



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.750

Amendment 1

(10/97)

SERIES X: DATA NETWORKS AND OPEN SYSTEM
COMMUNICATIONS

OSI management – Management functions and ODMA
functions

Information technology – Open Systems
Interconnection – Systems management:
Management knowledge management function

**Amendment 1: Extension for General
Relationship Model**

ITU-T Recommendation X.750 – Amendment 1

(Previously CCITT Recommendation)

ITU-T X-SERIES RECOMMENDATIONS
DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEM INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	X.600–X.629
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems Management framework and architecture	X.700–X.709
Management Communication Service and Protocol	X.710–X.719
Structure of Management Information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, Concurrency and Recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999

For further details, please refer to ITU-T List of Recommendations.

INTERNATIONAL STANDARD 10164-16

ITU-T RECOMMENDATION X.750

**INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION –
SYSTEMS MANAGEMENT: MANAGEMENT KNOWLEDGE
MANAGEMENT FUNCTION**

AMENDMENT 1

Extension for General Relationship Model

Source

The ITU-T Recommendation X.750, Amendment 1 was approved on the 24th of October 1997. The identical text is also published as ISO/IEC International Standard 10164-16.

FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, the ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 1998

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

CONTENTS

	<i>Page</i>
1) Subclause 3.6	1
2) Subclause 3.11	1
3) Clause 4.....	1
4) Clause 6.....	2
5) Subclause 6.3	2
6) Subclauses 6.3, 6.5, 7.1.1.1, 8.1.1.2, Table 8, A.2.1 and A.2.3	2
7) Subclause 7.1.2	2
8) Subclauses 7.1.2, 8.1.2.12, 8.6.2 and A.3.3	2
9) Subclause 7.3	2
10) New subclauses 8.1.2.12 and 8.1.2.13	2
11) Subclause 8.6.2	3
12) Subclause 11.1.1.2	3
13) Subclause 11.1.5.1	3
14) Subclause 11.1.5.2	3
15) Subclause 11.1.6.1	4
16) Subclause 11.2.1.2	4
17) Subclause A.1.....	4
18) Subclause A.2.4.....	4
19) Subclause A.3.1.....	5
20) Subclause A.3.3.....	6
21) Subclause A.3.5.....	7
22) Subclause A.4.1.....	8
23) Subclause A.4.5.....	8
24) Subclause B.1.....	9
25) Subclause B.2.....	9
26) Annexes A to F.....	10
27) Subclause C.1.3.....	11
28) Subclause C.4.....	11
29) Subclause D.3.1.....	11
30) New subclauses E.3.16 and E.3.17.....	12

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

**INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION –
SYSTEMS MANAGEMENT: MANAGEMENT KNOWLEDGE
MANAGEMENT FUNCTION**

AMENDMENT 1

Extension for General Relationship Model

1) Subclause 3.6

Add references to these SMO-defined terms inserted among existing terms in alphabetical order:

- e) managed object class

and renumber the list as from the existing e) which becomes f), to the end.

2) Subclause 3.11

Add references to these GRM-defined terms inserted before and after existing term in alphabetical order:

- binding;
- binding support;
- managed relationship class;
- relationship cardinality;
- relationship management notification;
- relationship management operation;
- role cardinality;
- relationship mapping specification;
- relationship class specification;
- role;
- unbinding;
- unbinding support.

3) Clause 4

Insert the following abbreviation by alphabetical order:

- GRM General relationship model

4) Clause 6

In this clause change:

- *Definition knowledge:* Information on the formal specification of managed object classes, name bindings, etc., e.g. templates for classes, name bindings.

to:

- *Definition knowledge:* Information on the formal specification of managed object classes and managed relationship classes and related specifications, e.g. templates for classes, attributes, name bindings.

Add an X in the Table 1 cell in the Definition knowledge column in the Relationship knowledge row.

