INTERNATIONAL TELECOMMUNICATION UNION

# ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

# Z.450

(10/2003)

SERIES Z: LANGUAGES AND GENERAL SOFTWARE ASPECTS FOR TELECOMMUNICATION SYSTEMS

Quality – Quality aspects of protocol-related Recommendations

# Quality aspects of protocol-related Recommendations

ITU-T Recommendation Z.450

ITU-T Z-SERIES RECOMMENDATIONS

**LANGUAGES AND GENERAL SOFTWARE ASPECTS FOR TELECOMMUNICATION SYSTEMS**

| | |
|---|---|
| FORMAL DESCRIPTION TECHNIQUES (FDT) | |
| Specification and Description Language (SDL) | Z.100–Z.109 |
| Application of formal description techniques | Z.110–Z.119 |
| Message Sequence Chart (MSC) | Z.120–Z.129 |
| Extended Object Definition Language (eODL) | Z.130–Z.139 |
| Testing and Test Control Notation (TTCN) | Z.140–Z.149 |
| User Requirements Notation (URN) | Z.150–Z.159 |
| PROGRAMMING LANGUAGES | |
| CHILL: The ITU-T high level language | Z.200–Z.209 |
| MAN-MACHINE LANGUAGE | |
| General principles | Z.300–Z.309 |
| Basic syntax and dialogue procedures | Z.310–Z.319 |
| Extended MML for visual display terminals | Z.320–Z.329 |
| Specification of the man-machine interface | Z.330–Z.349 |
| Data-oriented human-machine interfaces | Z.350–Z.359 |
| Human-computer interfaces for the management of telecommunications networks | Z.360–Z.369 |
| QUALITY | |
| Quality of telecommunication software | Z.400–Z.409 |
| **Quality aspects of protocol-related Recommendations** | **Z.450–Z.459** |
| METHODS | |
| Methods for validation and testing | Z.500–Z.519 |
| MIDDLEWARE | |
| Distributed processing environment | Z.600–Z.609 |

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

# ITU-T Recommendation Z.450

## Quality aspects of protocol-related Recommendations

### Summary

This Recommendation provides guidelines on the quality aspects of protocol-related ITU-T Recommendations with the objective of:

1) improving the quality of protocol-related Recommendations; and

2) helping to improve the interoperability of products based on these Recommendations.

In order to encourage the widest possible use of ITU-T protocol-related Recommendations, it is important that these Recommendations are of the highest possible quality. Several aspects of Quality that play an important role for the users of these ITU-T Recommendations have been selected for this purpose. These are: readability, completeness, correctness, consistency, unambiguity, implementability and testability.

Annex A contains a quality checklist to identify the level of compliance of a Recommendation with the quality aspects.

### Source

ITU-T Recommendation Z.450 was approved by ITU-T Study Group 17 (2001-2004) under the ITU-T Recommendation A.8 procedure on 29 October 2003.

## FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. 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 Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 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.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

## INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. 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, 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.

## CONTENTS

## Introduction

Following WTSC-96 endorsement of the importance of quality in ITU-T Recommendations, TSAG formed a Correspondence Group to develop a guideline document entitled *Guidelines on quality aspects of protocol-related Recommendations.* At the TSAG meeting in September 1998, the work of the Correspondence Group was approved as Supplement 1 to the A-series Recommendations. SG 10 contributed to the work of the TSAG Correspondence Group and subsequently approved a new Question with the objective of developing a new Recommendation based on Supplement 1. This Recommendation is the result of this work. All Study Groups should use this Recommendation when developing new protocol standards or evaluating existing ones.

# ITU-T Recommendation Z.450

## Quality aspects of protocol-related Recommendations

### 1 Scope

This Recommendation covers the readability, correctness, consistency, unambiguity, implementability and testability aspects of quality of protocol-related Recommendations and specifies a process that can assist in the development of higher-quality Recommendations.

### 2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

– ITU-T Recommendation X.290 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – General concepts*.

– ITU-T Recommendation X.291 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – Abstract test suite specification*.

– ITU-T Recommendation X.292 (2002), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – The Tree and Tabular Combined Notation (TTCN)*.

– ITU-T Recommendation X.293 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – Test realization*.

– ITU-T Recommendation X.294 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – Requirements on test laboratories and clients for the conformance assessment process*.

