 9}		o		o		-		-		-		-		
16	futureAutoBackupDestination	{swmf-att 10}	BackupDestination	c26		c26		c26		-		-		c27		
17	futureAutoBackupTriggerThreshold	{swmf-att 11}	INTEGER	c26		c26		c26		-		-		c27		
18	futureAutoRestoreAllowed	{swmf-att 12}	BOOLEAN	c28		c28		c28		-		-		c29		
19	futureAutoRestoreSource	{swmf-att 13}	CHOICE	c28		c28		c28		-		-		c29		
20	identityOfCreator	{swmf-att 14}	GraphicString	c18		c19		c18		-		-		c18		
21	identityOfLastModifier	{swmf-att 15}	GraphicString	c18		c19		c18		-		-		c18		
22	lastBackupDestination	{swmf-att 16}	CHOICE	c30		c31		c30		-		-		c30		
23	lastBackupTime	{swmf-att 17}	CHOICE	c30		c31		c30		-		-		c30		
24	lastRestoreSource	{swmf-att 18}	CHOICE	c32		c33		c32		-		-		c32		
25	lastRestoreTime	{swmf-att 19}	CHOICE	c32		c33		c32		-		-		c32		
26	nameBinding	{dmi-att 63}	OBJECT IDENTIFIER	o		m		x		-		-		x		
27	noteField	{swmf-att 20}	GraphicString	c34		c34		c34		-		-		c35		
28	objectClass	{dmi-att 65}	CHOICE	m		m		x		-		-		x		
29	operationalState	{dmi-att 35}	ENUMERATED	x		m		x		-		-		x		
30	packages	{dmi-att 66}	SET OF OBJECT IDENTIFIER	o		m		x		x		x		x		

Table D.11 (concluded) – Executable Software Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
31	proceduralStatus	{dmi-att 36}	SET OF INTEGER	o		m		-		-		-		-		
32	softwareId	{m3100-att 38}	CHOICE	o		m		x		-		-		x		
33	usageState	{dmi-att 39}	ENUMERATED	c36		m		c36		-		-		c36		
34	userLabel	{m3100-att 50}	GraphicString	o		o		o		-		-		-		
35	vendorName	{m3100-att 51}	GraphicString	o		o		o		-		-		-		
36	version	{m3100-att 52}	GraphicString	o		o		o		-		-		-		

c3: if D.5.1/1b then x else –.
 c4: if D.5.3/2a and D.5.1/1b then x else –.
 c4a: if D.5.3/2a then m else –.
 c5: if D.5.3/21a and D.5.1/1b then x else –.
 c5a: if D.5.3/21a then m else –.
 c6: if D.5.3/3a then o else –.
 c7: if D.5.3/3a then m else –.
 c8: if D.5.3/4a and D.5.1/1b then x else –.
 c9: if D.5.3/4a then m else –.
 c10: if D.5.3/6a and D.5.1/1b then x else –.
 c11: if D.5.3/6a then m else –.
 c12: if D.5.3/8a and D.5.1/1b then x else –.
 c13: if D.5.3/8a then m else –.
 c14: if D.5.3/9a and D.5.1/1b then x else –.
 c15: if D.5.3/9a then m else –.
 c16: if D.5.3/9a and D.5.1/1b then x else –.
 c17: if D.5.3/9a then m else –.
 c18: if D.5.3/9a and D.5.1/1b then x else –.
 c19: if D.5.3/9a then m else –.
 c20: if D.5.3/9a and D.5.1/1b then x else –.
 c21: if D.5.3/9a then m else –.
 c22: if D.5.3/10a and D.5.1/1b then x else –.
 c23: if D.5.3/10a then m else –.
 c24: if D.5.3/10a and D.5.1/1b then x else –.
 c25: if D.5.3/10a then m else –.
 c26: if D.5.3/11a then m else –.
 c27: if D.5.3/11a and D.5.1/1b then x else –.
 c28: if D.5.3/12a then m else –.
 c29: if D.5.3/12a and D.5.1/1b then x else –.
 c30: if D.5.3/13a and D.5.1/1b then x else –.
 c31: if D.5.3/13a then m else –.
 c32: if D.5.3/14a and D.5.1/1b then x else –.
 c33: if D.5.3/14a then m else –.
 c34: if D.5.3/16a then m else –.
 c35: if D.5.3/16a and D.5.1/1b then x else –.
 c36: if D.5.1/1b then x else –.

D.5.4 Attribute groups

There are no attribute groups specified for this managed object class.

D.5.5 Actions

Table D.12 – Executable Software Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	backup	{swmf-act 7}	softwareProcessingFailureParameter	c1			1.1	BackupArgument	Information Syntax SEQUENCE	c1		
							1.1.1	restoreSource	CHOICE	c:m		
							1.1.1.1	localObject	Object Instance	c:o.1		
							1.1.1.2	inLine	BIT STRING	c:o.1		
							1.1.1.3	offLine	GraphicString	c:o.1		
							1.1.2	additionalInfo	SET OF ANY	c:o		
							1.2	BackupReply	Reply Syntax SEQUENCE	c1		
							1.2.1	reply	CHOICE	c:m		
							1.2.1.1	success	NULL	c:o.2		
							1.2.1.2	inLine	BIT STRING	c:o.2		
							1.2.2	additionalInfo	SET OF ANY	c:o		
2	executeProgram	{swmf-act 2}	softwareProcessingFailureParameter	c7			2.1	executeProgramInfo	Information Syntax SET OF SEQUENCE	c7		
							2.1.2	significance	BOOLEAN	c:o		
							2.1.3	information	ANY DEFINED BY identifier	c:m		
							2.2	ExecuteProgramReply	Reply Syntax SEQUENCE	c7		
							2.2.1	processId	INTEGER	c:m		
							2.2.2	processOwner	GraphicString	c:m		
							2.2.3	startTime	GeneralizedTime	c:m		
							2.2.4	additionalInfo	SET OF SEQUENCE	c:o		
							2.2.4.1	identifier	OBJECT IDENTIFIER	c:m		
							2.2.4.2	significance	BOOLEAN	c:o		
							2.2.4.3	information	ANY DEFINED BY identifier	c:m		
3	install	{swmf-act 3}	softwareProcessingFailureParameter	c2			3.1	InstallInfo	Information Syntax SEQUENCE	c2		
							3.1.1	targetSoftware	SET OF CHOICE	c:m		
							3.1.1.1	distributedSoftwareId	GraphicString	c:o.1		

Table D.12 (continued) – Executable Software Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							3.1.1.2	distributedSoftwarePointer	CHOICE	c:o.1		
							3.1.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.2		
							3.1.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							3.1.1.2.1.2	AttributeValue	ANY	c:m		
							3.1.1.2.2	nonSpecificForm	OCTET STRING	c:o.2		
							3.1.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.2		
							3.1.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							3.1.1.2.3.2	AttributeValue	ANY	c:m		
							3.1.2	installInfo	SET OF SEQUENCE	c:m		
							3.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							3.1.2.2	significance	BOOLEAN	c:o		
							3.1.2.3	information	ANY DEFINED BY identifier	c:m		
4	restore	{swmf-act 8}		c3			4.1	RestoreArgument	Information Syntax SEQUENCE	c3		
							4.1.1	restoreSource	CHOICE	c:m		
							4.1.1.1	localObject	Object Instance	c:o.1		
							4.1.1.2	inLine	BIT STRING	c:o.1		
							4.1.1.3	offLine	Printable String	c:o.1		
							4.1.2	additionalInfo	SET OF ANY	c:o		
5	revert	{swmf-act 4}	softwareProcessingFailureParameter	c4			5.1	RevertArgument	Information Syntax SEQUENCE	c4		
							5.1.1	revertInfo	SEQUENCE OF CHOICE	c:m		
							5.1.1.1	patchId	GraphicString	c:o.3		
							5.1.1.2	patchPointer	CHOICE	c:o.3		
							5.1.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.4		
							5.1.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		