5) Subclause 6.3

In the 2nd item:

relationship role bindings

to:

relationship mapping specifications

6) Subclauses 6.3, 6.5, 7.1.1.1, 8.1.1.2, Table 8, A.2.1 and A.2.3

*In subclauses 6.3, 6.5, 7.1.1.1, 8.1.1.2, Table 8, and in behaviour text in A.2.1 and A.2.3, replace each occurrence of the term relationship class that is not preceded by managed with the term defined in GRM managed relationship class but do not change the label **supportedRelationshipClassList**.*

7) Subclause 7.1.2

In this subclause change GDMO and ASN.1 specifications to GDMO, GRM and ASN.1 specifications.

8) Subclauses 7.1.2, 8.1.2.12, 8.6.2 and A.3.3

In subclauses 7.1.2, 8.1.2.12, 8.6.2 and A.3.3, change all occurrences of GDMO template(s) to template(s).

9) Subclause 7.3

Add an X in the four Table 2 cells that are in either the Definition managed objects or Definition Directory objects columns and that are in either the Managed object class knowledge or Relationship knowledge rows. Change both occurrences of X (Note 7) to X and eliminate Note 7.

10) New subclauses 8.1.2.12 and 8.1.2.13

Renumber subclause 8.1.2.12 as 8.1.2.14 and insert two new subclauses before it:

8.1.2.12 Relationship class template managed object

The relationship class template managed object represents a GRM relationship class specification.

The relationship class template managed object class is defined as a subclass of the template managed object class. Its mandatory, read-only attributes are:

- a) derived from;
- b) behaviour.

Its optional, read-only attributes are:

- a) supports;
- b) qualified by;
- c) role specifier.

8.1.2.13 Relationship mapping template managed object

The relationship mapping template managed object represents a GRM relationship mapping specification.

The relationship mapping template managed object class is defined as a subclass of the template managed object class. Its mandatory, read-only attributes are:

- a) mapped relationship class;
- b) behaviour;
- c) role mapping specification;
- d) registered as.

Its optional, read-only attributes are:

- a) relationship object;
- b) operations mapping.

11) Subclause 8.6.2

In this subclause, change the two occurrences of GDMO specifications to GDMO, GRM or ASN.1 specifications.

Append to this subclause the following paragraphs:

The Directory auxiliary class relationship class template is defined for inclusion in Directory entries of Directory structural class Registered information and represents a GRM relationship class specification. It must contain the attributes Derived from and Behaviour. It may contain the attributes Supports, Qualified by, and Role specifier.

The Derived from attribute identifies the immediate superclasses of a managed relationship class. The Behaviour attribute specifies the behaviour associated with the managed relationship class. The Supports attribute defines the relationship management operations and notifications that a managed relationship supports. The Qualified by attribute specifies the attributes that are associated with the managed relationship as a whole. The Role specifier attribute identifies the roles associated with the managed relationship class.

The Directory auxiliary class Relationship mapping template is defined for inclusion in Directory entries of Directory structural class Registered information and represents a GRM relationship mapping specification. It must contain the attributes Behaviour, Mapped relationship class, Role mapping specification set, and Registered as. It may contain the attributes Relationship object and Operations mapping.

The Behaviour attribute defines the behaviour associated with the relationship mapping. The Mapped relationship class attribute specifies the managed relationship class to which the represented relationship mapping is related. The Role mapping specification set attribute identifies candidate managed object classes that may fulfil a given role. The Registered as attribute contains the object identifier of the represented relation mapping. The Relationship object attribute indicates the class of a relationship object that shall represent the managed relationship. The Operations mapping attribute specifies the mapping of a relationship management operation to one or more systems management operations.

12) Subclause 11.1.1.2

Add entries to Table 6 for relationship class template and relationship mapping template (inserting before the entry for the template class).

13) Subclause 11.1.5.1

Add entries to Table 9 for the new attributes mentioned in the new subclauses 8.1.2.12 and 8.1.2.13 .