– ITU-T Recommendation X.295 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – Protocol profile test specification*.

– ITU-T Recommendation X.296 (1995), *OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications – Implementation conformance statements*.

– ITU-T Recommendation X.680 (2002), *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*.

– ITU-T Recommendation X.681 (2002), *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification*.

– ITU-T Recommendation X.682 (2002), *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification*.

– ITU-T Recommendation X.683 (2002), *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications*.

- ITU-T Recommendation X.690 (2002), *Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*.
- ITU-T Recommendation X.691 (2002), *Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)*.
- ITU-T Recommendation X.692 (2002), *Information technology – ASN.1 encoding rules: Specification of Encoding Control Notation (ECN)*.
- ITU-T Recommendation X.693 (2001), *Information technology – ASN.1 encoding rules: XML Encoding Rules (XER)*.
- ITU-T Recommendation Z.100 (2002), *Specification and Description Language (SDL)*.
- ITU-T Recommendation Z.105 (2003), *SDL combined with ASN.1 modules (SDL/ASN.1)*.
- ITU-T Recommendation Z.107 (1999), *SDL with embedded ASN.1*.
- ITU-T Recommendation Z.109 (1999), *SDL combined with UML*.
- ITU-T Recommendation Z.110 (2000), *Criteria for use of formal description techniques by ITU-T*.
- ITU-T Recommendation Z.120 (1999), *Message sequence chart (MSC)*.
- ITU-T Recommendation Z.130 (2003), *Extended Object Definition Language (eODL): Techniques for distributed software component development – Conceptual foundation, notations and technology mappings*.
- ITU-T Recommendation Z.140 (2003), *Testing and Test Control Notation version 3 (TTCN-3): Core language*.
- ITU-T Recommendation Z.141 (2003), *Testing and Test Control Notation version 3 (TTCN-3): Tabular presentation format*.
- ITU-T Recommendation Z.142 (2003), *Testing and Test Control Notation version 3 (TTCN-3): Graphical presentation format*.
- ITU-T Recommendation Z.150 (2003), *User Requirements Notation (URN): Language requirements and framework*.

## 3 Definitions

This Recommendation defines the following terms:

**3.1 formal description technique (FDT)**: Standardized languages, e.g., SDL, MSC, TTCN and ASN.1, and other techniques, e.g., GDMO, intended for formal specification of system behaviour, structure and data.

**3.2 specification and description language (SDL)**: Internationally standardized formal language for specifying and describing real-time systems.

**3.3 message sequence charts (MSC)**: Internationally standardized language for describing sequences of messages interchanged between system components and their environment.

**3.4 abstract test suite (ATS)**: A test suite composed of abstract test cases.

**3.5 abstract test case**: A complete and independent specification of the actions required to achieve a specific test purpose, defined at the level of abstraction of a particular Abstract Test Method, starting in a stable testing state and ending in a stable testing state. This specification may involve one or more consecutive or concurrent connections.

**3.6 abstract test method**: The description of how an Implementation Under Test is to be tested, given an appropriate level of abstraction to make the description independent of any particular realization of a Means of Testing, but with enough detail to enable abstract test cases to be specified for this test method.

**3.7 executable test suite (ETS)**: A test suite composed of executable test cases.

**3.8 executable test case**: A realization of an abstract test case.

**3.9 tree and tabular combined notation (TTCN)**: A standardized notation for specifying conformance test suites in a manner that is independent of test methods, layered software architectures or protocols and which reflects the abstract testing methodology defined in ITU-T Recs X.290 and X.291.

**3.10 user requirements notation (URN)**: A high-level requirements notation that consists of two languages, Use Case Maps (UCM) and Graphical Requirements Language (GRL), which serve to cover initial phases of analysis and requirements capture.

**3.11 extended object definition language (eODL)**: A superset of IDL (Interface Description Language) covering the definition of objects with their interfaces, as well as the distribution of objects.

**3.12 validation**: A process of checking a specification to ensure that it is syntactically and semantically correct and represents the intended behaviour.

**3.13 conformance clause**: A part of a standard or Recommendation which identifies what in the standard or Recommendation should be met in order to conform to the standard or Recommendation.