Table D.12 (continued) – Executable Software Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
							5.1.1.2.1.2	AttributeValue	ANY	c:m		
							5.1.1.2.2	nonSpecificForm	OCTET STRING	c:o.4		
							5.1.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.4		
							5.1.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							5.1.1.2.3.2	AttributeValue	ANY	c:m		
							5.1.2	additionalInfo	SET OF SEQUENCE	c:o		
							5.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							5.1.2.2	significance	BOOLEAN	c:o		
							5.1.2.3	information	ANY DEFINED BY identifier	c:m		
							5.2	RevertReply	Reply Syntax SEQUENCE	c4		
							5.2.1	appliedPatches	SEQUENCE OF CHOICE	c:m		
							5.2.1.1	patchId	GraphicString	c:o.5		
							5.2.1.2	patchPointer	CHOICE	c:o.5		
							5.2.1.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.6		
							5.2.1.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							5.2.1.2.1.2	AttributeValue	ANY	c:m		
							5.2.1.2.2	nonSpecificForm	OCTET STRING	c:o.6		
							5.2.1.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.6		
							5.2.1.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							5.2.1.2.3.2	AttributeValue	ANY	c:m		
							5.2.2	additionalInfo	SET OF SEQUENCE	c:o		
							5.2.2.1	identifier	OBJECT IDENTIFIER	c:m		
							5.2.2.2	significance	BOOLEAN	c:o		
							5.2.2.3	information	ANY DEFINED BY identifier	c:m		

Table D.12 (continued) – Executable Software Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
6	terminateValidation	{swmf-act 5}	softwareProcessingFailureParameter	c5			6.1	TerminateValidationArgument	Information Syntax SEQUENCE	c5		
							6.1.1	terminateValidationInfo	ENUMERATED	c:m		
							6.1.2	additionalInfo	SET OF SEQUENCE	c:o		
							6.1.2.1	identifier	OBJECT IDENTIFIER	c:m		
							6.1.2.2	significance	BOOLEAN	c:o		
							6.1.2.3	information	ANY DEFINED BY identifier	c:m		
							6.2	TerminateValidationReply	Reply Syntax CHOICE	c5		
							6.2.1	noOutStandingValidation	NULL	c:o.7		
							6.2.2	validationCancelled	NULL	c:o.7		
							6.2.3	resultOfPartialValidation	CHOICE	c:o.7		
							6.2.3.1	validationTerminated	NULL	c:o.8		
							6.2.3.2	passValidation	NULL	c:o.8		
							6.2.3.3	passValidationWithResult	SET OF SEQUENCE	c:o.8		
							6.2.3.3.1	identifier	OBJECT IDENTIFIER	c:m		
							6.2.3.3.2	significance	BOOLEAN	c:o		
							6.2.3.3.3	information	ANY DEFINED BY identifier	c:m		
							6.2.3.4	failValidation	NULL	c:o.8		
							6.2.3.5	failValidationWithResult	SET OF SEQUENCE	c:o.8		
							6.2.3.5.1	identifier	OBJECT IDENTIFIER	c:m		
							6.2.3.5.2	significance	BOOLEAN	c:o		
							6.2.3.5.3	information	ANY DEFINED BY identifier	c:m		

Table D.12 (concluded) – Executable Software Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
7	validate	{swmf-act 6}	softwareProcessingFailureParameter	c6			7.1	ValidateInfo	Information Syntax CHOICE	c6		
							7.1.1	instanceDefaultValidationType	NULL	c:o.9		
							7.1.2	registeredValidationType	OBJECT IDENTIFIER	c:o.9		
							7.1.3	systemSpecificValidationType	SET OF SEQUENCE	c:o.9		
							7.1.3.1	identifier	OBJECT IDENTIFIER	c:m		
							7.1.3.2	significance	BOOLEAN	c:o		
							7.1.3.3	information	ANY DEFINED BY identifier	c:m		
							7.2	ValidateReply	Reply Syntax CHOICE	c6		
							7.2.1	validationTerminated	NULL	c:o.10		
							7.2.2	passValidation	NULL	c:o.10		
							7.2.3	passValidationWithResult	SET OF SEQUENCE	c:o.10		
							7.2.3.1	identifier	OBJECT IDENTIFIER	c:m		
							7.2.3.2	significance	BOOLEAN	c:o		
							7.2.3.3	information	ANY DEFINED BY identifier	c:m		
							7.2.4	failValidation	NULL	c:o.10		
							7.2.5	failValidationWithResult	SET OF SEQUENCE	c:o.10		
							7.2.5.1	identifier	OBJECT IDENTIFIER	c:m		
							7.2.5.2	significance	BOOLEAN	c:o		
							7.2.5.3	information	ANY DEFINED BY identifier	c:m		
c1: if D.5.3/13a then m else –. c2: if D.5.2/15a then m else –. c3: if D.5.2/14a then m else –. c4: if D.5.2/19a then m else –. c5: if D.5.2/24a then m else –. c6: if D.5.2/28a then m else –. c7: if D.5.2/32a then m else –.												

Table D.13 – Executable Software Notification support

						Support								
Index	Notification type template label	Value of object identifier for notification type	Con-straints and values	Sta-tus	Con-firm-ed	Non-con-firm-ed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Sta-tus	Sup-port	Additional information
1	attributeValueChange	{dmi-not1}		c1				1.1	AttributeVal ueChangeInfo		Information Syntax SEQUENCE	c1		
								1.1.1	sourceIndicat or	{dmi-att 26}	ENUMERATED	c:o		
								1.1.2	attributeIdenti fierList	{dmi-att 8}	SET OF CHOICE	c:o		
								1.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o.1		
								1.1.2.2	localForm	-	INTEGER	c:o.1		
								1.1.3	attributeVal ueChangeDefi nition	{dmi-att 10}	SET OF SEQUENCE	c:m		
								1.1.3.1	attributeID	-	CHOICE	c:m		
								1.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.2		
								1.1.3.1.2	localForm	-	INTEGER	c:o.2		
								1.1.3.2	oldAttributeV alue	-	ANY DEFINED BY attributeID	c:o		
								1.1.3.3	newAttribu teValue	-	ANY DEFINED BY attributeID	c:m		
								1.1.4	notificationId entifier	{dmi-att 16}	INTEGER	c:o		
								1.1.5	correlatedNoti fications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								1.1.5.1	correlatedNoti fications	{dmi-att 12}	SET OF INTEGER	c:m		
								1.1.5.2	sourceObjectI nst	-	CHOICE	c:o		
								1.1.5.2.1	distinguish edName	-	SEQUENCE OF SET OF SEQUENCE	c:o.3		
								1.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								1.1.5.2.1.2	AttributeVal ue	-	ANY	c:m		
								1.1.5.2.2	nonSpecificFo rm	-	OCTET STRING	c:o.3		
								1.1.5.2.3	localDistingui shedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.3		
								1.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		

Table D.13 (continued) – Executable Software Notification support

Support							Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
															1.1.5.2.3.2	AttributeValue	-	ANY	c:m		
															1.1.6	additionalText	{dmi-att 7}	GraphicString	c:o		
															1.1.7	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
															1.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m		
															1.1.7.2	significance	-	BOOLEAN	c:o		
															1.1.7.3	information	-	ANY DEFINED BY identifier	c:m		
2	autoBackupReport	{2 9 2 18 10 1}		c2											2.1	AutoBackupReportInfo		Information Syntax SEQUENCE	c2		
															2.1.1	backupResult	-	CHOICE	c:m		
															2.1.1.1	inLine	-	CHOICE	c:o.4		
															2.1.1.2	local	-	SEQUENCE	c:o.4		
															2.1.1.2.1	destination	-	CHOICE	c:m		
															2.1.1.2.1.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.5		
															2.1.1.2.1.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
															2.1.1.2.1.1.2	AttributeValue	-	ANY	c:m		
															2.1.1.2.1.2	nonSpecificForm	-	OCTET STRING	c:o.5		
															2.1.1.2.1.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.5		
															2.1.1.2.1.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
															2.1.1.2.1.3.2	AttributeValue	-	ANY	c:m		
															2.1.1.2.2	success	-	BOOLEAN	c:m		
															2.1.1.3	offLine	-	SEQUENCE	c:o.4		
															2.1.1.3.1	destination	-	GraphicString	c:m		
															2.1.1.3.2	result	-	CHOICE	c:m		
3	autoRestoreReport	{2 9 2 18 10 2}		c3											3.1	AutoRestoreReportInfo		Information Syntax SEQUENCE	c3		
															3.1.1	source	-	CHOICE	c:m		
															3.1.1.1	localObject	-	CHOICE	c:o.6		

