



INTERNATIONAL TELECOMMUNICATION UNION

# ITU-T

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

# M.3100

**Corrigendum 1**  
(07/98)

SERIES M: TMN AND NETWORK MAINTENANCE:  
INTERNATIONAL TRANSMISSION SYSTEMS,  
TELEPHONE CIRCUITS, TELEGRAPHY, FACSIMILE  
AND LEASED CIRCUITS

Telecommunications management network

---

**Generic network information model**

**Corrigendum 1**

ITU-T Recommendation M.3100 – Corrigendum 1

(Previously CCITT Recommendation)

---

## ITU-T M-SERIES RECOMMENDATIONS

### TMN AND NETWORK MAINTENANCE: INTERNATIONAL TRANSMISSION SYSTEMS, TELEPHONE CIRCUITS, TELEGRAPHY, FACSIMILE AND LEASED CIRCUITS

Introduction and general principles of maintenance and maintenance organization	M.10–M.299
International transmission systems	M.300–M.559
International telephone circuits	M.560–M.759
Common channel signalling systems	M.760–M.799
International telegraph systems and phototelegraph transmission	M.800–M.899
International leased group and supergroup links	M.900–M.999
International leased circuits	M.1000–M.1099
Mobile telecommunication systems and services	M.1100–M.1199
International public telephone network	M.1200–M.1299
International data transmission systems	M.1300–M.1399
Designations and information exchange	M.1400–M.1999
International transport network	M.2000–M.2999
<b>Telecommunications management network</b>	<b>M.3000–M.3599</b>
Integrated services digital networks	M.3600–M.3999
Common channel signalling systems	M.4000–M.4999

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

# **ITU-T RECOMMENDATION M.3100**

## **GENERIC NETWORK INFORMATION MODEL**

### **CORRIGENDUM 1**

#### **Summary**

This technical corrigendum corrects defects identified in Recommendation M.3100 (1995). It includes a table providing the relation between the defects and the corrections. These corrections are specified as changes to existing clauses of Recommendation M.3100 (1995).

#### **Source**

Corrigendum 1 to ITU-T Recommendation M.3100 was prepared by ITU-T Study Group 4 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on the 26th of June 1998.

## 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**

		<b>Page</b>
1	Introduction .....	1
2	Resolved defects.....	1



## Recommendation M.3100

### GENERIC NETWORK INFORMATION MODEL

#### CORRIGENDUM 1

(Geneva, 1998)

## 1 Introduction

This technical corrigendum corrects a number of defects to Recommendation M.3100 that have previously been documented and resolved in the M.3100 Implementors' Guide. This technical corrigendum replaces the Implementors' Guide as the authoritative source. However, the Implementors' Guide will be available on the ITU-T server until this corrigendum has been published.

Additional defects and resolutions will again be recorded in the Implementors' Guide and finally be published in an additional technical corrigendum or a revision of Recommendation M.3100.

## 2 Resolved defects

This technical corrigendum corrects the following defects reported against Recommendation M.3100 (1995):

Defect number	Issue	Correction No.
DR-M3100-2	Information flow in Figure I.6 should be from left to right and the sink and source labels should be corrected.	42
DR-M3100-3	Modify the subordinateCircuitPackSoftwareLoad attribute behaviour to add clarification.	17
DR-M3100-4	Add DELETE to (new version of) autocreated circuitPack name binding.	19, 43
DR-M3100-5	Additional user guidelines on use of supportedByObjectList.	40
DR-M3100-6	Multiple use of <i>package</i> label in OID assignments.	32
DR-M3100-7	Addition of Error Parameter to (some) name bindings.	19, 20, 21, 23, 24, 29
DR-M3100-8	Missing resource type attribute in equipmentR2 object class.	2, 18, 39
DR-M3100-9	Missing ProblemCause value.	38
DR-M3100-10	Missing probable cause values (derived from work on GSM).	35
DR-M3100-11	Missing probable cause values for real time clock.	35
DR-M3100-12	Duplicated behaviour definition in name binding.	25
DR-M3100-14	Typographical and editorial errors.	1, 3, 4, 12, 15, 16, 41
DR-M3100-15	Revised crossConnection.	8, 10, 13, 28, 36
DR-M3100-17	Missing namebinding from managedElementComplex.	26
DR-M3100-18	Possible values of equipmentHolderType.	14, 44
DR-M3100-19	Need of possibleCircuitPackList attribute.	44

Defect number	Issue	Correction No.
DR-M3100-21	Missing name bindings from managedElement (and subclasses) to managed objects named from system in X.700-series.	27
DR-M3100-22	Assignment of probable cause values related to radio relay management.	35
DR-M3100-23	Enhancement of the generalErrorParameter with addition of new cause codes.	29, 31, 37
DR-M3100-24	Several editors' errors are identified.	5, 6, 7, 30, 34, 36
DR-M3100-25	Definition of object identifier of e0-e4 levels.	33
DR-M3100-26	Clarification of delete behaviour of circuitPack.	19
DR-M3100-27	Modification to behaviour of equipmentHolder-equipmentHolder name binding to allow an equipmentHolder to contain both equipmentHolder and circuitPack objects.	22
DR-M3100-28	Addition of missing notifications to fabricR1 and gtp managed object class.	9, 11
DR-M3100-29	Add missing name binding of managedElementComplex to network	26a

### 1) Modify subclause 3.2.1 "Circuit Pack"

*Correct the ASN.1 module reference in the permitted value specification of the availabilityStatus attribute to read:*

**"Recommendation X.721: 1992": availabilityStatus PERMITTED VALUES**  
**ASN1DefinedTypesModule.CircuitPackAvailabilityStatus GET;;;**

### 2) New subclause 3.2.x "Equipment R2"

The differentiation of equipment objects currently must be done via userLabel, locationName or object-specific values of equipmentId. None of these solutions is adequate.