14) Subclause 11.1.5.2

Insert in Table 14 the following entries before the entry for the template class:

Relationship class template	managementRelationshipClass
Relationship mapping template	managementRelationshipMapping

15) Subclause 11.1.6.1

Insert in Table 15 the following entries before the entry for the template class:

relationshipClassTemplate	managementRelationshipClass
relationshipMappingTemplate	managementRelationshipMapping

16) Subclause 11.2.1.2

Add to Table 17 the following entries (inserting in alphabetical order):

Mapped relationship class	mappedRelationshipClass
Operations mapping	operationsMapping
Qualified by	qualifiedBy
Relationship object	relationshipObject
Role mapping specification set	roleMappingSpecificationSet
Role specifier	roleSpecifier
Supports	supports

17) Subclause A.1

In this subclause change:

--%PRAGMA version BIT STRING {v1990(0), v1994(1)} ::= {v1990, v1994}

to:

--<ASN1.Version 1990, 1994 MKMD >--

18) Subclause A.2.4

In this subclause change:

--%PRAGMA version BIT STRING {v1990(0), v1994(1)} ::= v1990

to:

--<ASN1.Version 1990 RepertoireASNIModule >--

Also in this subclause change:

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

to:

FROM Attribute-ASN1Module {joint-iso-itu-t(2) ms(9) smi(3) part2(2) asn1Module(2) 1}

19) Subclause A.3.1

Add to this subclause these managed object class definitions (inserting in alphabetical order):

relationshipClassTemplate MANAGED OBJECT CLASS

DERIVED FROM template;

CHARACTERIZED BY relationshipClassPackage PACKAGE

BEHAVIOUR relationshipClassBehaviour **BEHAVIOUR DEFINED AS !**

A managed object with this behaviour represents a **RELATIONSHIP CLASS** template.

!;;

ATTRIBUTES

derivedFrom GET,

behaviour GET;;;

CONDITIONAL PACKAGES

relationshipClassSupportsPackage PACKAGE

BEHAVIOUR relationshipClassSupportsBehaviour **BEHAVIOUR DEFINED AS !**

A managed object with this behaviour represents a template that was defined with the **SUPPORTS** construct.

!;;

ATTRIBUTES

supports GET;

REGISTERED AS {MKMD.mkmPackage 5}; **PRESENT IF** "managed object represents a template that was defined with the **SUPPORTS** construct",

relationshipClassQualifiedByPackage PACKAGE

BEHAVIOUR relationshipClassQualifiedByBehaviour **BEHAVIOUR DEFINED AS !**

A managed object with this behaviour represents a template that was defined with the **QUALIFIED BY** construct.

!;;

ATTRIBUTES

qualifiedBy GET;

REGISTERED AS {MKMD.mkmPackage 6}; **PRESENT IF** "managed object represents a template that was defined with the **QUALIFIED BY** construct",

relationshipClassRoleSpecifierPackage PACKAGE

BEHAVIOUR relationshipClassRoleSpecifierBehaviour **BEHAVIOUR DEFINED AS !**

A managed object with this behaviour represents a template that was defined with the role-specifier supporting production.

!;;

ATTRIBUTES

roleSpecifier GET;

REGISTERED AS {MKMD.mkmPackage 7}; **PRESENT IF** "managed object represents a template that was defined with the role-specifier supporting production";

REGISTERED AS {MKMD.mkmMObjectClass 16};

relationshipMappingTemplate MANAGED OBJECT CLASS

DERIVED FROM template;

CHARACTERIZED BY relationshipMappingPackage PACKAGE

BEHAVIOUR relationshipMappingBehaviour **BEHAVIOUR DEFINED AS !**

A managed object with this behaviour represents a **RELATIONSHIP MAPPING** template.

!;;

ATTRIBUTES

mappedRelationshipClass GET,

behaviour GET,

roleMappingSpecificationSet GET,

registeredAs GET;;;

CONDITIONAL PACKAGES

relationshipMappingRelationshipObjectPackage PACKAGE

BEHAVIOUR relationshipMappingRelationshipObjectBehaviour **BEHAVIOUR DEFINED AS !**

A managed object with this behaviour represents a template that was defined with the **RELATIONSHIP OBJECT** construct.

!;;

ATTRIBUTES

relationshipObject GET;

REGISTERED AS {MKMD.mkmPackage 8}; **PRESENT IF** "managed object represents a template that was defined with the **RELATIONSHIP OBJECT** construct",

relationshipMappingOperationsMappingPackage PACKAGE

BEHAVIOUR relationshipMappingOperationsMappingBehaviour BEHAVIOUR DEFINED AS !

A managed object with this behaviour represents a template that was defined with the OPERATIONS MAPPING construct.