Table D.13 (continued) – Executable Software Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								3.1.1.1.1	distinguish edName	-	SEQUENCE OF SET OF SEQUENCE	c:o.7		
								3.1.1.1.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								3.1.1.1.1.2	AttributeVal ue	-	ANY	c:m		
								3.1.1.1.2	nonSpecificFo rm	-	OCTET STRING	c:o.7		
								3.1.1.1.3	localDistingui shedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.7		
								3.1.1.1.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								3.1.1.1.3.2	AttributeVal ue	-	ANY	c:m		
								3.1.1.2	remoteSystem	-	GraphicString	c:o.6		
								3.1.2	success	-	BOOLEAN	c:m		
								3.1.3	additionalInfo	-	SET OF SEQUENCE	c:o		
								3.1.3.1	identifier	-	OBJECT IDENTIFIER	c:m		
								3.1.3.2	significance	-	BOOLEAN	c:o		
								3.1.3.3	information	-	ANY DEFINED BY identifier	c:m		
4	objectCr eation	{dmi-not6}		c4				4.1	ObjectInfo		Information Syntax SEQUENCE	c4		
								4.1.1	sourceIndicat or	{dmi-att 26}	ENUMERATED	c:o		
								4.1.2	attributeList	{dmi-att 9}	SET OF SEQUENCE	c:o		
								4.1.2.1	attributeId	-	CHOICE	c:m		
								4.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.8		
								4.1.2.1.2	localForm	-	INTEGER	c:o.8		
								4.1.2.2	attributeValue	-	ANY DEFINED BY attributeld	c:m		
								4.1.3	notificationId entifier	{dmi-att 16}	INTEGER	c:o		
								4.1.4	correlatedNoti fications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								4.1.4.1	correlatedNoti fications	{dmi-att 12}	SET OF INTEGER	c:m		
								4.1.4.2	sourceObjectI nst	-	CHOICE	c:o		

Table D.13 (continued) – Executable Software Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								4.1.4.2.1	distinguish edName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9		
								4.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								4.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								4.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o.9		
								4.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9		
								4.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								4.1.4.2.3.2	AttributeValue	-	ANY	c:m		
								4.1.5	additionalText	{ dmi-att 7 }	GraphicString	c:o		
								4.1.6	additionalInformation	{ dmi-att 6 }	SET OF SEQUENCE	c:o		
								4.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		
								4.1.6.2	significance	-	BOOLEAN	c:o		
								4.1.6.3	information	-	ANY DEFINED BY identifier	c:m		
5	objectDeletion	{ dmi-not7 }		c5				5.1	ObjectInfo		Information Syntax SEQUENCE	c5		
								5.1.1	sourceIndicator	{ dmi-att 26 }	ENUMERATED	c:o		
								5.1.2	attributeList	{ dmi-att 9 }	SET OF SEQUENCE	c:o		
								5.1.2.1	attributeId	-	CHOICE	c:m		
								5.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.10		
								5.1.2.1.2	localForm	-	INTEGER	c:o.10		
								5.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								5.1.3	notificationIdentifier	{ dmi-att 16 }	INTEGER	c:o		
								5.1.4	correlatedNotifications	{ dmi-att 12 }	SET OF SEQUENCE	c:o		
								5.1.4.1	correlatedNotifications	{ dmi-att 12 }	SET OF INTEGER	c:m		

Table D.13 (continued) – Executable Software Notification support

							Support							
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
								5.1.4.2	sourceObjectInst	-	CHOICE	c:o		
								5.1.4.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.11		
								5.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								5.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								5.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o.11		
								5.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.11		
								5.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								5.1.4.2.3.2	AttributeValue	-	ANY	c:m		
								5.1.5	additionalText	{dmi-att 7}	GraphicString	c:o		
								5.1.6	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								5.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		
								5.1.6.2	significance	-	BOOLEAN	c:o		
								5.1.6.3	information	-	ANY DEFINED BY identifier	c:m		
6	processingErrorAlarm	{dmi-not10}		m			alarmEffectOnServiceParameter	6.1	AlarmInfo		Information Syntax SEQUENCE	m		
								6.1.1	probableCause	{dmi-att 18}	CHOICE	m		
								6.1.1.1	globalValue	-	OBJECT IDENTIFIER	o.12		
								6.1.1.2	localValue	-	INTEGER	o.12		
								6.1.2	specificProblems	{dmi-att 27}	SET OF CHOICE	o		
								6.1.2.1	OBJECT IDENTIFIER	-	OBJECT IDENTIFIER	c:o.13		
								6.1.2.2	INTEGER	-	INTEGER	c:o.13		
								6.1.3	perceivedSeverity	{dmi-att 17}	ENUMERATED	m		

Table D.13 (continued) – Executable Software Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								6.1.4	backedUpStatus	{dmi-att 11}	BOOLEAN	o		
								6.1.5	backUpObject	{dmi-att 40}	CHOICE	o		
								6.1.5.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 14		
								6.1.5.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.5.1.2	AttributeValue	-	ANY	c:m		
								6.1.5.2	nonSpecificForm	-	OCTET STRING	c:o. 14		
								6.1.5.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 14		
								6.1.5.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.5.3.2	AttributeValue	-	ANY	c:m		
								6.1.6	trendIndication	{dmi-att 30}	ENUMERATED	o		
								6.1.7	thresholdInfo	{dmi-att 29}	SEQUENCE	o		
								6.1.7.1	triggeredThreshold	-	CHOICE	c:m		
								6.1.7.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 15		
								6.1.7.1.2	localForm	-	INTEGER	c:o. 15		
								6.1.7.2	observedValue	-	CHOICE	c:m		
								6.1.7.2.1	integer	-	INTEGER	c:o. 16		
								6.1.7.2.2	real	-	REAL	c:o. 16		
								6.1.7.3	thresholdLevel	-	CHOICE	c:o		
								6.1.7.3.1	up	-	SEQUENCE	c:o. 17		
								6.1.7.3.1.1	high	-	CHOICE	c:m		
								6.1.7.3.1.1.1	integer	-	INTEGER	c:o. 18		
								6.1.7.3.1.1.2	real	-	REAL	c:o. 18		

Table D.13 (continued) – Executable Software Notification support

							Support							
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
								6.1.7.3.1.2	low	-	CHOICE	c:o		
								6.1.7.3.1.2.1	integer	-	INTEGER	c:o.19		
								6.1.7.3.1.2.2	real	-	REAL	c:o.19		
								6.1.7.3.2	down	-	SEQUENCE	c:o.17		
								6.1.7.3.2.1	high	-	CHOICE	c:m		
								6.1.7.3.2.1.1	integer	-	INTEGER	c:o.20		
								6.1.7.3.2.1.2	real	-	REAL	c:o.20		
								6.1.7.3.2.2	low	-	CHOICE	c:m		
								6.1.7.3.2.2.1	integer	-	INTEGER	c:o.21		
								6.1.7.3.2.2.2	real	-	REAL	c:o.21		
								6.1.7.4	armTime	-	GeneralizedTime	c:o		
								6.1.8	notificationIdentifier	{dmi-att 16}	INTEGER	o		
								6.1.9	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	o		
								6.1.9.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								6.1.9.2	sourceObjectInst	-	CHOICE	c:o		
								6.1.9.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.22		
								6.1.9.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.9.2.1.2	AttributeValue	-	ANY	c:m		
								6.1.9.2.2	nonSpecificForm	-	OCTET STRING	c:o.22		
								6.1.9.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.22		
								6.1.9.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								6.1.9.2.3.2	AttributeValue	-	ANY	c:m		
								6.1.10	stateChangeDefinition	{dmi-att 28}	SET OF SEQUENCE	o		