*Add the following new subclass equipmentR2 managed object class definition in a new subclause 3.2.x between existing subclauses 3.2.2 and 3.2.3:*

**equipmentR2 MANAGED OBJECT CLASS**  
**DERIVED FROM equipmentR1;**  
**CHARACTERIZED BY**  
    **equipmentR2Package PACKAGE**  
    **BEHAVIOUR**  
        **equipmentR2PackageBehaviour BEHAVIOUR DEFINED AS**  
        **"The typeText attribute indicates the type of the equipment";;**  
    **ATTRIBUTES**  
        **typeText GET SET-BY-CREATE;;;**  
**REGISTERED AS {m3100ObjectClass 35};**

### 3) Subclause 3.3.2 "Connection Termination Point Sink"

*Correct the ASN.1 module reference in the permitted value specification of the downstreamConnectivityPointer attribute to read:*

**downstreamConnectivityPointer PERMITTED VALUES**  
*-- The allowed choices for the syntax of this attribute are restricted in the subtype*  
*-- CTPDownstreamPointer*  
**ASN1DefinedTypesModule.CTPDownstreamPointer GET SET-BY-CREATE;;;**



#### 4) Subclause 3.3.3 "Connection Termination Point Source"

*Correct the ASN.1 module reference in the permitted value specification of the upstreamConnectivityPointer attribute to read:*

**upstreamConnectivityPointer PERMITTED VALUES**

-- The allowed choices for the syntax of this attribute are restricted in the subtype  
-- CTPUpstreamPointer

ASN1DefinedTypesModule.CTPUpstreamPointer GET SET-BY-CREATE;;;

#### 5) Subclause 3.3.6 "Trail Termination Point Sink"

*Replace the last part of the behaviour statement:*

"The upstream connectivity pointer attribute points to the termination point managed object, within the same managed element, that sends information (traffic) to this termination point at the same layer, or is null. The referenced object shall be an instance of one of the following classes or its subclasses: Connection Termination Point Sink or Bidirectional (single or a concatenated sequence) or Trail Termination Point Source or Bidirectional."

*with the following:*

"The upstream connectivity pointer attribute points to the termination point managed object, within the same managed element, that sends information (traffic) to this termination point at the same layer, or is null. The referenced object shall be an instance of one of the following classes or its subclasses: Connection Termination Point Sink or Bidirectional (single or a concatenated sequence) or Trail Termination Point Source or Bidirectional (single or a concatenated sequence)."

#### 6) Subclause 3.4.2 "ConnectionR1"

The existing definition defines connectionPackage and connectionBehaviour which is already defined by the Connection object class.

*Modify the label of the package and the behaviour statement as follows:*

**connectionR1 MANAGED OBJECT CLASS**

**DERIVED FROM** pipe;

**CHARACTERIZED BY**

connectionR1Package PACKAGE

**BEHAVIOUR**

connectionR1Behaviour BEHAVIOUR DEFINED AS

"The Connection object class is a class of managed objects responsible for the transparent transfer of information between connection termination points. A connection is a component of a trail.

Several connections can be bundled into a higher rate trail.

A sequence of one or more connections are linked together to form a trail. A connection may be either uni- or bidirectional.";;

**ATTRIBUTES**

connectionId GET SET-BY-CREATE;;;

**CONDITIONAL PACKAGES**

serverTrailListPackage PRESENT IF "an instance supports it",

clientTrailPackage PRESENT IF "an instance supports it";

**REGISTERED AS** {m3100ObjectClass 23};

#### 7) Subclause 3.4.4 "TrailR1"

The existing definition defines trailBehaviour which is already defined by the Trail object class.

*Modify the behaviour label as follows:*

**trailR1 MANAGED OBJECT CLASS**

**DERIVED FROM** pipe;

**CHARACTERIZED BY**

trailR1Package PACKAGE

**BEHAVIOUR**

**trailR1Behaviour BEHAVIOUR  
DEFINED AS**

"Trail is a class of managed objects in layer networks which is responsible for the integrity of transfer of characteristic information from one or more other layer networks. A trail is composed of two Trail Termination Points and one or more connection and associated connection termination points.";;

**ATTRIBUTES**

trailId GET SET-BY-CREATE;;;

**CONDITIONAL PACKAGES**

serverConnectionListPackage PRESENT IF "an instance supports it",

clientConnectionListPackage PRESENT IF "an instance supports it";

**REGISTERED AS {m3100ObjectClass 25};**

**8) New subclause 3.5.x "Cross-connectionR1"**

An more efficient way to indicate a creation and deletion of cross-connections is required. In addition, the "redline" attribute is not required in some regions.

*Add the following new crossConnectionR1 managed object class definition in a new subclause 3.5.x following existing subclause 3.5.1:*

**crossConnectionR1 MANAGED OBJECT CLASS**

**DERIVED FROM crossConnection;**

**CHARACTERIZED BY**

crossConnectionR1Package PACKAGE

**BEHAVIOUR**

crossConnectionR1Behaviour BEHAVIOUR

**DEFINED AS**

"When the attribute value change notification package is present, the attributeValueChange notification shall be emitted when userLabel or redline changes value (when the attributes are present).

When the state change notification package is present, the stateChange notification shall be emitted when operationalState or administrativeState changes value.";;;

**CONDITIONAL PACKAGES**

userLabelPackage PRESENT IF "an instance supports it",

redlinePackage PRESENT IF "an instance supports it",

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.";

**REGISTERED AS {m3100ObjectClass 37};**

-- This managed object class is the preferred replacement of both crossConnection and

-- namedCrossConnection.

**9) New subclause 3.5.x "FabricR2"**

The fabric (and fabricR1) managed object classes defines administrativeState, operationalState and availabilityStatus attributes. However, the stateChangeNotificationPackage is not defined in object class to allow for unsolicited emission of state/status change notifications.

*Add the following new fabricR2 managed object class definition in a new subclause 3.5.x "FabricR2" between existing subclauses 3.5.3 and 3.5.4:*

**fabricR2 MANAGED OBJECT CLASS**

**DERIVED FROM fabricR1;**

**CHARACTERIZED BY**

fabricR2Package PACKAGE

**BEHAVIOUR**

**fabricR2Behaviour BEHAVIOUR**

**DEFINED AS**

"When the state change notification package is present, the stateChange notification shall be emitted when operationalState or administrativeState changes value.";;;

**CONDITIONAL PACKAGES**

stateChangeNotificationPackage PRESENT IF "the stateChange notification defined in Recommendation X.721 is supported by an instance of this class.";

REGISTERED AS {m3100ObjectClass 39};

**10) New subclause 3.5.x "Multipoint Cross-connection R1"**

*Add the following modified object class definition to a new subclause 3.5.x between existing subclauses 3.5.5 and 3.5.6:*

**mpCrossConnectionR1 MANAGED OBJECT CLASS**

**DERIVED FROM** mpCrossConnection;

**CHARACTERIZED BY**

mpCrossConnectionR1Package PACKAGE

**BEHAVIOUR**

mpCrossConnectionR1Behaviour BEHAVIOUR

**DEFINED AS**

"When the attribute value change notification package is present, the attributeValueChange notification shall be emitted when userLabel or redline changes value (when the attributes are present).

