

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Z.111

(04/2016)

SERIES Z: LANGUAGES AND GENERAL SOFTWARE
ASPECTS FOR TELECOMMUNICATION SYSTEMS

Formal description techniques (FDT) – Application of
formal description techniques

Notations and guidelines for the definition of ITU-T languages

Recommendation ITU-T Z.111

ITU-T

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

User Requirements Notation (URN) Z.150–Z.159

Testing and Test Control Notation (TTCN) Z.160–Z.179

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-machine interfaces for the management of telecommunications networks Z.360–Z.379

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

Processing environment architectures Z.600–Z.609

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

Recommendation ITU-T Z.111

Notations and guidelines for the definition of ITU-T languages

Summary

Recommendation ITU-T Z.111 provides meta-grammars for ITU-T Recommendations that define ITU-T languages in the ITU-T X.680 series and the ITU-T Z series of Recommendations on languages for specification, implementation, modelling and testing. This allows the description of these meta-grammars that define the abstract or concrete grammar (syntax, constraints and semantics) of languages without having to repeat the meta-grammar (such as lexical naming rules, or the description of Backus-Naur Form syntax) as a preamble or annex to each language definition.

This Recommendation draws common elements from the meta-grammars of various languages, covering issues such as common lexical rules, the use of a universal character set, and syntax and constraint description for languages at both the abstract and concrete level.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Z.111	2008-11-13	17	11.1002/1000/9621
2.0	ITU-T Z.111	2016-04-29	17	11.1002/1000/12862

Keywords

ITU-T languages, meta-grammars, abstract grammar, concrete grammar, syntax, constraints, semantics.

* To access the Recommendation, type the URL <http://handle.itu.int/> in the address field of your web browser, followed by the Recommendation's unique ID. For example, <http://handle.itu.int/11.1002/1000/11830-en>.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). 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, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2016

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

		Page
1	Scope.....	1
	1.1 Objective.....	1
	1.2 Application	1
2	References.....	1
3	Definitions	2
4	Abbreviations and acronyms	2
5	Conventions	2
	5.1 Grammars	2
	5.2 Basic definitions	2
	5.3 Presentation style.....	2
	5.4 Metalanguages	5
	Annex A – Use of graphical meta-symbols	20
	Appendix I – Mapping from the metamodel presentation to the textual presentation:	
	High-level description	21
	Bibliography.....	23

Introduction

Scope-objective: Notations and guidelines for the definition of formal languages are provided.

Coverage: The main features of this Recommendation are notations for defining the abstract and concrete syntaxes of languages, and a common structure for defining languages.

Applications: This Recommendation should be applied to new and possibly to revised formal language Recommendations.

Status/Stability: The notation and guidelines given are stable and have been used on some of the existing language Recommendations, such as Recommendations the ITU-T Z.100 series and ITU-T Z.151. There is scope for further guidelines, extensions to the notations given, algorithms for conversions between the textual and metamodel presentations of abstract grammars, mappings to XML-based concrete syntaxes for interchange formats, and the possibility of further formal notations, such as a formal notation for specifying constraints.

Associated work: This Recommendation is generally associated with the study of formal languages for telecommunications applications in ITU-T, in particular the languages defined by the X.680 series, Z.100 series, Z.120 series, Z.150 series and Z.160/Z.170 series of ITU-T Recommendations.

This Recommendation is also related to the unified modelling language (UML) and meta object facility (MOF) work of the object management group (OMG).

Background: Before the introduction of this Recommendation, the language Recommendations defined by ITU-T used different ways of describing the syntax and semantics of languages. This hinders the understanding, verifiability, and maintainability of these languages, and prevents their simple harmonization. Some language Recommendations have used grammars in Backus-Naur Form (BNF) to define concrete syntaxes and, in some cases, abstract syntaxes. More recent language definitions have used MOF metamodels to capture several aspects of languages. In both approaches, there has been a separation of abstract grammar from the concrete notation with a defined relationship between them. Each approach has its benefits and drawbacks. MOF metamodels are appealing due to their graphical nature (where associations and inherited concepts are explicit), whereas BNF grammars are easily analyzable by tools and there is considerable experience in their use.

This Recommendation gives guidelines for the definition of MOF-based metamodels describing languages in a way that is compatible with the approaches based on BNF grammars.

Traditional BNF-based grammars usually lack some of the capabilities of the MOF approach such as inheritance. MOF and UML-based meta-metamodels contain many features that make metamodels unnecessarily complex, difficult to understand semantically, and difficult to map to BNF grammars. This Recommendation focuses on a subset of modelling features that is expressive enough to describe language metamodels and that is isomorphic between the two approaches.