Table D.13 (continued) – Executable Software Notification support

							Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information	
								6.1.10.1	attributeID	-	CHOICE	c:m			
								6.1.10.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.23			
								6.1.10.1.2	localForm	-	INTEGER	c:o.23			
								6.1.10.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o			
								6.1.10.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m			
								6.1.11	monitoredAttributes	{dmi-att15}	SET OF SEQUENCE	o			
								6.1.11.1	attributeId	-	CHOICE	c:m			
								6.1.11.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.24			
								6.1.11.1.2	localForm	-	INTEGER	c:o.24			
								6.1.11.2	attributeValue	-	ANY DEFINED BY attributeId	c:m			
								6.1.12	proposedRepairActions	{dmi-att19}	SET OF CHOICE	o			
								6.1.12.1	OBJECT IDENTIFIER	-	OBJECT IDENTIFIER	c:o.25			
								6.1.12.2	INTEGER	-	INTEGER	c:o.25			
								6.1.13	additionalText	{dmi-att7}	GraphicString	o			
								6.1.14	additionalInformation	{dmi-att6}	SET OF SEQUENCE	o			
								6.1.14.1	identifier	-	OBJECT IDENTIFIER	c:m			
								6.1.14.2	significance	-	BOOLEAN	c:o			
								6.1.14.3	information	-	ANY DEFINED BY identifier	c:m			
7	stateChange	{dmi-not14}		c6				7.1	StateChangeInfo		Information Syntax SEQUENCE	c6			
								7.1.1	sourceIndicator	{dmi-att26}	ENUMERATED	c:o			
								7.1.2	attributeIdentifierList	{dmi-att8}	SET OF CHOICE	c:o			
								7.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o.26			
								7.1.2.2	localForm	-	INTEGER	c:o.26			

Table D.13 (concluded) – Executable Software Notification support

						Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
								7.1.3	stateChangeDefinition	{dmi-att 28}	SET OF SEQUENCE	c:m		
								7.1.3.1	attributeID	-	CHOICE	c:m		
								7.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 27		
								7.1.3.1.2	localForm	-	INTEGER	c:o. 27		
								7.1.3.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o		
								7.1.3.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m		
								7.1.4	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								7.1.5	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								7.1.5.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								7.1.5.2	sourceObjectInst	-	CHOICE	c:o		
								7.1.5.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 28		
								7.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								7.1.5.2.1.2	AttributeValue	-	ANY	c:m		
								7.1.5.2.2	nonSpecificForm	-	OCTET STRING	c:o. 28		
								7.1.5.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 28		
								7.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								7.1.5.2.3.2	AttributeValue	-	ANY	c:m		
								7.1.6	additionalText	{dmi-att 7}	GraphicString	c:o		
								7.1.7	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								7.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m		
								7.1.7.2	significance	-	BOOLEAN	c:o		
								7.1.7.3	information	-	ANY DEFINED BY identifier	c:m		
c1: if D.5.3/5a then m else –. c2: if D.5.3/11a then m else –. c3: if D.5.3/12a then m else –. c4: if D.5.3/7a then m else –. c5: if D.5.3/7a then m else –. c6: if D.5.3/23a then m else –.														

D.5.7 Parameters**Table D.14 – Executable Software Parameter support**

Index	Parameter template label	Value of object identifier for Parameter	Constraints and values	Status	Support	Additional information
1	alarmEffectOnServiceParameter	{2 9 2 18 5 1}	EVENT-INFO processingErrorAlarm	m		
2	softwareProcessingFailureParameter	{2 9 2 18 5 2}	SPECIFIC-ERROR	c1		
c1: if D.5.3/13a or D.5.3/15a or D.5.3/19a or D.5.3/24a or D.5.3/28a or D.5.3/32a then m else –.						

D.6 Software distributor managed object class**D.6.1 Statement of conformance to the managed object class**

The supplier of the implementation shall state whether or not all mandatory features of the software distributor managed object class are supported, and if the actual class supported is the same as the managed object class to which conformance is claimed.

Table D.15 – Software Distributor Managed object class support

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

If the answer to the actual class question in Table D.15 is “N”, the supplier of the implementation shall supply the actual class support details, in Table D.17.

Table D.16 – Software Distributor Actual class support

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

Table D.17 – Software Distributor Package support

Index	Package template label	Value of object identifier for package	Constraints and values	Status	Support	Additional information
1	allomorphicPackage	{dmi-pkg 17}	“if an object supports allomorphism”	c1		
2	packagesPackage	{dmi-pkg 16}	“any registered package, other than this package has been instantiated”.	c2		
3	softwareDistributorPackage	(not registered)	Mandatory	m		
4	topPackage	(not registered)	Mandatory	m		
c1: if D.6.1/1b then – else m. c2: if D.6.2/2a then m else –.						

D.6.3 Attributes

Table D.18 – Software Distributor Attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get		Replace		Add		Remove		Set to default		Additional information
				Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	Sta-tus	Sup-port	
1	administrativeState	{dmi-att 31}		m		m		m		-		-		c1		
2	allomorphs	{dmi-att 50}	SET OF CHOICE	c2		c3		-		-		-		-		
3	nameBinding	{dmi-att 63}	OBJECT IDENTIFIER	o		m		x		-		-		x		
4	objectClass	{dmi-att 65}	CHOICE	m		m		x		-		-		x		
5	operationalState	{dmi-att 35}	ENUMERATED	x		m		x		-		-		x		
6	packages	{dmi-att 66}	SET OF OBJECT IDENTIFIER	o		c1		x		x		x		x		
7	softwareDistributorId	{swmf-att 21}	NameType	o		m		x		-		-		x		
c1: if D.6.1/1b then x else –. c2: if D.6.2/1a then o else –. c3: if D.6.2/1a then m else –.																

D.6.4 Attribute groups

There are no attribute groups specified for this managed object class.

D.6.5 Actions

Table D.19 – Software Distributor Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	deliver	{swmf-act 1}	softwareProcessingFailureParameter	m			1.1	DeliverInfo	Information Syntax SEQUENCE	m		
							1.1.1	deliverId	CHOICE	o		
							1.1.1.1	globalValue	OBJECT IDENTIFIER	c:o.1		
							1.1.1.2	localValue	INTEGER	c:o.1		
							1.1.2	targetSoftware	SET OF CHOICE	m		
							1.1.2.1	distributedSoftwareId	GraphicString	o.2		
							1.1.2.2	distributedSoftwarePointer	CHOICE	o.2		
							1.1.2.2.1	distinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.3		
							1.1.2.2.1.1	AttributeType	OBJECT IDENTIFIER	c:m		
							1.1.2.2.1.2	AttributeValue	ANY	c:m		
							1.1.2.2.2	nonSpecificForm	OCTET STRING	c:o.3		
							1.1.2.2.3	localDistinguishedName	SEQUENCE OF SET OF SEQUENCE	c:o.3		
							1.1.2.2.3.1	AttributeType	OBJECT IDENTIFIER	c:m		
							1.1.2.2.3.2	AttributeValue	ANY	c:m		
							1.1.3	targetSystem	CHOICE	o		
							1.1.3.1	single	AE-title	c:o.4		
							1.1.3.2	multiple	SET OF AE-title	c:o.4		
							1.1.4	transferInfo	SEQUENCE	o		
							1.1.4.1	transferProtocol	CHOICE	c:m		
							1.1.4.1.1	globalValue	OBJECT IDENTIFIER	c:o.5		
							1.1.4.1.2	localValue	INTEGER	c:o.5		
							1.1.4.2	protocolSpecificInfo	ANY DEFINED BY transferProtocol	c:o		
							1.1.5	additionalInfo	SET OF SEQUENCE	o		
							1.1.5.1	identifier	OBJECT IDENTIFIER	c:m		
							1.1.5.2	significance	BOOLEAN	c:o		
							1.1.5.3	information	ANY DEFINED BY identifier	c:m		