When the state change notification package is present, the stateChange notification shall be emitted when operationalState or administrativeState changes value.";;;

**CONDITIONAL PACKAGES**

userLabelPackage PRESENT IF "an instance supports it",

redlinePackage PRESENT IF "an instance supports it",

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.";

REGISTERED AS {m3100ObjectClass 36};

**11) New subclause 3.5.x "Group Termination Point R1"**

*Add the following new gtpR1 managed object class definition in a new subclause 3.5.x following existing subclause 3.5.4:*

**3.5.x Group Termination Point R1**

**gtpR1 MANAGED OBJECT CLASS**

**DERIVED FROM** gtp;

**CONDITIONAL PACKAGES**

createDeleteNotificationsPackage PRESENT IF "the objectCreation and objectDeletion notifications defined in Recommendation X.721 are supported by an instance of this class.";

REGISTERED AS {m3100ObjectClass 38};

**12) Subclause 3.5.6 "Named Cross Connection"**

*Remove the extraneous character in the CHARACTERIZED BY keyword.*

**13) New subclause 4.2x "Red line" (package)**

The redline attribute is included in a new package.

*Insert a new subclause 4.2x "Red line" after existing subclause 4.27 "Protected" with the following definition:*

**redlinePackage PACKAGE**  
**ATTRIBUTES**  
    **redline**       **GET-REPLACE;**  
**REGISTERED AS {m3100Package 42};**

**14) Subclause 5.26 "Equipment Holder Type"**

*Replace the equipmentHolderTypeBehaviour:*

**"The equipment holder type attribute indicates the type of equipment holder using a character string. The possible values for the character string may be bay, shelf, drawer, slot and rack."**

*with:*

**"The equipment holder type attribute indicates the type of equipment holder using a character string. The example values for the character string may be bay, shelf, drawer, slot and rack."**

**15) Subclause 5.32 "holder Status"**

*Add missing space after m3100Attribute in the REGISTERED AS clause to read:*

**REGISTERED AS {m3100Attribute 59};**

**16) Subclause 5.48 "serialNumber"**

*Add missing space after m3100Attribute in the REGISTERED AS clause to read:*

**REGISTERED AS {m3100Attribute 69};**

**17) Subclause 5.54 "Subordinate Circuit Pack Software Load attribute"**

The subordinateCircuitPackSoftwareLoad attribute is intended to designate software to be loaded to a contained circuit pack. The SEQUENCE OF construct in the ASN.1 syntax is important for specifying the order in which the software is to be downloaded. To make this clearer, the behaviour of the attribute is amended.

*Following:*

**"The choice of sequence of ObjectInstance identifies an ordered set of software instances."**

*insert:*

**"This ordered set can be used to specify the order in which the software is to be downloaded. It is recommended to note in the ICS if the ordering is significant."**

**18) New subclause 5.6x with the Type Text attribute definition**

*Insert a new subclause after subclause 5.66 "Transmission Characteristics" with the following definitions:*

**typeText ATTRIBUTE**  
    **WITH ATTRIBUTE SYNTAX ASN1DefinedTypesModule.TypeText;**  
    **MATCHES FOR EQUALITY, SUBSTRINGS;**  
    **BEHAVIOUR**  
        **typeTextBehaviour BEHAVIOUR**  
        **DEFINED AS "This attribute gives a textual description of the type of the resource";;**  
**REGISTERED AS {m3100Attribute 70};**

**19) Subclause 6.3 "Circuit Pack name binding"**

There are two name bindings for circuitPack from equipment holder – autoCreated and explicitlyCreated. The autoCreated name binding is applicable when the NE automatically creates

the circuit pack object; for instance, when the circuit pack is first equipped, in order to report an objectCreation notification to indicate this to the manager. The management system may delete this circuit pack and recreate a new one in order to plan the specific type of the circuit pack, using the explicitlyCreated name binding. To enable this, a new name binding is defined.

*Insert the following definitions:*

```
circuitPack-equipmentHolder-autoCreated-R1 NAME BINDING
  SUBORDINATE OBJECT CLASS circuitPack AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS equipmentHolder AND SUBCLASSES;
  WITH ATTRIBUTE equipmentId;
  BEHAVIOUR circuitPack-equipmentHolder-autoCreated-R1-Beh;
  DELETE
    ONLY-IF-NO-CONTAINED-OBJECTS
      generalErrorParameter;
REGISTERED AS {m3100NameBinding 37};
```

```
circuitPack-equipmentHolder-autoCreated-R1-Beh BEHAVIOUR
DEFINED AS
  "This name binding is used to name an instance of an circuitPack relative to another equipmentHolder instance. The creation of the circuitPack object is the result of inserting the physical circuit pack into the resource represented by the superior object.
```

The management system may delete this circuit pack and recreate a new one in order to plan the specific type of the circuit pack, using the explicitlyCreated name binding";

```
circuitPack-equipmentHolder-explicitlyCreated-R1 NAME BINDING
SUBORDINATE OBJECT CLASS circuitPack AND SUBCLASSES;
NAMED BY SUPERIOR OBJECT CLASS
  equipmentHolder AND SUBCLASSES;
WITH ATTRIBUTE equipmentId;
BEHAVIOUR circuitPack-equipmentHolder-explicitlyCreated-R1-Beh;
CREATE
  WITH-REFERENCE-OBJECT,
  WITH-AUTOMATIC-INSTANCE-NAMING
    createErrorParameter
    generalErrorParameter;
DELETE
  ONLY-IF-NO-CONTAINED-OBJECTS;
REGISTERED AS {m3100NameBinding 46};
```

*Add the circuitPack-equipmentHolder-explicitlyCreated-R1-Beh behaviour.*

```
circuitPack-equipmentHolder-explicitlyCreated-R1-Beh BEHAVIOUR
DEFINED AS
  "This name binding is used to name an instance of a circuitPack relative to another equipmentHolder instance. The creation of the circuitPack object is the result of system management protocol. If the circuitPackType is incompatible with the types supported by the equipmentHolder, the create request will result in a CMIP processing failure error. The generalErrorParameter is then used to report the error and may provide the value of the circuitPackType attribute. This parameter may be used for other CMIP processing failure if needed.";
```