Recommendation ITU-T Z.111

Notations and guidelines for the definition of ITU-T languages

1 Scope

This Recommendation provides notation and guidelines for the definition of formal languages defined in new and optionally in revised language Recommendations that define formal languages or description techniques (such as those listed in clause 3.1.2 of the Recommendation "Criteria for use of formal description techniques by ITU-T" [b-ITU-T Z.110]).

1.1 Objective

The objective is to provide a basis for a common structure and meta-grammar notations to be used in formal language Recommendations, so that it is not necessary for each language Recommendation to describe its structure and the meta-grammars used: instead each language Recommendation can contain a reference to this Recommendation. By using a common structure and meta-grammar notations, it is also easier to integrate the different formal languages, therefore making it easier to use the languages together both with and without the support of tools. The use of a common structure and meta-grammar notations makes it easier to build tools that combine the ITU-T languages with each other and other notations. The use of a common structure and meta-grammar notations in different formal language Recommendations also makes it easier to understand several formal language Recommendations, because the structure and meta-grammar only has to be learnt once.

1.2 Application

When a new Recommendation is being drafted for a formal language, the application of this Recommendation should be applied. If it is decided not to apply this Recommendation, the reasons should be stated in the formal language Recommendation.

When a Recommendation for an existing formal language is being revised, the application of this Recommendation should be considered taking into account costs and benefits. If this Recommendation is not applied or is applied partially, the revised formal language Recommendation should at least contain a statement explaining that the structure and meta-language notations of the formal language Recommendation predated the approval of this Recommendation.

The structure and meta-grammar notations given in this Recommendation are also likely to be useful for the definition of other formal languages, not just formal languages defined in ITU-T Recommendations. If it is later decided to consider making a Recommendation for such a formal language, the prior application of this Recommendation will be a benefit.

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 T.55] Recommendation ITU-T T.55 (2008), *Use of the universal multiple-octet coded character set (UCS)*.
<http://www.itu.int/rec/T-REC-T.55>

[OMG UML] *OMG Unified Modeling Language (OMG UML): Superstructure, version 2.5, formal/15-03-01 of the Object Management Group (OMG).*
<<http://www.omg.org/spec/UML/2.5/PDF> >

3 Definitions

No specific definition is provided in this Recommendation.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations:

BNF	Backus-Naur Form
MOF	Meta Object Facility
OCL	Object Constraint Language
OMG	Object Management Group
SDL	Specification and Description Language
UML	Unified Modeling Language
XML	eXtensible Markup Language

5 Conventions

If a language Recommendation conforms to this Recommendation, it conforms to the conventions defined in this clause and therefore these do not need to be repeated in the language Recommendation.

5.1 Grammars

A description only conforms to a Recommendation if it conforms to both the *Concrete grammar* and *Abstract grammar* of the corresponding language definition: that is, the description must be both recognizable as the language defined in the Recommendation and have the same meaning as defined by the *Semantics* in the Recommendation. If further concrete grammars are defined (in additional clauses, annexes or Recommendations), each of the concrete grammars has a definition of its own syntax and of its relationship to the abstract grammar (that is, how to transform into the abstract syntax). Using this approach, there is only one definition of the semantics of a language: the semantics of each of the concrete grammars is identified via its relationship to the abstract grammar. This approach also ensures that any further grammars are equivalent.

For some constructs of the concrete grammar, there may be no directly equivalent abstract syntax. In these cases, a *Model* is given for the transformation from concrete syntax into the concrete syntax of other constructs that (directly or indirectly via further models) have an abstract syntax. Items that have no mapping to the abstract syntax (such as comments) do not have any formal meaning.

5.2 Basic definitions

Some general concepts and conventions are used throughout a language Recommendation; their definitions are given in the following subclauses.

5.3 Presentation style

The following presentation style is used to separate the different language issues under each topic.

5.3.1 Division of text

A language Recommendation is organized by language features described by an optional introduction, which by convention shall be informative rather than normative (see clause 5.3.2), followed by titled enumeration items for:

- a) *Abstract grammar* – Described by abstract syntax (either a textual grammar or a graphical metamodel) and static conditions (that is, static constraints) for a model in the language to be well-formed for an abstract model where artifacts of the concrete syntax are ignored.
- b) *Concrete grammar* – Described by the graphical syntax, static conditions and rules for the graphical syntax to be well-formed (including drawing rules), the relationship of this syntax with the abstract syntax.
- c) *Semantics* – Gives meaning to a construct, provides the properties it has, the way in which it is interpreted and any dynamic conditions that have to be fulfilled for the construct to behave well in the sense of the language defined.
- d) *Model* – Gives the mapping for notations that do not have a direct abstract syntax and modelled in terms of other concrete syntax constructs. A notation that is modelled by other constructs is known as a shorthand, and is considered to be derived syntax for the transformed form.