!;;

ATTRIBUTES

operationsMapping GET;

REGISTERED AS {MKMD.mkmPackage 9}; PRESENT IF "managed object represents a template that was defined with the OPERATIONS MAPPING construct";

REGISTERED AS {MKMD.mkmObjectClass 17};

20) Subclause A.3.3

In this subclause change:

BEHAVIOUR derivedFromBehaviour BEHAVIOUR DEFINED AS !

This set-valued attribute identifies the immediate superclasses of a managed object class.

!;;

to:

BEHAVIOUR derivedFromBehaviour BEHAVIOUR DEFINED AS !

This set-valued attribute identifies the immediate superclasses of a managed object class or managed relationship class.

!;;

Add to subclause A.3.3 these attribute definitions (inserting in alphabetical order):

mappedRelationshipClass ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.MappedRelationshipClass;

MATCHES FOR EQUALITY;

REGISTERED AS {MKMD.mkmAttribute 46};

operationsMapping ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.OperationsMapping;

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

REGISTERED AS {MKMD.mkmAttribute 47};

qualifiedBy ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.QualifiedBy;

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

REGISTERED AS {MKMD.mkmAttribute 48};

relationshipObject ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.RelationshipObject;

MATCHES FOR EQUALITY;

REGISTERED AS {MKMD.mkmAttribute 49};

roleMappingSpecificationSet ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.RoleMappingSpecificationSet;

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

REGISTERED AS {MKMD.mkmAttribute 50};

roleSpecifier ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.RoleSpecifier;

MATCHES FOR EQUALITY;

REGISTERED AS {MKMD.mkmAttribute 51};

supports ATTRIBUTE

WITH ATTRIBUTE SYNTAX GrmAttributeDefinitionModule.Supports;

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

REGISTERED AS {MKMD.mkmAttribute 52};

21) Subclause A.3.5

In this subclause change:

```
--%PRAGMA version BIT STRING {v1990(0), v1994(1)} ::= {v1990, v1994}
```

to:

```
--<ASN1.Version 1990, 1994 DefinitionASN1Module >--
```

Also in subclause A.3.5 change:

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

to:

```
FROM Attribute-ASN1Module {joint-iso-itu-t(2) ms(9) smi(3) part2(2) asn1Module(2) 1}
```

Add to the same subclause this new ASN.1 module (after END of DefinitionASN1Module):

```
-- <ASN1.Version 1990, 1994 GrmAttributeDefinitionModule >--
```

```
GrmAttributeDefinitionModule {joint-iso-itu-t(2) ms(9) function(2) part16(16) asn1Modules(2) 6}
```

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

```
BEGIN
```

```
-- EXPORTS everything;
```

```
IMPORTS
```

```
DefinedType, Identifier, OptionallyRegisteredAs, TemplateLabel, TemplateList
```

```
FROM DefinitionASN1Module {joint-iso-itu-t(2) ms(9) function(2) part16(16) asn1Modules(2) 1}
```

```
-- this Recommendation / International Standard
```

```
;
```

```
LabelAndParameters ::= SEQUENCE {
    label                TemplateLabel,
    parameters           TemplateList OPTIONAL}
MappedRelationshipClass ::= TemplateLabel
MappingPair ::= SEQUENCE {
    relationshipOperation RelationshipOperation,
    mapsTo               MapsTo}
MapsTo ::= SET OF MapsToPair
MapsToPair ::= SEQUENCE {
    systemsMgtOperation SystemsMgtOperation,
    roleOrRelationshipObject RoleOrRelationshipObject}
OperationsMapping ::= SET OF MappingPair
OptionalIdentifier ::= CHOICE {
    present              Identifier,
    absent              NULL}
OptionalIdentifierAndRole ::= SEQUENCE {
    optionalOperationName OptionalIdentifier,
    optionalRoleName     Identifier OPTIONAL}
OptionalLabelAndParameters ::= SEQUENCE {
    label                TemplateLabel OPTIONAL,
    parameters           TemplateList OPTIONAL}
QualifiedBy ::= TemplateList
RelationshipObject ::= SEQUENCE {
    class                TemplateLabel,
    qualifies            TemplateList OPTIONAL}
RelationshipOperation ::= CHOICE {
    establish            [0] EXPLICIT OptionalIdentifier,
    terminate           [1] EXPLICIT OptionalIdentifier,
    query               [2] OptionalIdentifierAndRole,
    notify              [3] EXPLICIT OptionalIdentifier,
    userDefined         [4] EXPLICIT OptionalIdentifier,
    bind                [5] OptionalIdentifierAndRole,
    unbind              [6] OptionalIdentifierAndRole}
```