## **20) Subclause 6.7 "Cross connection name binding"**

*Add the following name bindings including the generalErrorParameter:*

```
crossConnection-fabric-R1 NAME BINDING
  SUBORDINATE OBJECT CLASS crossConnection AND SUBCLASSES;
  NAMED BY SUPERIOR OBJECT CLASS fabric AND SUBCLASSES;
  WITH ATTRIBUTE crossConnectionId;
  BEHAVIOUR crossConnection-fabricBehaviour;
  DELETE
```

**ONLY-IF-NO-CONTAINED-OBJECTS**  
    **generalErrorParameter;**  
**REGISTERED AS {m3100NameBinding 39};**

**crossConnection-mpCrossConnection-R1 NAME BINDING**  
    **SUBORDINATE OBJECT CLASS crossConnection AND SUBCLASSES;**  
    **NAMED BY SUPERIOR OBJECT CLASS mpCrossConnection AND SUBCLASSES;**  
    **WITH ATTRIBUTE crossConnectionId;**  
    **BEHAVIOUR crossConnection-mpCrossConnectionBehaviour;**  
    **DELETE**

**ONLY-IF-NO-CONTAINED-OBJECTS**  
    **generalErrorParameter;**  
**REGISTERED AS {m3100NameBinding 40};**

## **21) Subclause 6.8 "Equipment" name binding definitions**

*Add the following name bindings including the generalErrorParameter:*

**equipment-managedElement-R1 NAME BINDING**  
    **SUBORDINATE OBJECT CLASS equipment AND SUBCLASSES;**  
    **NAMED BY SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES;**  
    **WITH ATTRIBUTE equipmentId;**  
    **BEHAVIOUR equipmentNameBindingBehaviour;**  
    **CREATE**  
        **WITH-REFERENCE-OBJECT,**  
        **WITH-AUTOMATIC-INSTANCE-NAMING**  
        **generalErrorParameter;**  
    **DELETE**  
    **ONLY-IF-NO-CONTAINED-OBJECTS**  
        **generalErrorParameter;**  
**REGISTERED AS {m3100NameBinding 41};**

**equipment-equipment-R1 NAME BINDING**  
    **SUBORDINATE OBJECT CLASS equipment AND SUBCLASSES;**  
    **NAMED BY SUPERIOR OBJECT CLASS equipment AND SUBCLASSES;**  
    **WITH ATTRIBUTE equipmentId;**  
    **BEHAVIOUR equipmentNameBindingBehaviour;**  
    **CREATE**  
        **WITH-REFERENCE-OBJECT,**  
        **WITH-AUTOMATIC-INSTANCE-NAMING**  
        **generalErrorParameter;**  
    **DELETE**  
    **ONLY-IF-NO-CONTAINED-OBJECTS**  
        **generalErrorParameter;**  
**REGISTERED AS {m3100NameBinding 42};**

## **22) Subclause 6.9 "Equipment Holder" name binding definitions**

Back planes may be modelled as equipment holders. In some cases, the back planes may be provided with circuitry that allows, for example, for the emission of alarms so that they may be considered as normal cards (thus modelled with the circuitPack object class), which are inserted into the appropriate equipment holder (e.g. a subrack or a rack).

*Replace:*

**equipmentHolder-equipmentHolderBeh BEHAVIOUR**  
**DEFINED AS**  
    **"This name binding is used to name an instance of an equipmentHolder relative to another equipmentHolder instance. When an equipmentHolder contains another equipmentHolder, the superior equipmentHolder shall not contain any circuitPack.";**

*with:*

**equipmentHolder-equipmentHolderBeh BEHAVIOUR  
DEFINED AS**

**"This name binding is used to name an instance of an equipmentHolder relative to another equipmentHolder instance.";**

**23) Subclause 6.10 "Event Forwarding Discriminator" name binding definitions**

*Add the following name binding including the generalErrorParameter:*

**eventForwardingDiscriminator-managedElement-R1 NAME BINDING**

**SUBORDINATE OBJECT CLASS**

**"Recommendation X.721:1992": eventForwardingDiscriminator AND SUBCLASSES;**

**NAMED BY SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES;**

**WITH ATTRIBUTE "Recommendation X.721:1992": discriminatorId;**

**BEHAVIOUR equipmentNameBindingBehaviour;**

**CREATE**

**WITH-REFERENCE-OBJECT,**

**WITH-AUTOMATIC-INSTANCE-NAMING**

**generalErrorParameter;**

**DELETE**

**ONLY-IF-NO-CONTAINED-OBJECTS**

**generalErrorParameter;**

**REGISTERED AS {m3100NameBinding 43};**

**24) Subclause 6.11 "Fabric" name binding definitions**

*Add the following name binding including the generalErrorParameter:*

**fabric-managedElement-R1 NAME BINDING**

**SUBORDINATE OBJECT CLASS fabric AND SUBCLASSES;**

**NAMED BY SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES;**

**WITH ATTRIBUTE fabricId;**

**CREATE**

**WITH-REFERENCE-OBJECT,**

**WITH-AUTOMATIC-INSTANCE-NAMING**

**generalErrorParameter;**

**DELETE**

**ONLY-IF-NO-CONTAINED-OBJECTS**

**generalErrorParameter;**

**REGISTERED AS {m3100NameBinding 44};**

**25) Subclause 6.14 "Managed Element" name binding definitions**

The managedElementCreateBehaviour behaviour has been defined twice.

*Modify the managedElement-managedElementComplex name binding to reference the "older" definition as follows:*

**managedElement-managedElementComplex NAME BINDING**

**SUBORDINATE OBJECT CLASS managedElement AND SUBCLASSES;**

**NAMED BY**

**SUPERIOR OBJECT CLASS managedElementComplex AND SUBCLASSES;**

**WITH ATTRIBUTE managedElementId;**

**BEHAVIOUR managedElementCreateBehaviour;**

**REGISTERED AS {m3100NameBinding 34};**

**26) Subclause 6.14 "Managed Element" name binding definitions**

The existing M.3100 managedElement-managedElementComplex name binding does not allow for a management system to create and delete managedElement object instances that are contained in a managedElementComplex. This functionality may be required when the managedElementComplex

represents an adapter device and each managedElement represents a managed device which is accessed through the adapter device.