**3.14 specification**: Prescription of the design of an aspect of a product or a set of products.

**3.15 implementation conformance statement (ICS)**: A statement made by the supplier of an implementation or system claimed to conform to a given specification, stating which capabilities have been implemented. The ICS can take several forms: protocol ICS, profile ICS, profile-specific ICS, managed object ICS and information object ICS.

**3.16 implementation extra information for testing (IXIT)**: A statement made by the supplier or an implementor of an IUT which contains or references all of the information (in addition to that given in ICS) related to the IUT and its testing environment, which will enable the test laboratory to run an appropriate test suite against the IUT. An IXIT can take several forms: protocol IXIT, profile IXIT, profile-specific IXIT, information object IXIT and TMP implementation statement.

**3.17 test tool**: Hardware and/or software, excluding the test suite itself, used to carry out or assist in carrying out the testing required.

**3.18 means of testing (MOT)**: Hardware and/or software, and the procedures for its use, including the executable test suite itself, used to carry out the testing required.

**3.19 implementation under test (IUT)**: An implementation of one or more requirements specifications, being that part of a system, which is to be studied by testing.

**3.20 interoperability testing**: Testing to assess the ability of two or more systems to exchange information and to make mutual use of the information that has been exchanged.

**3.21 reference implementation**: An implementation of one or more standards or specifications, against which a Means of Testing and test tools for those standards or specifications are tested for the purpose of validation of those Means of Testing or test tools.

## 4 Abbreviations

This Recommendation uses the following abbreviations:

| | |
|---|---|
| ASN.1 | Abstract Syntax Notation One |
| ATS | Abstract Test Suite |
| eODL | Extended Object Definition Language |
| ETS | Executable Test Suite |
| FD | Formal Definition |
| FDT | Formal Description Technique |
| GRL | Graphical Requirements Language |
| IUT | Implementation Under Test |
| MOT | Means of Testing |
| MSC | Message Sequence Chart |
| PICS | Protocol Implementation Conformance Statement |
| PIXIT | Protocol Implementation Extra Information for Testing |
| SDL | Specification and Description Language |
| TTCN-2 | Tree and Tabular Combined Notation version 2 |
| TTCN-3 | Testing and Test Control Notation version 3 |
| UCM | Use Case Map |
| URN | User Requirements Notation |

## 5 Conventions

The acronym "TTCN" will refer to either TTCN-2 defined in ITU-T Rec. X.292 (2002) or TTCN-3 defined in ITU-T Rec. Z.140 (2003).

## 6 Developing high-quality Recommendations

### 6.1 Overview of the quality aspects

In preparing new Recommendations or making an assessment of the quality of existing Recommendations, the following aspects should be considered:

- **Readability** – This requires that a Recommendation is well-structured and can be easily read and understood. This implies good use of the natural languages with adequate use of figures, examples and references.
- **Completeness** – A Recommendation should be complete in the sense that it includes all parts necessary for its implementation. This includes clearly identified, precise and unambiguous specifications, and may include SDL specifications, MSC requirements, data and protocol format descriptions in ASN.1, the Abstract Test Suite, the Protocol Implementation Conformance Statement Proforma and the Protocol Implementation Extra Information for Testing Proforma.
- **Correctness** – Specifications should be error-free and represent the intended behaviour; this especially applies to the parts of the Recommendation expressed using formal description techniques.

- **Consistency** – Different parts of the Recommendation or a set of related Recommendations should not contain contradictory information.
- **Unambiguity** – A Recommendation should not be subject to different interpretations; implementations from different suppliers should be compatible; this is especially important where interoperability is required in a multi-vendor environment.
- **Implementability** – A specification should not be unnecessarily complex. This means that the specification should permit only a minimum number of optional features, avoiding those options whose sole purpose is to achieve consensus. The specification should be based on the best technical solution.
- **Testability** – If products are based on a specification contained in a Recommendation, it should be possible to test whether the product conforms to the specification.

### 6.2 Upfront identification

Each Study Group should identify upfront during the formulation of the Study Group Questions and subsequently in the development phase of Recommendations, which quality aspects need to be fulfilled based on market requirements. A checklist to facilitate this identification is contained in Annex A.

As a general guideline, all Recommendations that address the interoperability of equipment, networks, protocols or services should consider all quality aspects addressed here.