Representation	::= CHOICE { nameBindingSuperior nameBindingSubordinate attribute relationshipObjectPointer operation RoleMappingSpecification ::= SEQUENCE { role relatedClasses representedBy qualifies RoleMappingSpecificationSet ::= SET OF RoleMappingSpecification RoleOrRelationshipObject ::= CHOICE { role relationshipObject RoleSpecifier ::= SEQUENCE { role compatibleWith permittedRoleCardinality requiredRoleCardinality bindingSupport unbindingSupport permittedRelationshipCardinality optionalRegisteredAs SupportedChoice ::= CHOICE { establish terminate query notify userDefined Supports ::= SET OF SupportedChoice SystemsMgtOperation ::= CHOICE { get replace add remove create delete action notification END	[0] TemplateLabel, [1] TemplateLabel, [2] TemplateLabel, [3] TemplateLabel, NULL} Identifier, TemplateList, [0] EXPLICIT Representation OPTIONAL, [1] TemplateList OPTIONAL} Identifier, NULL} Identifier, [0] TemplateLabel OPTIONAL, [1] DefinedType OPTIONAL, [2] DefinedType OPTIONAL, [3] EXPLICIT OptionalIdentifier OPTIONAL, [4] EXPLICIT OptionalIdentifier OPTIONAL, [5] DefinedType OPTIONAL, OptionallyRegisteredAs} [0] EXPLICIT OptionalIdentifier, [1] EXPLICIT OptionalIdentifier, [2] EXPLICIT OptionalIdentifier, [3] EXPLICIT OptionalIdentifier, [4] EXPLICIT OptionalIdentifier} [0] LabelAndParameters, [1] LabelAndParameters, [2] LabelAndParameters, [3] LabelAndParameters, [4] OptionalLabelAndParameters, [5] TemplateList, [6] LabelAndParameters, [7] LabelAndParameters}
-----------------------	--	--

22) Subclause A.4.1

In this subclause, in the DERIVED FROM section, remove the extraneous blank between the double-quote and CCITT.

Also, in this subclause, in the ATTRIBUTES section, change:

discoveryId;

to:

discoveryId GET;

23) Subclause A.4.5

In this subclause change:

--%PRAGMA version BIT STRING {v1990(0), v1994(1)} ::= {v1990, v1994}

to:

--<ASN1.Version 1990, 1994 DiscoveryASNIModule >--

Also in this subclause change:

```
FROM InformationFramework {joint-iso-ccitt(2) ds(5) modules(1) informationFramework(1)}
-- ITU-T Rec.X.501 | ISO/IEC 9594-2
```

to:

```
FROM InformationFramework {joint-iso-itu-t(2) ds(5) modules(1) informationFramework(1) 2}
-- NOTE – This Recommendation | International Standard imports
-- RelativeDistinguishedName from CCITT Rec. X.501 (1988) | ISO/IEC 9594-2:1990.
-- The specification for this syntax can now be found in an informative
-- annex of ITU-T Rec. X.711 (1997) | ISO/IEC 9596-1:1998.
```

In the same subclause, in the production for **MITSearch**, ensure that blank space is apparent between **discoveryScope** and **Scope**.