*Add the following definitions to subclause 6.14:*

**managedElement-managedElementComplex-explicitlyCreated NAME BINDING**  
**SUBORDINATE OBJECT CLASS** managedElement AND SUBCLASSES;  
**NAMED BY**  
**SUPERIOR OBJECT CLASS** managedElementComplex AND SUBCLASSES;  
**WITH ATTRIBUTE** managedElementId;  
**BEHAVIOUR** managedElement-managedElementComplex-explicitlyCreateBehaviour;  
**CREATE**  
    **WITH-REFERENCE-OBJECT,**  
    **WITH-AUTOMATIC-INSTANCE-NAMING**  
        createErrorParameter;  
**DELETE**  
    **DELETES-CONTAINED-OBJECTS;**  
**REGISTERED AS** {m3100NameBinding 45};

**managedElement-managedElementComplex-explicitlyCreateBehaviour BEHAVIOUR**  
**DEFINED AS**  
    "This name binding is used to name the managedElement object instance relative to a managedElementComplex object. The managedElement object is explicitly created by management protocol. Creation of the managedElement may trigger the automatic creation of certain inherent objects contained within the managedElement, and the deletion of the managedElement may trigger automatic deletion of all objects contained in the managedElement.";

## 26a) Subclause 6.15 "Managed Element Complex" name binding definitions

The existing M.3100 does not allow for managedElementComplex to be named by network.

*Add the following definitions to subclause 6.15:*

**managedElementComplex-network NAME BINDING**  
**SUBORDINATE OBJECT CLASS** managedElementComplex AND SUBCLASSES;  
**NAMED BY SUPERIOR OBJECT CLASS**  
    networkR1 AND SUBCLASSES;  
**WITH ATTRIBUTE** managedElementComplexId;  
**BEHAVIOUR** managedElementComplex-networkBeh;  
**REGISTERED AS** {m3100NameBinding 56};

**managedElementComplex-networkBeh BEHAVIOUR**  
**DEFINED AS**  
    "This name binding is used to name the managedElementComplex object to a network object. The managedElementComplex object is not created or deleted by system management protocol.";

## 27) New subclause 6.23 "Additional name bindings for managed object classes in the X.700-series"

Recommendations in the X.700-series normally name from system rather than from managedElement. Hence, additional name bindings are required in order for TMN applications to make use of the definitions in these Recommendations.

*Add the following name bindings:*

-- Rec. X.745

**testActionPerformer-managedElement NAME BINDING**  
**SUBORDINATE OBJECT CLASS** "Rec. X.745 (1993)" : testActionPerformer AND SUBCLASSES;  
**NAMED BY**



**SUPERIOR OBJECT CLASS   managedElement AND SUBCLASSES;  
 WITH ATTRIBUTE       "Rec. X.745 (1993)" : testActionPerformerId;  
 CREATE  
     WITH-REFERENCE-OBJECT,  
     WITH-AUTOMATIC-INSTANCE-NAMING;  
 DELETE  
     ONLY-IF-NO-CONTAINED-OBJECTS;  
 REGISTERED AS {m3100NameBinding 47};**

**testObject-testActionPerfomer NAME BINDING  
 SUBORDINATE OBJECT CLASS "Rec. X.745 (1993)" : testObject AND SUBCLASSES;  
 NAMED BY  
 SUPERIOR OBJECT CLASS "Rec. X.745 (1993)" : testActionPerformer AND SUBCLASSES;  
 WITH ATTRIBUTE       "Rec. X.745 (1993)" : testObjectId;  
 CREATE  
     WITH-REFERENCE-OBJECT,  
     WITH-AUTOMATIC-INSTANCE-NAMING;  
 DELETE  
     ONLY-IF-NO-CONTAINED-OBJECTS;  
 REGISTERED AS {m3100NameBinding 48};**

*-- Rec. X.738*

**simpleScanner-managedElement NAME BINDING  
 SUBORDINATE OBJECT CLASS "Rec. X.738 (1993)" : simpleScanner AND SUBCLASSES;  
 NAMED BY  
 SUPERIOR OBJECT CLASS   managedElement AND SUBCLASSES;  
 WITH ATTRIBUTE       "Rec. X.739 (1993)" : scannerId;  
 CREATE  
     WITH-REFERENCE-OBJECT,  
     WITH-AUTOMATIC-INSTANCE-NAMING;  
 DELETE  
     ONLY-IF-NO-CONTAINED-OBJECTS;  
 REGISTERED AS {m3100NameBinding 49};**

*-- Rec. X.742*

**usageMeteringControl-managedElement NAME BINDING  
 SUBORDINATE OBJECT CLASS       "Rec. X.742 (1995)" : usageMeteringControlObject  
     AND SUBCLASSES;  
 NAMED BY SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES;  
 WITH ATTRIBUTE       "Rec. X.742 (1995)" : controlObjectId;  
 DELETE  
     DELETES-CONTAINED-OBJECTS;  
 REGISTERED AS {m3100NameBinding 50};**

*-- Rec. X.746*

**scheduler-managedElement NAME BINDING  
 SUBORDINATE OBJECT CLASS "Rec. X.746 (1995)" : scheduler AND SUBCLASSES;  
 NAMED BY SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES;  
 WITH ATTRIBUTE  
     "Rec. X.746 (1995)" : schedulerID;  
 CREATE  
     WITH-REFERENCE-OBJECT,  
     WITH-AUTOMATIC-INSTANCE-NAMING  
         "Rec. X.738 (1993)":conflictingPackagesRequestedError;  
 DELETE  
     ONLY-IF-NO-CONTAINED-OBJECTS;  
 REGISTERED AS {m3100NameBinding 51};**

**applicationProcess-managedElement NAME BINDING**

**SUBORDINATE OBJECT CLASS "Rec. X.723 (1993)" : applicationProcess AND SUBCLASSES ;  
NAMED BY**

**SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES ;**

**WITH ATTRIBUTE "Rec. X.723 (1993)" : applicationProcessId ;**

**REGISTERED AS {m3100NameBinding 54};**

**subsystem-managedElement NAME BINDING**

**SUBORDINATE OBJECT CLASS "Rec. X.723 (1993)" : subsystem AND SUBCLASSES ;  
NAMED BY**

**SUPERIOR OBJECT CLASS managedElement AND SUBCLASSES ;**

**WITH ATTRIBUTE "Rec. X.723 (1993)" : subsystemId ;**

**REGISTERED AS {m3100NameBinding 55};**

**28) Subclause 7.4 "Connect" (action)**

In the modified ConnectInformation (see item 36) the redline and the name for the cross connection both appear.