### 6.3 Formal description techniques

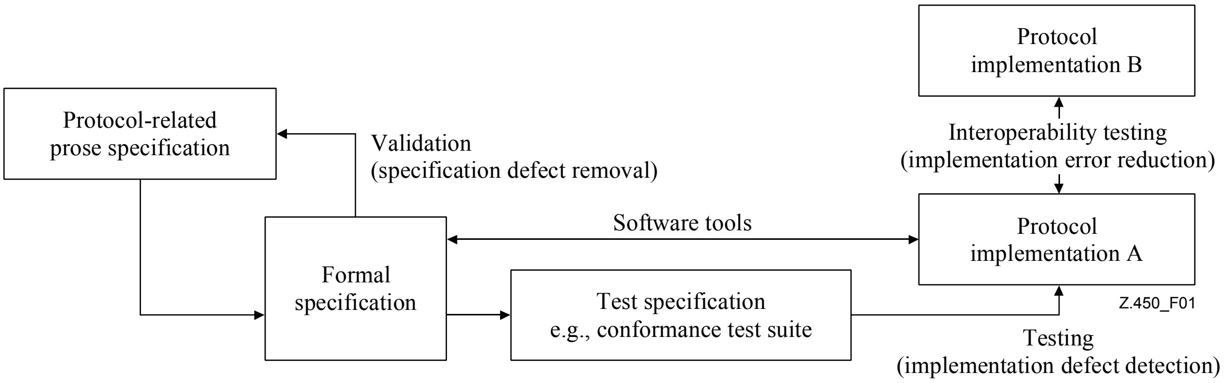
Formal specifications should be made available in machine-readable format to allow companies to use their facilities for simulation, validation, automatic code generation or other types of automatic processing should they wish to do so.

Guidelines on the use of FDTs are contained in ITU-T Rec. Z.110. It provides criteria for their use and should, therefore, be taken as a reference in conjunction with this Recommendation.

Whenever a discrepancy between a natural language description and a FD, or between two FDs, is detected, the discrepancy should be resolved by changing the natural language description or the FDs without necessarily giving preference to one over the other(s). The goal should be the elimination of ambiguity and discrepancies.

### 6.4 Overview of the process

The quality of Recommendations is closely connected with the use of Formal Description Techniques (FDTs) and the use of computer-based tools. The main components of the quality process are formal specifications, validation of specifications and testing of prototype implementations. Figure 1 illustrates the complete process.


Protocol implementation B
Protocol-related prose specification
Validation (specification defect removal)
Interoperability testing (implementation error reduction)
Software tools
Protocol implementation A
Formal specification
Test specification e.g., conformance test suite
Z.450_F01
Testing (implementation defect detection)

NOTE 1 – Not all of the illustrated processes in this figure are required for all protocols.
NOTE 2 – Several of the relations indicated by the arrows could be supported by software tools.

**Figure 1/Z.450 – Flow diagram to illustrate the quality process**

The main component of the process is the formal specification expressed using an FDT. In this process, a precise and unambiguous formal specification may first be produced from the natural language specification. The FDTs should be widely available, allowing the process to be largely automated using commercially available computer-based tools.

The second key component of the process is the specification validation step in which specification defects are detected and removed. Once the specification is verified, it can be used to produce two or more implementations by automatic generation of the implementation code to assess the ease of implementation and subsequently test the conformance and interoperability. The purpose of this is to ensure that the specification is error-free and describes the intended behaviour. At the same time, implementability of the specification can be verified and the conformance test suite developed and validated.

It is preferable to have the conformance test suites generated directly from the formal specification using software tools so that the individual test cases will be traceable to the specification.

## 7 Quality checks

Before approving each new protocol-related Recommendation, the quality checklist should be completed and made available to the Study Group meeting. This list will identify the level of compliance with the quality aspects. The list should be included in the report to make the level of quality transparent to readers.

## 8 Specific guidelines

### 8.1 Readability

Recommendations should be clear, well-structured and easily read and understood. This applies both to the natural language text and the formal specification. This applies to all Recommendations irrespective of whether they are protocol-related or not.

Specifically, the functional purpose of the Recommendation should be clearly stated in natural language.