To avoid misunderstandings, a language Recommendation that follows this Recommendation and uses informative introductions shall include a clause that states:

- Where a section has introductory text followed by a titled enumeration items (such as *Abstract grammar*, *Concrete grammar*, *Semantics*, *Model*) the introductory text before the first titled enumeration item in a clause is for information only, and is treated as if written as a NOTE (that is, informative not normative), except if it is explicitly stated as normative.

5.3.2 Titled enumeration items

Where a topic in the language definition has an introduction followed by a titled enumeration item, the introduction is considered to be an informal part of the Recommendation presented only to aid understanding and not to make that Recommendation complete.

If there is no text for a titled enumeration item (*Abstract grammar*, *Concrete grammar*, *Semantics*, *Model*), the whole item is omitted.

It is permitted for a Recommendation to define additional titled enumeration items, such as an *Example* section.

The remainder of this clause describes the other special formalisms used in each titled enumeration item and the titles used. It can also be considered as an example of the typographical layout of first-level titled enumeration items defined above where this text is part of an introductory section.

a) *Abstract grammar*

The notations for defining the abstract grammar are defined in clause 5.4.1. In the rest of this clause, the two notations are taken as equivalent. A rule in the textual abstract syntax (see clause 5.4.1.1) is equivalent to a model element in a graphical metamodel (see clause 5.4.1.2). The terms 'model element' and 'graphical metamodel' can be substituted for 'rule' and 'abstract syntax' respectively.

If the titled enumeration item *Abstract grammar* is omitted, then there is no additional abstract syntax for the topic being introduced, and the concrete syntax shall map onto the abstract syntax defined by another numbered text clause.

The rules in the abstract syntax may be referred to from any of the titled enumeration items by use of the rule name in italics.

The rules in the formal notation may be accompanied by paragraphs that define conditions which shall be satisfied by a well-formed piece (text and/or graphics) in the language being defined and which can be checked without dynamic interpretation of that piece. The static conditions at this point refer only to the abstract syntax. The static conditions shall be expressed in natural language, preferably supplemented by expression of the constraints in a more formal constraint language. Where it is possible to capture a constraint in the abstract syntax rather than by a static condition, this is preferable provided it does not make the syntax too complex. Static conditions, which are only relevant for the concrete syntax, are defined with the concrete syntax. Together with the abstract syntax, the static conditions for the abstract syntax define the abstract grammar of the language.

b) *Concrete grammar*

The concrete syntax shall be specified in the extended Backus-Naur Form of syntax description defined in clause 5.4.2, except where a graphical metamodel has been used for the abstract grammar, in which case it is permitted to extend the abstract metamodel to include concrete attributes. The use of natural language to define concrete syntax or definition by examples should be avoided.

The concrete syntax is accompanied by paragraphs defining the static conditions which must be satisfied in a well-formed definition and which can be checked without interpretation of a definition. Static conditions (if any) for the abstract grammar also apply. Where it is possible to capture a constraint in the concrete syntax rather than by a static condition, this is preferable provided it does not make the syntax too complex. The concrete grammar should not repeat constraints present in the abstract grammar.

In many cases there is a simple relationship between the concrete and abstract grammar, because the concrete syntax rule is simply represented by a single rule (or metamodel element) in the abstract grammar. When the same name is used in the abstract grammar and concrete syntax in order to signify that they represent the same concept, the text "<x> in the concrete syntax represents X in the abstract grammar" is implied in the language description and therefore does not need to be stated explicitly. In this context, spaces and hyphens are treated as equivalent, case is ignored but underlined semantic sub-categories (see clause 5.4.2) are significant, so that <integer name> represents *Integer-name* in the abstract syntax.

Concrete syntax that is not a shorthand form is strict concrete syntax. The relationship from concrete syntax to abstract syntax is defined only for the strict concrete syntax. Strict syntax is always defined in the *Concrete grammar*. The syntax for a shorthand is allowed to be defined in the *Model*.

The relationship between concrete syntax and abstract syntax is omitted if the topic being defined is a shorthand form that is modelled by other constructs of the defined language (see *Model* below).

When the name of a non-terminal ends in the concrete grammar with the word "diagram" and there is a name in the abstract grammar that differs only by ending in the word *definition*, then the two rules represent the same concept. For example, <system diagram> in the concrete grammar and *System-definition* in the abstract grammar correspond.

When the name of a non-terminal ends in the concrete grammar with the word "area" and there is a name in the abstract grammar that differs only by having the word *area* deleted, then the two rules represent the same concept. For example, <state partition area> in the concrete grammar and *State-partition* in the abstract grammar correspond.

c) *Semantics*

Properties are relations between different concepts in the defined language. Properties are used in the rules for a model to be well-formed.

An example (from the specification of SDL-2010 in [b-ITU-T Z.101] clause 9 Agents, *Semantics*) of a property is the complete valid input signal set of an agent. This property is used in the static condition "Signals mentioned in <output body>s of the state machine of an agent type shall be in the