*Add the following text to the behaviour of the connect action to clarify the usage of the parameters:*

**"When the namedCrossConnection parameter is present, a namedCrossConnection (or subclass of namedCrossConnection) instance will be created. When userLabel and/or redline is present, crossConnectionR1 (or subclass of crossConnectionR1) instance will be created. When none of these parameters are present, it is up to the local system to deduce which instance will be created. The namedCrossConnection field shall not be used together with either the userLabel field or the redline field in the same action request."**

**29) New subclause 9.3 "General Error"**

The defect report addressed missing error reporting capabilities in the existing model. While CMISE errors allows many error reporting situations to be reported, only one CMISE error, processing failure, allows additional specific errors to be defined. ProcessingFailure is further augmented by the SpecificError field where operation/object specific error information is included. This field is registered using the parameter template.

This defect report proposed that new name bindings be added to facilitate exchanging additional information for the CMIP processing error failure code. In order to minimize the impact for the existing definitions, enhancements are proposed for name bindings only. This will not require reregistering object class definitions.

*Add to new subclause 9.3 the following definitions:*

**generalErrorParameter PARAMETER**

**CONTEXT SPECIFIC-ERROR;**

**WITH SYNTAX ASN1DefinedTypesModule.GeneralError;**

**BEHAVIOUR**

**generalErrorParameterBehaviour BEHAVIOUR DEFINED AS**

**"If an error or set of errors occurs that cannot be described by existing CMISE errors or other error parameters, the error(s) will be communicated using this parameter, with the error or errors described by a cause code and optionally provided text.**

**An attributeList can be provided if the error condition can be further described by the state of the object attributes.**

**Related object(s) can also be provided. The related objects may contribute to the condition that does not allow the operation to take place. An example would be if objects are configured for a particular service offering that is in conflict with the service offering that is being provisioned.";;**

**REGISTERED AS {m3100Parameter 3};**

### 30) Subclause 10.2 "ASN.1 Module"

The ASN.1 module InformationFramework {joint-iso-ccitt ds(5) modules(1) informationFramework(1)} originally defined in Recommendation X.501 (1988) has been replaced with a new version defined using ASN.1 (1992). The original version is now contained in Recommendation X.711 (1997).

*Modify:*

```
IMPORTS
RDNSequence
FROM InformationFramework {joint-iso-ccitt ds(5) modules(1)
informationFramework(1)}
```

*to read:*

```
IMPORTS
RDNSequence
FROM InformationFramework {joint-iso-ccitt ds(5) modules(1) informationFramework(1)}
-- Note – This Recommendation imports RDNSequence from CCITT Rec. X.501 (1988). The
-- specification for this syntax can now be found in an informative annex of
-- Rec. X.711 (1997)/ISO/IEC 9596-1:1998.
```

### 31) Subclause 10.2 "ASN.1 Module"

The new type GeneralError references AttributeList to be imported from Recommendation X.721.

*Replace:*

```
"ProbableCause, AdministrativeState, AvailabilityStatus
FROM Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module (2) 1} ;"
```

*with:*

```
"ProbableCause, AdministrativeState, AvailabilityStatus, AttributeList
FROM Attribute-ASN1Module {joint-iso-ccitt ms(9) smi(3) part2(2) asn1Module (2) 1} ;"
```

### 32) Subclause 10.2 "ASN.1 Module"

*Replace:*

```
m3100Parameter OBJECT IDENTIFIER ::= {m3100InformationModel package(5)}
```

*with:*

```
m3100Parameter OBJECT IDENTIFIER ::= {m3100InformationModel parameter(5)}
```

### 33) Subclause 10.2 "ASN.1 Module"

*Add the following object identifier assignments for characteristic information:*

```
vpCI CharacteristicInformation ::= {characteristicInfo 17}
vcCI CharacteristicInformation ::= {characteristicInfo 18}
```

```
e0CI CharacteristicInformation ::= {characteristicInfo 19}
e1CI CharacteristicInformation ::= {characteristicInfo 20}
e2CI CharacteristicInformation ::= {characteristicInfo 21}
e3CI CharacteristicInformation ::= {characteristicInfo 22}
e4CI CharacteristicInformation ::= {characteristicInfo 23}
```

### 34) Subclause 10.2 "ASN.1 Module"

Spelling mistake in the list of probableCause value definitions.

*Modify:*

```
"lowTemperatue ProbableCause ::= localValue : 130"
```

to read:

**"lowTemperature ProbableCause ::= localValue : 130"**

### 35) Subclause 10.2 "ASN.1 module"

Recommendation M.3100 defines localValues for probable causes to be used by applications using the TMN application context as described in clause 11.

Remove comments stating reservation of probable cause values.

*Remove the following statement:*

*"-- Service ProblemType is for further study"*

*Add the following to the list of probable cause allocations:*

<b>receiveFailure</b>	<b>ProbableCause ::= localValue : 17</b>
<b>transmitFailure</b>	<b>ProbableCause ::= localValue : 18</b>
<b>modulationFailure</b>	<b>ProbableCause ::= localValue : 19</b>
<b>demodulationFailure</b>	<b>ProbableCause ::= localValue : 20</b>
<b>broadcastChannelFailure</b>	<b>ProbableCause ::= localValue : 21</b>
<b>connectionEstablishmentError</b>	<b>ProbableCause ::= localValue : 22</b>
<b>invalidMessageReceived</b>	<b>ProbableCause ::= localValue : 23</b>
<b>localNodeTransmissionError</b>	<b>ProbableCause ::= localValue : 24</b>
<b>remoteNodeTransmissionError</b>	<b>ProbableCause ::= localValue : 25</b>
<b>routingFailure</b>	<b>ProbableCause ::= localValue : 26</b>
<i>-- Values 27-50 are reserved for communications alarm related probable causes</i>	

