



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.722

Amendment 3
(08/97)

**SERIES X: DATA NETWORKS AND OPEN SYSTEM
COMMUNICATION**

OSI management – Structure of Management Information

Information technology – Open Systems
Interconnection – Structure of management
information: Guidelines for the definition of
managed objects

**Amendment 3: Guidelines for the use of Z in
formalizing the behaviour of managed objects**

ITU-T Recommendation X.722 – Amendment 3

(Previously CCITT Recommendation)

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

PUBLIC DATA NETWORKS	X.1-X.199
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	X.200-X.299
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	X.300-X.399
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	X.600-X.699
Networking	X.600-X.629
Efficiency	X.630-X.649
Naming, Addressing and Registration	X.650-X.679
Abstract Syntax Notation One (ASN.1)	X.680-X.699
OSI MANAGEMENT	X.700-X.799
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	X.730-X.799
SECURITY	X.800-X.849
OSI APPLICATIONS	X.850-X.899
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 10165-4**ITU-T RECOMMENDATION X.722****INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION –
STRUCTURE OF MANAGEMENT INFORMATION: GUIDELINES FOR
THE DEFINITION OF MANAGED OBJECTS****AMENDMENT 3****Guidelines for the use of Z in formalizing the behaviour of managed objects****Summary**

This Amendment to CCITT Rec. X.722 | ISO/IEC 10165-4 contains an illustrative example that demonstrates current best practice in the use of the Z formal description language for specifying Managed Object behaviour. It aims to establish a common basis and understanding of this particular formal approach which will help achieve consistency in similar developments. It should provide a useful starting point for GDMO users wishing to use Z to improve their behaviour specifications.

Formal specifications of MO behaviour are valuable since they are clear and unambiguous. The act of producing a formal specification forces the details of the behaviour to be analysed closely. Thus it can also be used as a tool to identify and correct ambiguities in a specification that will remain in natural language.

This Amendment contains a technical guide on the use of the Z language for defining the behaviour of managed objects which support OSI management interworking. It is informative and not normative. It does not require Formal Definition Techniques (FDTs) to be used to specify MO behaviour. If FDTs are to be used, it does not require Z to be used; other languages such as SDL are also suitable.

Source

The ITU-T Recommendation X.722, Amendment 3 was approved on the 9th of August 1997. The identical text is also published as ISO/IEC International Standard 10165-4.

**Documents**

Document Sharing Hub

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) Table of contents	1
2) Subclause 2.1	1
3) New subclause 2.3.....	1
4) New Annex B	1
Annex B – Guidelines for the use of Z in formalizing the behaviour of Managed Objects	2

INTERNATIONAL STANDARD

ITU-T RECOMMENDATION

**INFORMATION TECHNOLOGY – OPEN SYSTEMS INTERCONNECTION –
STRUCTURE OF MANAGEMENT INFORMATION: GUIDELINES FOR
THE DEFINITION OF MANAGED OBJECTS****AMENDMENT 3****Guidelines for the use of Z in formalizing the behaviour of managed objects****1) Table of contents**

Add the following reference to the table of contents:

Annex B – Guidelines for the use of Z in formalizing the behaviour of managed objects

2) Subclause 2.1

Add the following reference to 2.1:

- CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2:1992, *Information technology – Open Systems Interconnection – Systems Management: State management function.*

3) New subclause 2.3

Add a new subclause as follows:

2.3 Additional references

- ISO/IEC 13568:¹⁾, *Information technology – Z specification language.*
 - ¹⁾ Presently at the stage of draft.

4) New Annex B

Add a new Annex B, as follows:

Annex B

Guidelines for the use of Z in formalizing the behaviour of Managed Objects

(This annex does not form an integral part of this Recommendation | International Standard)

B.1 Introduction

This annex contains a technical guide on the use of the Z language for defining the behaviour of managed objects which support OSI management interworking. It is informative and not normative. It does not require Formal Definition Techniques (FDTs) to be used to specify MO behaviour. If FDTs are to be used, it does not require Z to be used; other languages such as SDL are also suitable. Even if Z is to be used, other ways of specifying MO behaviour are possible.

Formal specifications of MO behaviour can be directly valuable because they are clear and unambiguous. The act of producing a formal specification forces the details of the behaviour to be analysed closely. Thus, it can also be used as a tool to identify and correct ambiguities which might go undetected in a specification relying solely on natural language. For these reasons formal specification can be useful to improve behaviour specification.

This annex contains an illustrative example that demonstrates current best practice. It aims to establish a common basis and understanding of this particular formal approach which will help achieve consistency in similar developments. It should provide a useful starting point for GDMO users wishing to use Z to improve their behaviour specifications.