Table D.20 – Software Distributor Notification support

							Support							
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
1	deliverResultNotification	{swmf-not3}		m				1.1	DeliverResultInfo		Information Syntax SEQUENCE	m		
								1.1.1	deliverId	-	CHOICE	o		
								1.1.1.1	globalValue	-	OBJECT IDENTIFIER	c:o.1		
								1.1.1.2	localValue	-	INTEGER	c:o.1		
								1.1.2	deliverResult	-	ENUMERATED	m		
								1.1.3	additionalInfo	-	SET OF SEQUENCE	o		
								1.1.3.1	identifier	-	OBJECT IDENTIFIER	c:m		
								1.1.3.2	significance	-	BOOLEAN	c:o		
								1.1.3.3	information	-	ANY DEFINED BY identifier	c:m		
2	objectCreation	{dmi-not6}		m				2.1	ObjectInfo		Information Syntax SEQUENCE	m		
								2.1.1	sourceIndicator	{dmi-att26}	ENUMERATED	c:o		
								2.1.2	attributeList	{dmi-att9}	SET OF SEQUENCE	c:o		
								2.1.2.1	attributeId	-	CHOICE	c:m		
								2.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o.8		
								2.1.2.1.2	localForm	-	INTEGER	c:o.8		
								2.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								2.1.3	notificationIdentifier	{dmi-att16}	INTEGER	c:o		
								2.1.4	correlatedNotifications	{dmi-att12}	SET OF SEQUENCE	c:o		
								2.1.4.1	correlatedNotifications	{dmi-att12}	SET OF INTEGER	c:m		
								2.1.4.2	sourceObjectInst	-	CHOICE	c:o		
								2.1.4.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9		
								2.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								2.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								2.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o.9		
								2.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o.9		
								2.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								2.1.4.2.3.2	AttributeValue	-	ANY	c:m		

Table D.20 (continued) – Software Distributor Notification support

Support							Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed								
								2.1.5	additionalText	{dmi-att 7}	GraphicString	c:o		
								2.1.6	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								2.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		
								2.1.6.2	significance	-	BOOLEAN	c:o		
								2.1.6.3	information	-	ANY DEFINED BY identifier	c:m		
3	objectDeletion	{dmi-not7}		m				3.1	ObjectInfo		Information Syntax SEQUENCE	m		
								3.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o		
								3.1.2	attributeList	{dmi-att 9}	SET OF SEQUENCE	c:o		
								3.1.2.1	attributeId	-	CHOICE	c:m		
								3.1.2.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 10		
								3.1.2.1.2	localForm	-	INTEGER	c:o. 10		
								3.1.2.2	attributeValue	-	ANY DEFINED BY attributeId	c:m		
								3.1.3	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								3.1.4	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								3.1.4.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								3.1.4.2	sourceObjectInst	-	CHOICE	c:o		
								3.1.4.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 11		
								3.1.4.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								3.1.4.2.1.2	AttributeValue	-	ANY	c:m		
								3.1.4.2.2	nonSpecificForm	-	OCTET STRING	c:o. 11		
								3.1.4.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 11		
								3.1.4.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								3.1.4.2.3.2	AttributeValue	-	ANY	c:m		
								3.1.5	additionalText	{dmi-att 7}	GraphicString	c:o		
								3.1.6	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								3.1.6.1	identifier	-	OBJECT IDENTIFIER	c:m		
								3.1.6.2	significance	-	BOOLEAN	c:o		
								3.1.6.3	information	-	ANY DEFINED BY identifier	c:m		

Table D.20 (concluded) – Software Distributor Notification support

						Support								
Index	Notification type template label	Value of object identifier for notification type	Constraints and values	Status	Confirmed	Non-confirmed	Additional information	Subindex	Notification field name label	Value of object identifier of attribute type associated with field	Constraints and values	Status	Support	Additional information
4	stateChange	{dmi-not14}		m				4.1	StateChangeInfo		Information Syntax SEQUENCE	m		
								4.1.1	sourceIndicator	{dmi-att 26}	ENUMERATED	c:o		
								4.1.2	attributeIdentifierList	{dmi-att 8}	SET OF CHOICE	c:o		
								4.1.2.1	globalForm	-	OBJECT IDENTIFIER	c:o. 26		
								4.1.2.2	localForm	-	INTEGER	c:o. 26		
								4.1.3	stateChangeDefinition	{dmi-att 28}	SET OF SEQUENCE	c:m		
								4.1.3.1	attributeID	-	CHOICE	c:m		
								4.1.3.1.1	globalForm	-	OBJECT IDENTIFIER	c:o. 27		
								4.1.3.1.2	localForm	-	INTEGER	c:o. 27		
								4.1.3.2	oldAttributeValue	-	ANY DEFINED BY attributeID	c:o		
								4.1.3.3	newAttributeValue	-	ANY DEFINED BY attributeID	c:m		
								4.1.4	notificationIdentifier	{dmi-att 16}	INTEGER	c:o		
								4.1.5	correlatedNotifications	{dmi-att 12}	SET OF SEQUENCE	c:o		
								4.1.5.1	correlatedNotifications	{dmi-att 12}	SET OF INTEGER	c:m		
								4.1.5.2	sourceObjectInst	-	CHOICE	c:o		
								4.1.5.2.1	distinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 28		
								4.1.5.2.1.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								4.1.5.2.1.2	AttributeValue	-	ANY	c:m		
								4.1.5.2.2	nonSpecificForm	-	OCTET STRING	c:o. 28		
								4.1.5.2.3	localDistinguishedName	-	SEQUENCE OF SET OF SEQUENCE	c:o. 28		
								4.1.5.2.3.1	AttributeType	-	OBJECT IDENTIFIER	c:m		
								4.1.5.2.3.2	AttributeValue	-	ANY	c:m		
								4.1.6	additionalText	{dmi-att 7}	GraphicString	c:o		
								4.1.7	additionalInformation	{dmi-att 6}	SET OF SEQUENCE	c:o		
								4.1.7.1	identifier	-	OBJECT IDENTIFIER	c:m		
								4.1.7.2	significance	-	BOOLEAN	c:o		
								4.1.7.3	information	-	ANY DEFINED BY identifier	c:m		

D.6.7 Parameters

There are no parameters specified for this managed object class.

Anexo E⁶⁾**Formulario de MRCS para vinculación de nombre**

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

E.1 Introduction

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

E.2 Instructions for completing the MRCS proforma for name binding to produce an MRCS

The MRCS proforma contained in this annex is comprised of information in tabular form, in accordance with ITU-T Rec. X.724 | ISO/IEC 10165-6. The supplier of the implementation shall state which items are supported in the tables below and if necessary provide additional information.

Notations used in the Status and Support columns are specified in A.1.3.