24) Subclause B.1

In this subclause change:

```
--%PRAGMA version BIT STRING {v1990(0), v1994(1)} ::= v1994
```

to:

```
--< ASN1.Version 1994 RepertoireDirectoryASN1Module >--
```

25) Subclause B.2

In this subclause change:

```
--%PRAGMA version BIT STRING {v1990(0), v1994(1)} ::= v1994
```

to:

```
--< ASN1.Version 1994 DefinitionDirectoryASN1Module >--
```

Add to this subclause this new ASN.1 module (after **END** of **DefinitionDirectoryASN1Module**):

```
--<ASN1.Version 1994 GrmDefinitionDirectoryASN1Module >--
```

```
GrmDefinitionDirectoryASN1Module {joint-iso-itu-t(2) ms(9) function(2) part16(16) asn1Modules(2) 7}
```

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

```
BEGIN
```

```
-- EXPORTS everything;
```

```
IMPORTS
```

```
ATTRIBUTE, OBJECT-CLASS, CONTENT-RULE
```

```
FROM InformationFramework {joint-iso-itu-t(2) ds(5) modules(1) informationFramework(1) 2}
-- ITU-T Rec. X.501 | ISO/IEC 9594-2
```

```
behaviour, derivedFrom, managementTemplate, registeredAs, registeredInformation
```

```
FROM DefinitionDirectoryASN1Module
{joint-iso-itu-t(2) ms(9) function(2) part16(16) asn1Modules(2) 4}
-- this Recommendation | International Standard
```

```
MappedRelationshipClass, MappingPair, RelationshipObject, RoleMappingSpecification, RoleSpecifier, SupportedChoice
```

```
FROM GrmAttributeDefinitionModule
{joint-iso-itu-t(2) ms(9) function(2) part16(16) asn1Modules(2) 6}
-- this Recommendation | International Standard
```

```
mkmDirectoryObjectClass, mkmDirectoryAttributeType
```

```
FROM MKMD {joint-iso-itu-t(2) ms(9) function(2) part16(16) asn1Modules(2) 5}
-- this Recommendation | International Standard
```

```
;
```

```

managementRelationshipClass OBJECT-CLASS ::= {
    SUBCLASS OF    managementTemplate
    KIND           auxiliary
    MUST CONTAIN   { derivedFrom |
                    behaviour }
    MAY CONTAIN    { supports |
                    qualifiedBy |
                    roleSpecifier }
    ID             { mkmDirectoryObjectClass 16 }}

managementRelationshipMapping OBJECT-CLASS ::= {
    SUBCLASS OF    managementTemplate
    KIND           auxiliary
    MUST CONTAIN   { mappedRelationshipClass |
                    behaviour |
                    roleMappingSpecificationSet |
                    registeredAs }
    MAY CONTAIN    { relationshipObject |
                    operationsMapping }
    ID             { mkmDirectoryObjectClass 17 }}

-- Definition of DIT content rules
-- Only one of the specified auxiliary object classes can be
-- included in a given entry of class registeredInformation
registeredManagementRelationshipInformationCR CONTENT-RULE ::= {
    STRUCTURAL OBJECT CLASS    registeredInformation
    AUXILIARY OBJECT CLASS     { managementRelationshipClass |
                                managementRelationshipMapping }}

-- Definition of used attributes
mappedRelationshipClass ATTRIBUTE ::= {
    WITH SYNTAX    MappedRelationshipClass
    SINGLE VALUE   TRUE
    ID             { mkmDirectoryAttributeType 47 }}

operationsMapping ATTRIBUTE ::= {
    WITH SYNTAX    MappingPair
    ID             { mkmDirectoryAttributeType 48 }}

qualifiedBy ATTRIBUTE ::= {
    WITH SYNTAX    TemplateLabel
    ID             { mkmDirectoryAttributeType 49 }}

relationshipObject ATTRIBUTE ::= {
    WITH SYNTAX    RelationshipObject
    SINGLE VALUE   TRUE
    ID             { mkmDirectoryAttributeType 50 }}

roleMappingSpecificationSet ATTRIBUTE ::= {
    WITH SYNTAX    RoleMappingSpecification
    ID             { mkmDirectoryAttributeType 51 }}

roleSpecifier ATTRIBUTE ::= {
    WITH SYNTAX    RoleSpecifier
    SINGLE VALUE   TRUE
    ID             { mkmDirectoryAttributeType 52 }}

supports ATTRIBUTE ::= {
    WITH SYNTAX    SupportedChoice
    ID             { mkmDirectoryAttributeType 53 }}
END

```

26) Annexes A to F

Add this footnote to the titles of Annexes A to F (and renumber the existing two footnotes):

- 1) Users of this Recommendation | International Standard may freely reproduce the contents of this annex so that it can be used for its intended purpose.