It is aimed at an audience familiar with the basic concepts of managed object specification using the GDMO templates, and the Z language.

For the remainder of this annex, the terms “managed object” and “MO” will be used to refer to a managed object class definition given using the GDMO templates.

B.2 Language issues

The Z notation is a formal specification notation based on set theory and predicate calculus. It possesses sufficient expressive power to be able to describe single classes of managed objects.

However, there exists no notion of encapsulation in Z. A Z specification typically consists of a model of some state and a collection of operations to modify the state. There is no method built into Z to parcel the state and its operations up into a single module and re-use it in another specification. The consequence of this becomes apparent when it becomes necessary to describe managed objects which inherit variables and behaviour from other managed object class definitions.

The effect of inheritance can be achieved by the technique of schema inclusion at the expense of some clarity. In all other respects Z is suitable for expressing single classes of managed objects.

B.3 What needs to be translated

The behaviour definitions, or parts there of, need to be translated from the informal description into Z. The extent to which the remaining parts of the GDMO templates need to be formalized depends largely on the needs of the specifier.

The GDMO templates already include a semi-formal definition of data types in ASN.1. It is possible to write a Z specification using these ASN.1 definitions as a basis for types used in the Z specification, and this saves a significant amount of work.

However, if a specification is written in this way, then it makes it a greater task for the specifier to ensure that it is syntactically correct. Without Z specifications of the ASN.1 definitions, it is not possible to use existing Z tools which provide support for checking the syntax and static semantics of a Z specification.

In summary, it is possible to improve the behaviour definitions by using Z without re-writing the ASN.1 data types, but there is a significant benefit to be gained by a full translation of the ASN.1 data types into Z. Examples of how to convert ASN.1 Basic Types into Z are provided in B.7.1.

B.3.1 From GDMO templates to Z

This subclause contains general guidelines of how to go about translating a managed object from its informal description as given in this Recommendation | International Standard into Z. It should be stressed at the outset that such a translation can only be carried out informally since a formal translation would require, as a minimum, that both the source and target languages be formal.

Moreover, as with any mapping between two distinct languages, there is bound to be some mismatch between their constructs. The problem multiplies when one of the languages happens to be informal or to include informal components.

In this subclause some of the main features of the templates defined in this Recommendation | International Standard are listed together with the ways in which they differ from or correspond to constructs in Z. In the process, general ways of resolving the mismatch or advice on how they may be tackled individually on an ad-hoc basis is offered.

This annex will concentrate on what is necessary to describe the behaviour of a managed object. Additional information on how to convert ASN.1 types is provided in B.6.

B.3.2 Datatypes

The first step is to rewrite the datatypes from this Recommendation | International Standard as Z types. ASN.1 provides the usual facilities of datatyping but its constructors are biased towards the description of datastreams communicated between systems.

In ASN.1, the type constructors are defined as forms of list. In Z, types are sets. Although it is possible to model the ASN.1 type constructors as sequences in Z, it is sometimes more natural to consider the operations available on the ASN.1 types and to map them to Z types which more clearly describe their structure. The ASN.1 sequence and set types can be mapped to Z tuples. The ASN.1 sequence-of type can be mapped to a Z sequence. The ASN.1 set-of type can be mapped to a Z set.

ASN.1 includes special support for encoding, such as type labels and default values. This does not need to be represented in Z since it doesn't affect the behaviour definition.

Subclause B.6.2 provides additional information on how to convert ASN.1 types.

B.3.3 MO Attributes

Managed objects are defined to have certain management attributes. These attributes have a datatype defined in ASN.1. They are assigned object identifiers. They also may have a matches-for property. Two ways to model such attributes have been proposed:

- simple attribute types; and
- attribute types as schemas.

The simplest is to represent the MO attribute within the MO as a Z variable with the appropriate datatype. Then separately we will need a constant definition which represents the object identifier of that attribute. This constant will be related to the actual attribute by convention only. We can use the actual fixed matches-for property when matching operations are defined for that attribute. An example of this is given in B.6.3.

It is also possible to encapsulate all these properties of an attribute in a single schema type which will then be the type of the Z variable modelling the MO attribute. Thus, the schema will include the value of the attribute as well as the object identifier and the matches-for property if any. An example of this is given in B.6.4. Where matching rules other than equality are required, it is possible to define the matches-for parameter as a Z relation over the type of the value of the attribute. This allows the formal representation of arbitrary ad-hoc matching rules, which may be important for scoping, filtering and object selection.

It is difficult to model ASN.1 type ANY in Z. One case where this is common is to give lists of attribute values. Thus, a fully formal model will probably require a Z free type combining the attribute types already defined. An example of this can be found in B.6.1 and B.6.5.

Object identifiers are formally modelled by a given set.

[OBJECTID]