## 8.2 Completeness of Recommendations

### 8.2.1 Conformance clause

In many cases, specification of mandatory requirements and optional features may be scattered throughout the Recommendation, often mixed with tutorial or explanatory text and examples. It is essential to identify and isolate each normative item in the Recommendation. For this purpose, all normative requirements and optional features should be summarized in a special section, the conformance clause, of a Recommendation indicating individual references to the parts in the Recommendation where the full specification is given.

### 8.2.2 Tools for completeness checking

For checking completeness the following tools may be used:

a) *Protocol Implementation Conformance Statement (PICS)*

For each relevant Recommendation, PICS may be developed for use by the implementor of the Recommendation to declare what mandatory and optional features affecting interoperability have been implemented and which have not.

b) *Protocol Implementation Extra Information for Testing (PIXIT)*

For each relevant Recommendation, a PIXIT may be specified to facilitate testing.

### 8.2.3 Testing methodology

The checklist should indicate which testing methodology has been used. One such methodology is specified in the X.290-series of Recommendations.

## 8.3 Correctness

### 8.3.1 Validation of specifications

The process of validation is illustrated in Figure 1. It consists of implementation, simulation and testing using commercial computer-based tools. This process requires formal descriptions of protocols and the use of tools to establish correctness of the specification prior to approval.

Examples of guidance for validation of specifications are given in ITU-T Rec. Z.110 and the Methodology Supplement to ITU-T Rec. Z.100. The Working Group members through pilot projects within their own organizations could do validation.

### 8.3.2 Abstract Test Suite (ATS)

For each relevant Recommendation, a corresponding Abstract Test Suite expressed in the standardized (ITU-T Rec. X.292) Tree and Tabular Combined Notation version 2 (TTCN-2) or Testing and Test Control Notation version 3 (TTCN-3) (ITU-T Rec. Z.140) and in compliance with the X.290-series of Recommendations may be developed.

### 8.3.3 Publication phase

Specific steps should be taken through the publication phase to ensure the correctness of the published Recommendation.

### 8.3.4 Maintenance

Procedures are currently in place (Resolution 1) for identifying and correcting errors in approved Recommendations.

## 8.4 Consistency

Different parts of the Recommendation should not contain contradictory information. The goal should be the elimination of ambiguity and discrepancies.

### 8.5 Unambiguity

A Recommendation should not be subject to different interpretations; implementations from different suppliers should be compatible; this is especially important where interoperability is required in a multi-vendor environment.

### 8.6 Implementability

#### 8.6.1 Reduction of the number of options

Options reduce the chances of interoperability. The number of options should be reduced so that only the essential options are included and there are no redundancies.

#### 8.6.2 Best technical solutions

Each Recommendation should be based on the principle of 'best technical solution' rather than a compromise solution to achieve consensus.

### 8.7 Testability

Testing interfaces, such as points of control and observation, should be defined for example according to a standardized testing methodology and an appropriate Abstract Test Suite should be specified for each protocol. Reduced complexity of Recommendations is also essential for testability of implementations.

## 9 Use of tools in developing Recommendations

### 9.1 Early error detection

The use of appropriate and suitable software tools such as protocol validation, syntax check tools, etc., in conjunction with formal description techniques during the development of Recommendations is essential for early error detection and the development of a common view during the consensus process. The use of tools makes errors visible early, thereby resulting in substantial savings in later efforts to uncover and correct errors.

### 9.2 Timely Recommendations

The use of automated techniques is essential for timely production of Recommendations. These tools reduce the total time required to develop a Recommendation and the associated Abstract Test Suites.

### 9.3 Lower cost

Appropriate steps should be taken to reduce the cost to develop Recommendations. The use of computer-aided techniques and automation tools during the development of Recommendations will reduce the overall cost of developing Recommendations.

# Annex A

# Quality checklist for protocol-related Recommendations

## A.1 General checklist

The checklist in Table A.1 is provided to assist the developers of new protocol-related Recommendations in determining which clauses of the Quality Guidelines have been addressed.