E.3 Symbols, abbreviations and terms

The following abbreviation is used in this MRCS proforma:

dmi-nb joint-iso-ccitt ms(9) smi(3) part2(2) nameBinding(6)

E.4 Statement of conformance to the name binding**Table E.1 – Name Binding support**

Index	Name binding template label	Value of object identifier for name binding	Constraints and values	Status	Support	Additional information
1	softwareUnit-subsystem	{2 9 2 18 6 2}	Superior class: “Rec. X.723 ISO/IEC 10165-5:1994”: subsystem AND SUBCLASSES	o		
2	softwareUnit-system	{2 9 2 18 6 4}	Superior class: “Rec. X.721 ISO/IEC 10165-2:1992”: system AND SUBCLASSES	o		
3	software-equipment	{0 0 13 3100 0 6 18}	Superior class: “Rec. M.3100:1995”: equipment AND SUBCLASSES	o		
4	software-managedElement	{0 0 13 3100 0 6 20}	Superior class: “Rec. M.3100:1995”: managedElement AND SUBCLASSES	o		
5	software-software	{0 0 13 3100 0 6 19}	Superior class: “Rec. M.3100:1995”: software AND SUBCLASSES	o		

6) Comunicado sobre derechos de autor del formulario de MRCS

Los usuarios de esta Recomendación | Norma Internacional pueden reproducir libremente el formulario de MRCS de este anexo a fin de que pueda ser utilizado para los fines previstos, y pueden además publicar el MRCS cumplimentado. En la Rec. UIT-T X.724 | ISO/CEI 10165-6 se especifican las instrucciones para rellenar el formulario de MRCS.

Table E.1 (concluded) – Name Binding support

Subindex	Operation	Constraints and values	Status	Support	Additional information
1.1	Create support		c:m		
1.2	Create with reference object		-		
1.3	Create with automatic instance naming		c:m		
1.4	Delete support		c:m		
1.5	Delete only if no contained objects		c:x		
1.6	Delete contained objects		c:m		
2.1	Create support		c:m		
2.2	Create with reference object		-		
2.3	Create with automatic instance naming		c:m		
2.4	Delete support		c:m		
2.5	Delete only if no contained objects		c:x		
2.6	Delete contained objects		c:m		
3.1	Create support		c:m		
3.2	Create with reference object		c:m		
3.3	Create with automatic instance naming		c:m		
3.4	Delete support		c:m		
3.5	Delete only if no contained objects		c:m		
3.6	Delete contained objects		c:x		
4.1	Create support		c:m		
4.2	Create with reference object		c:m		
4.3	Create with automatic instance naming		c:m		
4.4	Delete support		c:m		
4.5	Delete only if no contained objects		c:m		
4.6	Delete contained objects		c:x		
5.1	Create support		c:m		
5.2	Create with reference object		c:m		
5.3	Create with automatic instance naming		c:m		
5.4	Delete support		c:m		
5.5	Delete only if no contained objects		c:m		
5.6	Delete contained objects		c:x		

Anexo F⁷⁾

Formulario de MIDS

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

F.1 Actions

Table F.1 – Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	backup	{swmf-act 7}	softwareProcessingFailureParameter	o			1.1	BackupArgument	Information Syntax SEQUENCE	c:m		
							1.1.1	backupDestination	CHOICE	c:m		
							1.1.1.1	localObject	Object Instance	c:o.1		
							1.1.1.2	inLine	NULL	c:o.1		
							1.1.1.3	offLine	GraphicString	c:o.1		
							1.1.2	additionalInfo	SET OF ANY	c:o		
							1.2	BackupReply	Reply Syntax SEQUENCE	c:m		
							1.2.1	reply	CHOICE	c:m		
							1.2.1.1	success	NULL	c:o.2		
							1.2.1.2	inLine	BIT STRING	c:o.2		
							1.2.2	additionalInfo	SET OF ANY	c:o		
2	restore	{swmf-act 8}		o			2.1	RestoreArgument	Information Syntax SEQUENCE	c:m		
							2.1.1	restoreSource	CHOICE	c:m		
							2.1.1.1	localObject	Object Instance	c:o.1		
							2.1.1.2	inLine	BIT STRING	c:o.1		
							2.1.1.3	offLine	Printable String	c:o.1		
							2.1.2	additionalInfo	SET OF ANY	c:o		

⁷⁾ Comunicado sobre derechos de autor del formulario de MIDS

Los usuarios de esta Recomendación | Norma Internacional pueden reproducir libremente el formulario de MIDS de este anexo a fin de que pueda ser utilizado para los fines previstos, y pueden además publicar el MIDS cumplimentado. En la Rec. UIT-T X.724 | ISO/CEI 10165-6 se especifican las instrucciones para rellenar el formulario de MIDS.

Clase de objeto soporte lógico de la Recomendación M.3100

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

NOTA – Este anexo repite las plantillas GDMO asociadas con la clase de objeto soporte lógico especificadas en la Recomendación M.3100. Las plantillas de la Recomendación M.3100 se incluyen en esta Especificación sólo a título de información. Las plantillas asociadas con la clase de objeto soporte lógico que se especifican en la Rec. X.721 del CCITT | ISO/CEI 10165-2 no se incluyen aquí.

G.1 Software Object Class**software MANAGED OBJECT CLASS****DERIVED FROM "Recommendation X.721:1992":top;****CHARACTERIZED BY softwarePackage PACKAGE****BEHAVIOUR softwareBehaviour ;****ATTRIBUTES****softwareId****GET;;;****CONDITIONAL PACKAGES****createDeleteNotificationsPackage****PRESENT IF "the objectCreation and objectDeletion notifications defined in Recommendation X.721 are supported by an instance of this class.",****attributeValueChangeNotificationPackage****PRESENT IF "the attributeValueChange notification defined in****Recommendation X.721 is supported by an instance of this class.",****stateChangeNotificationPackage****PRESENT IF "the stateChange notification defined in****Recommendation X.721 is supported by an instance of this class.",****administrativeOperationalStatesPackage****PRESENT IF "an instance supports it.",****affectedObjectListPackage****PRESENT IF "an instance supports it.",****softwareProcessingErrorAlarmPackage****PRESENT IF "an instance supports it.",****userLabelPackage****PRESENT IF "an instance supports it",****vendorNamePackage****PRESENT IF "an instance supports it",****versionPackage****PRESENT IF "an instance supports it",****currentProblemListPackage****PRESENT IF "an instance supports it";****REGISTERED AS {m3100ObjectClass 4};****softwareBehaviour BEHAVIOUR****DEFINED AS**

"The Software object class is a class of managed objects that represent logical information stored in equipment, including programs and data tables. Software may be nested within other software, thereby creating a containment relationship. When the attribute value change notification package is present, the attributeValueChange notification defined in Recommendation X.721 shall be emitted when the value of one of the following attribute changes: alarm status, affected object list, user label, version, and current problem list. Because the above attributes are all in conditional packages, the behaviour for emitting the attribute value change notification applies only when the corresponding conditional packages are present in the managed object. When the state change notification package is present, the stateChangeNotification defined in Recommendation X.721 shall be emitted if the value of administrative state or operational state changes (when the administrativeOperationalStates conditional package is present).";

G.2 Packages**G.2.1 Administrative Operational States Package****administrativeOperationalStatesPackage PACKAGE****ATTRIBUTES****"Recommendation X.721:1992":administrativeState****GET-REPLACE,****"Recommendation X.721:1992":operationalState****GET;****REGISTERED AS {m3100Package 1};**

G.2.2 Affected Object List Package

affectedObjectListPackage PACKAGE

ATTRIBUTES

affectedObjectList

GET;

REGISTERED AS {m3100Package 2};

G.2.3 Attribute Value Change Notification Package

attributeValueChangeNotificationPackage PACKAGE

NOTIFICATIONS

"Recommendation X.721:1992":attributeValueChange;

REGISTERED AS {m3100Package 4};

G.2.4 Create Delete Notifications Package

createDeleteNotificationsPackage PACKAGE

NOTIFICATIONS

"Recommendation X.721:1992":objectCreation,

"Recommendation X.721:1992":objectDeletion;

REGISTERED AS {m3100Package 10};

G.2.5 Current Problem List Package

currentProblemListPackage PACKAGE

ATTRIBUTES

currentProblemList

GET;

REGISTERED AS {m3100Package 13};

softwareProcessingErrorAlarmPackage PACKAGE

ATTRIBUTES

alarmStatus

GET;

NOTIFICATIONS

"Recommendation X.721:1992":processingErrorAlarm;