<b>realTimeClockFailure</b>	<b>ProbableCause ::= localValue : 70</b>
<i>-- An equipment alarm to be issued if the system detects that the real time clock has failed.</i>	
<b>antennaFailure</b>	<b>ProbableCause ::= localValue : 71</b>
<b>batteryChargingFailure</b>	<b>ProbableCause ::= localValue : 72</b>
<b>diskFailure</b>	<b>ProbableCause ::= localValue : 73</b>
<b>frequencyHoppingFailure</b>	<b>ProbableCause ::= localValue : 74</b>
<b>iODeviceError</b>	<b>ProbableCause ::= localValue : 75</b>
<b>lossOfSynchronisation</b>	<b>ProbableCause ::= localValue : 76</b>
<b>lossOfRedundancy</b>	<b>ProbableCause ::= localValue : 77</b>
<b>powerSupplyFailure</b>	<b>ProbableCause ::= localValue : 78</b>
<b>signalQualityEvaluationFailure</b>	<b>ProbableCause ::= localValue : 79</b>
<b>tranceiverFailure</b>	<b>ProbableCause ::= localValue : 80</b>
<i>-- Values 81-100 are reserved for equipment alarm related probable causes</i>	

<b>coolingSystemFailure</b>	<b>ProbableCause ::= localValue : 134</b>
<b>externalEquipmentFailure</b>	<b>ProbableCause ::= localValue : 135</b>
<b>externalPointFailure</b>	<b>ProbableCause ::= localValue : 136</b>
<i>-- Values 137-150 are reserved for environmental alarm related probable causes</i>	

<b>lossOfRealTime</b>	<b>ProbableCause ::= localValue : 157</b>
<i>-- A processing error alarm to be issued if the system detects that it has lost the time in</i>	
<i>-- the real time clock but the clock itself is working. This could happen e.g. during a power</i>	
<i>-- cut in a small NE which does not have battery backup for the real time clock.</i>	

<b>reinitialized</b>	<b>ProbableCause ::= localValue : 158</b>
<i>-- A processing error alarm to be issued after the system has reinitialised. This will indicate</i>	
<i>-- to the management systems that the view they have of the managed system may no longer</i>	
<i>-- be valid. Usage example: The managed</i>	
<i>-- system issues this alarm after a reinitialization with severity warning to inform the</i>	
<i>-- management system about the event. No clearing notification will be sent.</i>	

<b>applicationSubsystemFailure</b>	<b>ProbableCause ::= localValue : 159</b>
<b>configurationOrCustomisationError</b>	<b>ProbableCause ::= localValue : 160</b>
<b>databaseInconsistency</b>	<b>ProbableCause ::= localValue : 161</b>

<b>fileError</b>	<b>ProbableCause ::= localValue : 162</b>
<b>outOfMemory</b>	<b>ProbableCause ::= localValue : 163</b>
<b>softwareError</b>	<b>ProbableCause ::= localValue : 164</b>
<b>timeoutExpired</b>	<b>ProbableCause ::= localValue : 165</b>
<b>underlyingResourceUnavailable</b>	<b>ProbableCause ::= localValue : 166</b>
<b>versionMismatch</b>	<b>ProbableCause ::= localValue : 167</b>

*-- Values 168-200 are reserved for processing error alarm related probable causes.*

<b>bandwidthReduced</b>	<b>ProbableCause ::= localValue : 201</b>
<b>congestion</b>	<b>ProbableCause ::= localValue : 202</b>
<b>excessiveErrorRate</b>	<b>ProbableCause ::= localValue : 203</b>
<b>excessiveResponseTime</b>	<b>ProbableCause ::= localValue : 204</b>
<b>excessiveRetransmissionRate</b>	<b>ProbableCause ::= localValue : 205</b>
<b>reducedLoggingCapability</b>	<b>ProbableCause ::= localValue : 206</b>
<b>systemResourcesOverload</b>	<b>ProbableCause ::= localValue : 207</b>

### 36) Subclause 10.2 "ASN.1 Module"

The parameter of the connect action needs to be updated with the optional userLabel and redline fields. The existing namedCrossConnection cannot be used because it always contains both the cross-connection name and the redline.

*Update the ConnectionInformation ASN.1 type definition as follows:*

```

ConnectInformation ::= SEQUENCE OF SEQUENCE {
    itemType CHOICE {
        unidirectional [0] ConnectionType,
        bidirectional [1] ConnectionTypeBi,
        addleg [2] AddLeg
    },
    administrativeState AdministrativeState OPTIONAL,
    namedCrossConnection [3] NamedCrossConnection OPTIONAL,
    userLabel [4] UserLabel OPTIONAL,
    redline [5] Boolean OPTIONAL
}

```

### 37) Subclause 10.2 "ASN.1 Module"

*Add the following productions to the ASN.1 module:*

```

GeneralError ::= SEQUENCE OF SEQUENCE {
    cause GeneralErrorCause,
    details GraphicString OPTIONAL,
    relatedObjects [0] SET OF ObjectInstance OPTIONAL,
    attributeList [1] AttributeList OPTIONAL
}

```

```

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

```

<b>objectInIncompatibleState</b>	<b>GeneralErrorCause ::= localValue : 1</b>
<b>noValidRelatedObject</b>	<b>GeneralErrorCause ::= localValue : 2</b>
<b>involvedInOffering</b>	<b>GeneralErrorCause ::= localValue : 3</b>
<b>serviceNotSupported</b>	<b>GeneralErrorCause ::= localValue : 4</b>
<b>provisioningOrderConflict</b>	<b>GeneralErrorCause ::= localValue : 5</b>
<b>equipmentFailure</b>	<b>GeneralErrorCause ::= localValue : 6</b>
<b>maxNumberExceeded</b>	<b>GeneralErrorCause ::= localValue : 7</b>
<b>containedObjects</b>	<b>GeneralErrorCause ::= localValue : 8</b>

*-- ObjectInIncompatibleState is used to specify that the object is in a state  
-- that does not allow the operation. Details of the state should be  
-- provided.*

- *NoValidRelatedObject* is used to specify related objects that do not exist in the MIB.
- *InvolvedInOffering* is used to identify object(s) that are already involved in a conflicting service offering.
- *ServiceNotSupported* is used to indicate that the operation is attempting to initiate a service that is not supported by the equipment.
- *ProvisioningOrderConflict* is used to identify that a service is being provisioned in an order that is not supported by the equipment.
- *EquipmentFailure* is used to indicate that an equipment failure as occurred during the operation.
- *MaxNumberExceeded* is used to indicate that requested create operation cannot be completed as the maximum number of instances are reached.
- *ContainedObjects* is used to indicate that requested delete operation cannot be completed as there are contained instances.

### **38) Subclause 10.2 "ASN.1 Module"**

A disconnection action may fail because the TPs referenced in the action are not already connected to each other. This situation is address by existing problem cause values.