**Table A.1/Z.450 – Quality checklist**

| Item | Quality aspect | Clause | Addressed | | |
|---|---|---|---|---|---|
| 1 | Readability | 8.1 | Yes | No | N/A |
| 2 | Completeness | 8.2 | Yes | No | N/A |
| 3 | Conformance clause | 8.2.1 | Yes | No | N/A |
| 4 | Completeness checking | 8.2.2 | Yes | No | N/A |
| 5 | Testing methodology | 8.2.3 | Yes | No | N/A |
| 6 | Correctness | 8.3 | Yes | No | N/A |
| 7 | Formal descriptions | 8.3.1 | Yes | No | N/A |
| 8 | Abstract Test Suite | 8.3.2 | Yes | No | N/A |
| 9 | Publication phase | 8.3.3 | Yes | No | N/A |
| 10 | Maintenance | 8.3.4 | Yes | No | N/A |
| 11 | Consistency | 8.4 | Yes | No | N/A |
| 12 | Unambiguity | 8.5 | Yes | No | N/A |
| 13 | Implementability | 8.6 | Yes | No | N/A |
| 14 | Reduced number of options | 8.6.1 | Yes | No | N/A |
| 15 | Best technical solution | 8.6.2 | Yes | No | N/A |
| 16 | Testability | 8.7 | Yes | No | N/A |
| 17 | Use of tools | 9 | Yes | No | N/A |
| 18 | Early error detection | 9.1 | Yes | No | N/A |
| 19 | Timely Recommendations | 9.2 | Yes | No | N/A |
| 20 | Lower cost | 9.3 | Yes | No | N/A |

## A.2 Specific checklist

In addition to the quality checklist of Table A.1, a more specific set of questions are needed to:

- identify upfront what quality level should be aimed at;
- assess whether the quality goals have been met;
- distinguish between various means/levels of qualities;
- enable quality control across these means.

## A.3 Specific questions

### Natural language texts

1) Is readability of the text validated?

2) Does any label refer to items that do not appear in the Description Technique specification?

3) Are the required measures completed on the missing references?

**Illustrations**

4) Is the readability of the illustrations validated?

5) Do the illustrations use Description Technique notations only?

6) Does any label refer to items that do not appear in the natural language texts?

7) Does any label refer to items that do not appear in the Description Technique specification?

8) Are the required measures completed on the missing references?

**Description Techniques**

9) Is use of Description Techniques required for the Recommendation?

10) Has ITU-T Rec. Z.110 been used to identify relevant Description Techniques?

11) Is one Description Technique only used?

12) Is a combination of Description Techniques used?

13) If yes, which techniques are used?

14) Are alphanumeric notations provided for the Description Techniques?

15) Are graphic notations provided for the Description Techniques?

16) Is the syntax of the Description Techniques formally specified?

17) Is the semantics of the Description Techniques formally specified?

18) Are mappings between the used Description Techniques formally specified?

19) Are the Description Techniques standardized?

20) If yes, which technique by which organization?

**Description Technique specification**

21) Is the Description Technique specification well-structured?

22) Is completeness of the specification validated?

23) What testing methodology has been used for completeness validation?

24) Was the correctness of the specification verified?

25) Is consistency within the Description Technique specification validated?

26) What testing methodology has been used for consistency validation?

27) Is consistency to natural language texts and illustrations validated?

28) Is the specification unambiguous?

**Completeness**

29) Is the conformance clause specified in the standard?

30) Is the Protocol Implementation Conformance Statement (PICS) proforma specified?

31) Is the conformance Abstract Test Suite specified?

32) Is the conformance testing methodology specified?

33) Is the Protocol Implementation Extra Information for Testing (PIXIT) proforma specified?

**Implementability**

34) Are all not strictly needed options removed?

35) Is the best technical solution specified?

36) Is a computable algorithm provided for the specification?

**Testability**

37) Are test specifications needed for the Recommendation?

38) Are interoperability tests needed?

39) Are abstract test suites provided for the specification?

40) Are the abstract test suites produced automatically from the specifications?

41) Are testing facilities offered for implementations?

**Implementation testing**

42) Are implementation tests needed for the Recommendation?

43) What implementations are provided?

44) What implementations are tested by using the abstract test suites?

45) Is module testing undertaken?

46) Is system testing undertaken?

47) Is user testing undertaken?

48) Is interoperability testing undertaken?

49) What performance tests have been undertaken?

50) Are the performance measurements satisfactory?

# SERIES OF ITU-T RECOMMENDATIONS

| | |
|---|---|
| Series A | Organization of the work of 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 | Cable networks and 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, Internet protocol aspects and Next Generation Networks |
| **Series Z** | **Languages and general software aspects for telecommunication systems** |

Printed in Switzerland
Geneva, 2004