27) Subclause C.1.3

In this subclause, add the following text after the line:

- Not applicable or out of scope

The value of 'm' in Status column for the receiving of parameters, of tables of type MICS or PICS, indicates that there is a minimum requirement for the implementation to be able to receive the parameter. The Additional information column shall be used to state whether the implementation provides support for more than the minimum requirement.

28) Subclause C.4

In Table C.3, add these rows in index order (after existing rows):

17	Relationship class template managed object class	c6		
18	Relationship mapping template managed object class	c6		

In Table C.5, change the status from m to o and add the following Note to the table:

NOTE – The supplier of implementation shall indicate the application contexts supported.
--

In Table C.6, add these rows in index order (after existing rows):

17	ITU-T Rec. X.750 ISO/IEC 10164-16	Tables E.85-E.90	relationship class template managed object class	–	c26			
18	ITU-T Rec. X.750 ISO/IEC 10164-16	Tables E.91-E.96	relationship mapping template managed object class	–	c27			

Following Table C.6, add these conditions after existing ones:

c26: if C.3/17a then m else –

c27: if C.3/18a then m else –

29) Subclause D.3.1

In Table D.1, add these rows in index order (after existing rows for first set of columns and before second set of columns):

50	mappedRelationshipClass	{MKMD.mkmAttribute 46}	–	–		o.4	
51	operationsMapping	{MKMD.mkmAttribute 47}	–	–		o.4	
52	qualifiedBy	{MKMD.mkmAttribute 48}	–	–		o.4	
53	relationshipObject	{MKMD.mkmAttribute 49}	–	–		o.4	
54	roleMappingSpecificationSet	{MKMD.mkmAttribute 50}	–	–		o.4	
55	roleSpecifier	{MKMD.mkmAttribute 51}	–	–		o.4	
56	supports	{MKMD.mkmAttribute 52}	–	–	–	o.4	

In Table D.1, add these rows in index order (after existing rows for second set of columns):

50	–		–		–		–		
51	–		–		–		–		
52	–		–		–		–		
53	–		–		–		–		
54	–		–		–		–		
55	–		–		–		–		
56	–		–		–		–		

30) New subclauses E.3.16 and E.3.17

At the end of Annex E (after Table E.84), add the new subclauses E.3.16 and E.3.17 and the following tables:

E.3.16 Relationship class template managed object class

Table E.85 – Relationship class template 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	relationshipClassTemplate	{MKMD.mkmMObjectClass 16}		

If the answer to the actual class question in Table E.85 is No, the supplier of the implementation shall fill in the actual class support Table E.86.

Table E.86 – Actual class support

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

Table E.87 – Package support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information
1	templateDefinitionPackage	{MKMD.mkmPackage 3}	–	o		
2	relationshipClassSupports Package	{MKMD.mkmPackage 5}	–			
3	relationshipClassQualifiedByPackage	{MKMD.mkmPackage 6}	–			
4	relationshipClassRoleSpecifierPackage	{MKMD.mkmPackage 7}	–	o		

Table E.88 – Relationship class template managed object attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get	
				Status	Support	Status	Support
1	objectClass	{MKMD.dmiAttribute 65}	–	x		m	
2	nameBinding	{MKMD.dmiAttribute 63}	–	x		m	
3	packages	{MKMD.dmiAttribute 66}	–	x		m	
4	allomorphs	{MKMD.dmiAttribute 50}	–	x		c76	
5	templateName	{MKMD.mkmAttribute 7}	–	x		m	
6	templateDefinition	{MKMD.mkmAttribute 37}	–	x		c77	
7	derivedFrom	{MKMD.mkmAttribute 10}	–	x		m	
8	behaviour	{MKMD.mkmAttribute 15}	–	x		m	
9	supports	{MKMD.mkmAttribute 52}	–	x		c78	
10	qualifiedBy	{MKMD.mkmAttribute 48}	–	x		c79	
11	roleSpecifier	{MKMD.mkmAttribute 51}	–	x		c80	

c76: if (not E.85/1b) then m else –
c77: if E.87/1a then m else –
c78: if E.87/2a then m else –
c79: if E.87/3a then m else –
c80: if E.87/4a then m else –