REGISTERED AS {m3100Package 26};

G.2.6 Software Processing Error Alarm Package

softwareProcessingErrorAlarmPackage PACKAGE

ATTRIBUTES

alarmStatus

GET;

NOTIFICATIONS

"Recommendation X.721:1992":processingErrorAlarm;

REGISTERED AS {m3100Package 26};

G.2.7 State Change Notification Package

stateChangeNotificationPackage PACKAGE

NOTIFICATIONS

"Recommendation X.721:1992":stateChange;

REGISTERED AS {m3100Package 28};

G.2.8 User Label Package

userLabelPackage PACKAGE

ATTRIBUTES

userLabel

GET-REPLACE;

REGISTERED AS {m3100Package 32};

G.2.9 Vendor Name Package

vendorNamePackage PACKAGE

ATTRIBUTES

vendorName

GET-REPLACE;

REGISTERED AS {m3100Package 33};

versionPackage PACKAGE

ATTRIBUTES

version

GET-REPLACE;

REGISTERED AS {m3100Package 34};

G.3 Attributes

G.3.1 Affected Object List

affectedObjectList ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.ObjectList;

MATCHES FOR EQUALITY, SET-COMPARISON, SET-INTERSECTION;

BEHAVIOUR affectedObjectListBehaviour ;

REGISTERED AS {m3100Attribute 2};

-- ObjectList ::= SET OF ObjectInstance

affectedObjectListBehaviour

BEHAVIOUR

DEFINED AS

"The Affected Object List attribute type specifies the object instances which can be directly affected by a change in state or deletion of a given managed object. The attribute does not force internal details to be specified, but only the necessary level of detail required for management.";

affectedObjectListPackage PACKAGE

ATTRIBUTES

affectedObjectList

GET;

REGISTERED AS {m3100Package 2};

G.3.2 Alarm Status

alarmStatus ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.AlarmStatus;

MATCHES FOR EQUALITY;

BEHAVIOUR alarmStatusBehaviour;

REGISTERED AS {m3100Attribute 6};

*-- AlarmStatus ::= SET OF INTEGER { underRepair(0), critical(1), major(2), minor(3),**-- alarmOutstanding(4) }*

alarmStatusBehaviour BEHAVIOUR

DEFINED AS

"The Alarm Status attribute type indicates the occurrence of an abnormal condition relating to an object. This attribute may also function as a summary indicator of alarm conditions associated with a specific resource. It is used to indicate the existence of an alarm condition, a pending alarm condition such as threshold situations, or (when used as a summary indicator) the highest severity of active alarm conditions. When used as a summary indicator, the order of severity (from highest to lowest) is: activeReportable-Critical activeReportable-Major activeReportable-Minor activeReportable-Indeterminate activeReportable-Warning activePending cleared";

attributeValueChangeNotificationPackage PACKAGE

NOTIFICATIONS

"Recommendation X.721:1992":attributeValueChange;

REGISTERED AS {m3100Package 4};

G.3.3 Current Problem List

currentProblemList ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.CurrentProblemList;

BEHAVIOUR currentProblemListBehaviour;

REGISTERED AS {m3100Attribute 17};

-- CurrentProblem ::= SEQUENCE {
-- problem [0] ProbableCause,
-- alarmStatus [1] AlarmStatus}
-- CurrentProblemList ::= SET OF CurrentProblem

```

-- AlarmStatus ::= ENUMERATED {
--     cleared(0),
--     activeReportable-Indeterminate(1),
--     activeReportable-Warning(2),
--     activeReportable-Minor(3),
--     activeReportable-Major(4),
--     activeReportable-Critical(5),
--     activePending(6)}
-- ProblemCause ::= CHOICE {
--     unknown NULL,
--     integerValue INTEGER}

-- The values of integer value for ProblemCause and integerValue for ResourceProblem shall always be
-- assigned by this Recommendation. No values of integerValue for ResourceProblem have been assigned.
-- The following values are used for integerValue of ProblemCause.

-- noSuchTpInstance ProblemCause ::= integerValue : 0
-- noSuchGtpInstance ProblemCause ::= integerValue : 1
-- noSuchTpPoolInstance ProblemCause ::= integerValue : 2
-- mismatchingTpInstance ProblemCause ::= integerValue : 3
-- mismatchingGtpInstance ProblemCause ::= integerValue : 4
-- partOfGtp ProblemCause ::= integerValue : 5
-- involvedInCrossConnection ProblemCause ::= integerValue : 6
-- memberOfTpPool ProblemCause ::= integerValue : 7
-- alreadyMemberOfGtp ProblemCause ::= integerValue : 8
-- noTpInTpPool ProblemCause ::= integerValue : 9
-- noMoreThanOneTpIsAllowed ProblemCause ::= integerValue : 10
-- noMoreThanTwoTpsAreAllowed ProblemCause ::= integerValue : 11

```

currentProblemListBehaviour**BEHAVIOUR****DEFINED AS**

"The Current Problem List attribute type identifies the current existing problems, with severity, associated with the managed object.";

currentProblemListPackage PACKAGE

ATTRIBUTES

currentProblemList

GET;

REGISTERED AS {m3100Package 13};

G.3.4 Software Id**softwareId ATTRIBUTE**

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.NameType;

MATCHES FOR EQUALITY;

BEHAVIOUR softwareIdBehaviour;

REGISTERED AS {m3100Attribute 38};

```

-- NameType ::= CHOICE {
--     numericName INTEGER,
--     pString GraphicString}

```

softwareIdBehaviour BEHAVIOUR**DEFINED AS**

"The Software Id is an attribute type whose distinguished value can be used as an RDN when naming an instance of the Software object class.";

softwareProcessingErrorAlarmPackage PACKAGE

ATTRIBUTES

alarmStatus

GET;

NOTIFICATIONS

"Recommendation X.721:1992":processingErrorAlarm;

REGISTERED AS {m3100Package 26};

G.3.5 User Label

userLabel ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.UserLabel;

MATCHES FOR EQUALITY, SUBSTRINGS;

BEHAVIOUR userLabelBehaviour;

REGISTERED AS {m3100Attribute 50};

-- *UserLabel ::= GraphicString*

userLabelBehaviour BEHAVIOUR

DEFINED AS

"The User Label attribute type assigns a user friendly name to the associated object.";

userLabelPackage PACKAGE

ATTRIBUTES

userLabel

GET-REPLACE;

REGISTERED AS {m3100Package 32};

G.3.6 Vendor Name

vendorName ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.VendorName;

MATCHES FOR EQUALITY, SUBSTRINGS;

BEHAVIOUR vendorNameBehaviour;

REGISTERED AS {m3100Attribute 51};

-- *VendorName ::= GraphicString*

vendorNameBehaviour BEHAVIOUR

DEFINED AS

"The Vendor Name attribute type identifies the vendor of the associated managed object.";

vendorNamePackage PACKAGE

ATTRIBUTES

vendorName

GET-REPLACE;

REGISTERED AS {m3100Package 33};

G.3.7 Version

version ATTRIBUTE

WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.Version;

MATCHES FOR EQUALITY, SUBSTRINGS;

BEHAVIOUR versionBehaviour;

REGISTERED AS {m3100Attribute 52};

-- *Version ::= GraphicString*

versionBehaviour BEHAVIOUR

DEFINED AS

"The Version attribute type identifies the version of the associated managed object.";

versionPackage PACKAGE

ATTRIBUTES

version

GET-REPLACE;

REGISTERED AS {m3100Package 34};

Anexo H

Correspondencia entre el modelo POSIX y el modelo de gestión de soporte lógico

(Este anexo no es parte integrante de esta Recomendación | Norma Internacional)

El modelo POSIX consta de los siguientes objetos:

- **Distribución:** Una distribución consta de uno o más productos, en una forma pronta para la instalación. La distribución de soporte lógico existe al nivel más alto. Una distribución puede considerarse un conjunto de productos que puede estar contenido en un medio de distribución o formar parte de la memoria de ficheros de un sistema. La distribución puede no estar constreñida a las líneas de productos y de arquitectura.
- **Soporte lógico instalado:** Un objeto soporte lógico instalado consta de uno o más productos, en una forma pronta para la utilización. Para obtener un objeto soporte lógico instalado, un administrador instala uno o más productos de una distribución y configura estos productos. El soporte lógico instalado es un nivel equivalente al de distribución, pero después de que los productos del soporte lógico han sido instalados en un sistema de ficheros. Adviértase que el proceso de configuración cae fuera del alcance del documento POSIX.
- **Producto:** El nivel de producto es la agrupación más alta para el correspondiente soporte lógico. El nivel de producto define atributos que describen el sistema (o sistemas) operativo y la arquitectura (o arquitecturas) de soporte lógico que el producto soporta. Estos atributos se aplican a todos los subproductos y conjuntos de ficheros contenidos en el producto. Los productos contienen conjuntos de ficheros, que pueden opcionalmente agruparse en subproductos.
- **Conjunto de ficheros:** Un conjunto de ficheros consta de un grupo de ficheros. El nivel que sigue a productos es el nivel que describe los conjuntos de ficheros. El conjunto de ficheros es la unidad de soporte lógico que es cargada y gestionada por el proveedor de productos de soporte lógico. Es también la unidad más pequeña que puede ser seleccionada para carga por el administrador del sistema. Los conjuntos de ficheros contienen los ficheros que componen realmente el producto. Además, incluyen todos los ficheros de descripción de conjuntos de ficheros y personalizan los guiones utilizados por las herramientas de instalación/actualización y remoción para efectuar la instalación y la desinstalación.
- **Subproducto:** Los subproductos son simplemente una notación conveniente para combinar conjuntos de ficheros. Los subproductos pueden constar de otros subproductos, o de conjuntos de ficheros, o de alguna combinación de éstos, en la medida en que puedan ser resueltos a un grupo de conjuntos de ficheros. Un subproducto no tiene un comportamiento único. Es simplemente un modo conveniente de especificar conjuntos de ficheros. El nivel subproducto es opcional. Los subproductos proporcionan una organización intermedia de conjuntos de ficheros para productos que contienen muchos conjuntos de ficheros. Se soportan múltiples niveles de subproductos; un subproducto puede contener otro subproducto, y un conjunto de ficheros puede pertenecer a más de un subproducto.
- **Guión de control:** Un guión de control es un fichero ejecutable. Los guiones de control proporcionan función para la pre-comprobación, post-instalación, etc. Los guiones de control son los ficheros de descripción de productos que describen los ficheros de conjuntos (ÍNDICE e INFO) así como los guiones de control suministrados por el vendedor. El fichero ÍNDICE describe la estructura de productos para el conjunto de ficheros. El fichero INFO describe los ficheros y sus atributos para el conjunto de ficheros. Los guiones de control incluyen operaciones de pre-comprobación, post-comprobación y de personalización para la instalación y remoción.
- **Fichero:** Los ficheros son el nivel más bajo del objeto dentro de una distribución o una instalación. Además de los ficheros que componen el conjunto de ficheros, hay guiones de control que contienen los guiones de comprobación y personalizan guiones para la instalación y remoción de productos y conjuntos de ficheros. Los ficheros son los ficheros y directorios efectivos que componen el conjunto de ficheros, incluidos los atributos de fichero tales como propietario, grupo y permisos.

- **Haz:** Un haz (bundle) es un objeto soporte lógico utilizado para construir agrupaciones a partir de otros objetos soporte lógico tales como la totalidad o partes de otros haces y productos. Un haz es una agrupación de objetos de soporte lógico. Los haces no contienen objetos de soporte lógico dentro del espacio de nombre del haz, sino que en vez de eso designan objetos de soporte lógico que utilizan atributos. Los haces pueden designar otros haces. Un objeto de soporte lógico puede ser referenciado por más de un haz. El cuadro siguiente establece la correspondencia entre las instrucciones de soporte lógico POSIX y las operaciones de soporte lógico comparables definidas en esta Recomendación | Norma Internacional, si es aplicable.

Cuadro H.1 – Correspondencia entre las instrucciones de soporte lógico POSIX y las operaciones de unidad de soporte lógico

Instrucciones POSIX	Operaciones ISO
swask	Fuera de alcance
swconfig	INSTALACIÓN/utilización
swcopy	CREACIÓN o ENTREGA (nota)
swinstall	INSTALACIÓN
swlist	OBTENCIÓN
swmodify	FIJACIÓN
swpackage	Fuera de alcance
swremove	SUPRESIÓN
swverify	VALIDACIÓN
NOTA – Esta Recomendación Norma Internacional utilizaba la distribución de soporte lógico y la unidad de soporte lógico para la entrega de soporte lógico, mientras que POSIX solamente el soporte lógico de distribución. En POSIX una distribución es un objeto estático que contiene un conjunto de objetos de soporte lógico. En esta Recomendación Norma Internacional, una distribución de soporte lógico es un objeto más dinámico, que controla la distribución de unidades de soporte lógico de una sistema a otro. El concepto es definido por una tarea «copia» o «instalación» en POSIX. Una tarea en POSIX significa genéricamente aplicar un conjunto de objetos de soporte lógico a un conjunto de objetos de sistema deseados.	

El soporte lógico de distribución en POSIX establece correspondencia con una unidad de soporte lógico en la ISO. La unidad de soporte lógico de la ISO establece correspondencia con los objetos producto, conjunto de ficheros y fichero POSIX.

El Cuadro H.2 muestra la correspondencia del objeto de soporte lógico POSIX con las clases de objeto gestionado definidas en esta Recomendación | Norma Internacional. No hay ningún objeto POSIX correspondiente a la funcionalidad de la clase de objeto de distribución de soporte lógico definida en esta Recomendación | Norma Internacional.

Cuadro H.2 – Correspondencia de los objetos de soporte lógico POSIX con las clases de objeto gestionado de soporte lógico

Clases de objeto POSIX	Clases de objeto gestionado
Soporte lógico de distribución	Unidad de soporte lógico
Producto	Unidad de soporte lógico
Conjunto de ficheros	Unidad de soporte lógico
Fichero	Unidad de soporte lógico
Haz	En estudio

NOTA – La clase de objeto de ISO correspondiente a la clase de objeto POSIX queda en estudio. (Podría corresponder a la unidad de soporte lógico.)

El Cuadro H.3 establece la correspondencia entre los atributos POSIX y los atributos de unidad de soporte lógico comparables definidos en esta Recomendación | Norma Internacional, si es aplicable.

Cuadro H.3 – Correspondencia entre los atributos de soporte lógico POSIX y los atributos de unidad de soporte lógico

Atributos de unidad de soporte lógico ISO	Atributos de conjunto de ficheros POSIX (a menos que se indique otra cosa)
SoftwareID	Soporte lógico (un identificador de conjunto de ficheros universalmente único; incluye el prefijo de producto)
UserLabel	Título
VendorName	Título_vendedor (en el producto)
Version	Revisión
CurrentProblem	–
OperationalState	Estado
AdministrativeState	Estado
ProceduralStatus	Estado
AvailabilityStatus	Estado
FileType	Tipo (en el fichero)
FileLocation	Trayecto (en el fichero)
FileSize	Tamaño (en el fichero)
Checksum	Suma de control (en el fichero)
NoteField	Descripción
DateOfCreation	–
IdentityOfCreator	–
DateOfLastModification	Hora m
IdentityOfLastModifier	–
DateDelivered	–
DateInstalled	Hora_creación

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedios
Serie I	Red digital de servicios integrados
Serie J	Transmisiones de señales radiofónicas, de televisión y de otras senales multimedios
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	Mantenimiento: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos y comunicación entre sistemas abiertos
Serie Z	Lenguajes de programación