*Add the following ASN.1 definition to the ProblemCause ASN.1 value assignments:*

**notAlreadyConnected      ProblemCause ::= integerValue : 13**

### **39) Subclause 10.2 "ASN.1 Module"**

*Add the following ASN.1 definition:*

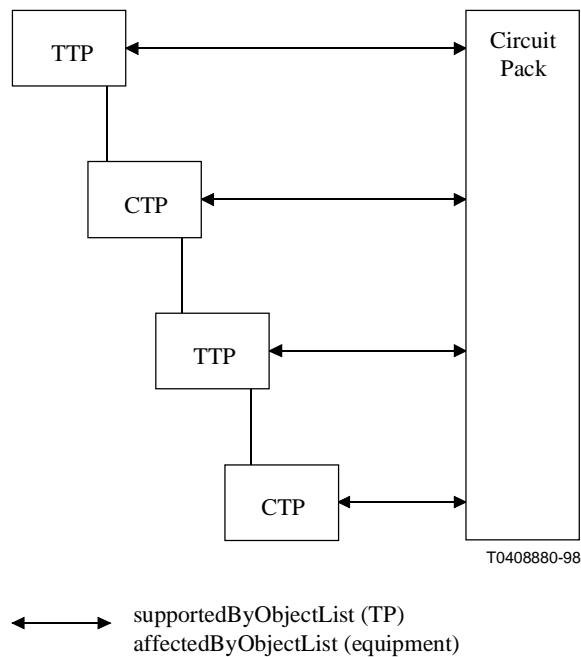
**TypeText ::= GraphicString**

### **40) Subclause I.2 "Use of Supported by Object List"**

This subclause contains text to provide guidance on the use of this attribute. The text is extended to cover the specific example of termination points and the relationship with circuit packs.

*Add the following text and figure before subclause I.3:*

"All termination point instances will be related to equipment when they are providing service. The supportedByObjectList attribute in the TP object instance will point to an instance of circuit pack, and the circuit pack will point to all related TPs using the affectedObjectList attribute. The TPs indicate relationships with each other through the connectivity pointer attributes and naming, the supportedByObjectList attribute is generally not be used to indicate relationships between TPs (although exceptions exists such as in Recommendation I.751). This is illustrated in the following figure:"



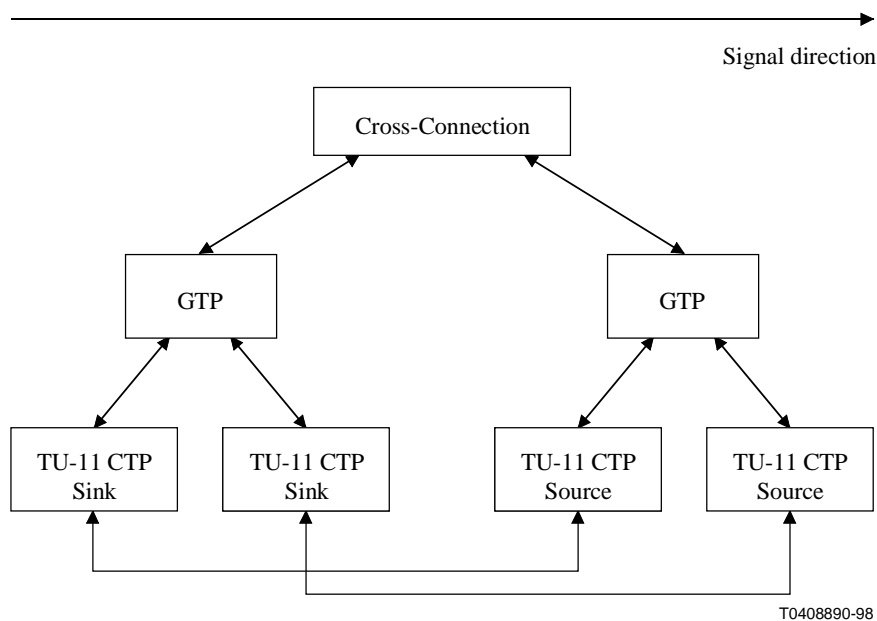
**41) Subclause I.5.1 "Point-to-point cross-connection between two termination points"**

*Update the user guideline to read:*

"Figure I.2 provides an example of the simplest type of cross-connection: a point-to-point cross-connection between two termination points. The Upstream and Downstream connectivity pointers in each termination point point to the other termination point, the cross-connection object pointers in the termination points point to the cross-connection object and the From and To termination pointers in the cross-connection object point to the termination points."

**42) Subclause I.5.6 "Unidirectional cross-connection using GTPs"**

The information flow in Figure I.6 should be from left to right and the sink and source labels should be corrected (note the different semantic of CTPs and TTPs) as follows:



**Figure I.6/M.3100 – Unidirectional 2xTU-11 point-to-point-cross-connection with proposed model**

**43) New subclause I.10 "System initialization"**

*Add a new subclause I.10 as follows:*

**"I.10 System initialization**

Having object creation notifications as mandatory may give the impression that when the system is set up for the first time and the management information tree is created, the management network will be flooded with object creation notifications of any autoinstantiated objects. However, at that point the event forwarding discriminators are either not yet created or if created will need manager to configure the destination for the event reports sent. Therefore in practise there will be no flood of event reports as the event forwarding discriminators are not yet fully functional.

In normal operation after the event forwarding discriminators have been configured with appropriate destinations, the management systems set as destinations will get the object creation notifications when e.g. new equipment is added. In case there is a need to do major re-equipping to the system then the management system may always suspend the event forwarding for the time of the equipping in order to avoid the flood of notifications, and resume the event forwarding after the changes have been done.

When a system is initialized, it is expected that either the network object, the managedElement object, or the managedElementComplex object is auto-created, so that subtending objects be either auto-created or created by the manager."

**44) New subclause I.11 "Use of equipmentHolder acceptableCircuitPackList attribute"**

*Add a new subclause I.11 as follows:*

**"I.11 Use of equipmentHolder acceptableCircuitPackList attribute**

This attribute can be used to know what circuit packs can be supported by a given equipmentHolder. When coming-up initially, the default value should be the list of all circuit packs the equipmentHolder can support. At that time, the manager can query the agent to retrieve the set of circuit packs supported by the equipmentHolder.

In cases where a software addition allows the agent to support new types of packs over the same equipmentHolder, an attribute value change notification will then be sent out."



## 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
<b>Series M</b>	<b>TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits</b>
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 communication
Series Y	Global information infrastructure
Series Z	Programming languages