Table E.88 (concluded)

Index	Replace		Add		Remove		Set to default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	x		–		–		–		
2	x		–		–		–		
3	x		x		x		–		
4	x		x		x		–		
5	x		–		–		–		
6	x		–		–		–		
7	x		–		–		–		
8	x		–		–		–		
9	x		–		–		–		
10	x		–		–		–		
11	x		–		–		–		

Table E.89 – Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information
1	getTextualRepresentation	{MKMD.mkmAction 2}	–	c77		

Table E.89 (concluded)

Index	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	1.1	AttributeId (INFORMATION SYNTAX)	–	c:m		
	1.2	TextualRepresentation (REPLY SYNTAX)	–	c:m		

Table E.90 – Parameter support

Index	Parameter template label	Value of object identifier for parameter	Constraints and values	Status	Support	Additional information
1	attributeNotAvailableViaThisAction	{MKMD.mkmParameter 1}		c77		

E.3.17 Relationship mapping template managed object class

Table E.91 – Relationship mapping template 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	relationshipMappingTemplate	{MKMD.mkmMObjectClass 17}		

If the answer to the actual class question in Table E.91 is No, the supplier of the implementation shall fill in the actual class support Table E.92.

Table E.92 – Actual class support

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

Table E.93 – Package support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information
1	templateDefinitionPackage	{MKMD.mkmPackage 3}	–	o		
2	relationshipMappingRelationshipObjectPackage	{MKMD.mkmPackage 8}	–			
3	relationshipMappingOperationsMappingPackage	{MKMD.mkmPackage 9}	–			

Table E.94 – Relationship mapping template managed object attribute support

Index	Attribute template label	Value of object identifier for attribute	Constraints and values	Set by create		Get	
				Status	Support	Status	Support
1	objectClass	{MKMD.dmiAttribute 65}	–	x		m	
2	nameBinding	{MKMD.dmiAttribute 63}	–	x		m	
3	packages	{MKMD.dmiAttribute 66}	–	x		m	
4	allomorphs	{MKMD.dmiAttribute 50}	–	x		c81	
5	templateName	{MKMD.mkmAttribute 7}	–	x		m	
6	templateDefinition	{MKMD.mkmAttribute 37}	–	x		c82	
7	mappedRelationshipClass	{MKMD.mkmAttribute 46}	–	x		m	
8	behaviour	{MKMD.mkmAttribute 15}	–	x		m	
9	roleMappingSpecificationSet	{MKMD.mkmAttribute 50}	–	x		m	
10	registeredAs	{MKMD.mkmAttribute 14}	–	x		m	
11	relationshipObject	{MKMD.mkmAttribute 49}	–	x		c83	
12	operationsMapping	{MKMD.mkmAttribute 47}	–	x		c84	
c81: if (not E.91/1b) then m else – c82: if E.93/1a then m else – c83: if E.93/2a then m else – c84: if E.93/3a then m else –							

Table E.94 (concluded)

Index	Replace		Add		Remove		Set to default		Additional information
	Status	Support	Status	Support	Status	Support	Status	Support	
1	x		–		–		–		
2	x		–		–		–		
3	x		x		x		–		
4	x		x		x		–		
5	x		–		–		–		
6	x		–		–		–		
7	x		–		–		–		
8	x		–		–		–		
9	x		–		–		–		
10	x		–		–		–		
11	x		–		–		–		
12	x		–		–		–		

Table E.95 – Action support

Index	Action type template label	Value of object identifier for action type	Constraints and values	Status	Support	Additional information
1	getTextualRepresentation	{MKMD.mkmAction 2}	–	c82		

Table E.95 (concluded)

Index	Subindex	Action field name label	Constraints and values	Status	Support	Additional information
1	1.1	AttributeId (INFORMATION SYNTAX)	–	c:m		
	1.2	TextualRepresentation (REPLY SYNTAX)	–	c:m		

Table E.96 – Parameter support

Index	Parameter template label	Value of object identifier for parameter	Constraints and values	Status	Support	Additional information
1	attributeNotAvailableViaThisAction	{MKMD.mkmParameter 1}		c82		

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure
Series Z	Programming languages