

Superseded by a more recent version



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Z.100

(03/93)

PROGRAMMING LANGUAGES

**CCITT SPECIFICATION AND DESCRIPTION
LANGUAGE (SDL)**

ITU-T Recommendation Z.100

Superseded by a more recent version

(Previously "CCITT Recommendation")

FOREWORD

The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the International Telecommunication Union. 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, established the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

ITU-T Recommendation Z.100 was revised by the ITU-T Study Group X (1988-1993) and was approved by the WTSC (Helsinki, March 1-12, 1993).

NOTES

1 As a consequence of a reform process within the International Telecommunication Union (ITU), the CCITT ceased to exist as of 28 February 1993. In its place, the ITU Telecommunication Standardization Sector (ITU-T) was created as of 1 March 1993. Similarly, in this reform process, the CCIR and the IFRB have been replaced by the Radiocommunication Sector.

In order not to delay publication of this Recommendation, no change has been made in the text to references containing the acronyms “CCITT, CCIR or IFRB” or their associated entities such as Plenary Assembly, Secretariat, etc. Future editions of this Recommendation will contain the proper terminology related to the new ITU structure.

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

© ITU 1995

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or

TABLE OF CONTENTS

	<i>Page</i>
1 Introduction to SDL.....	1
1.1 Introduction.....	1
1.1.1 Objective	1
1.1.2 Application	1
1.1.3 System specification.....	2
1.2 SDL grammars	2
1.3 Basic definitions.....	3
1.3.1 Definition, type and instance.....	3
1.3.2 Environment	5
1.3.3 Errors.....	5
1.4 Presentation style	5
1.4.1 Division of text.....	5
1.4.2 Titled enumeration items	5
1.5 Metalanguages	7
1.5.1 Meta IV	7
1.5.2 BNF.....	9
1.5.3 Metalanguage for graphical grammar.....	10
1.6 Differences to SDL-88	11
2 Basic SDL.....	13
2.1 Introduction	13
2.2 General rules	13
2.2.1 Lexical rules	13
2.2.2 Visibility rules, names and identifiers	16
2.2.3 Informal text.....	20
2.2.4 Drawing rules	20
2.2.5 Partitioning of diagrams	20
2.2.6 Comment	21
2.2.7 Text extension	22
2.2.8 Text symbol.....	22
2.3 Basic data concepts	22
2.3.1 Data type definitions	23
2.3.2 Variable	23
2.3.3 Values and literals	23
2.3.4 Expressions.....	23
2.4 System structure	23
2.4.1 Organisation of SDL specifications.....	23
2.4.1.1 Framework	23
2.4.1.2 Package.....	24
2.4.1.3 Referenced definition	26
2.4.2 System	28
2.4.3 Block	30
2.4.4 Process.....	32
2.4.5 Service.....	37
2.4.6 Procedure.....	39
2.5 Communication	42
2.5.1 Channel.....	42
2.5.2 Signal route	44

2.5.3	Connection	47
2.5.4	Signal.....	48
2.5.5	Signal list definition	49
2.6	Behaviour.....	50
2.6.1	Variables.....	50
2.6.1.1	Variable definition.....	50
2.6.1.2	View definition.....	51
2.6.2	Start	51
2.6.3	State.....	52
2.6.4	Input	53
2.6.5	Save	55
2.6.6	Spontaneous transition	56
2.6.7	Label.....	57
2.6.8	Transition	58
2.6.8.1	Transition body	58
2.6.8.2	Transition terminator.....	59
2.6.8.2.1	Nextstate.....	59
2.6.8.2.2	Join	60
2.6.8.2.3	Stop.....	60
2.6.8.2.4	Return	61
2.7	Action.....	62
2.7.1	Task	62
2.7.2	Create	63
2.7.3	Procedure call.....	64
2.7.4	Output.....	65
2.7.5	Decision.....	68
2.8	Timer	70
2.9	Internal input and output	71
2.10	Examples.....	72
3	Structural Decomposition Concepts in SDL.....	82
3.1	Introduction.....	82
3.2	Partitioning.....	82
3.2.1	General	82
3.2.2	Block partitioning.....	83
3.2.3	Channel partitioning.....	86
3.3	Refinement.....	89
4	Additional Concepts of Basic SDL.....	91
4.1	Introduction	91
4.2	Macro	91
4.2.1	Lexical rules	91
4.2.2	Macro definition.....	91
4.2.3	Macro call.....	95
4.3	Generic system definition.....	97
4.3.1	External synonym.....	97
4.3.2	Simple expression.....	97
4.3.3	Optional definition	98
4.3.4	Optional transition string.....	100
4.4	Asterisk state	102

4.5	Multiple appearance of state	103
4.6	Asterisk input	103
4.7	Asterisk save	103
4.8	Implicit transition	103
4.9	Dash nextstate	104
4.10	Priority Input	104
4.11	Continuous signal.....	105
4.12	Enabling condition	106
4.13	Imported and Exported value.....	109
4.14	Remote procedures.....	112
5	Data in SDL.....	115
5.1	Introduction.....	115
5.1.1	Abstraction in data types	115
5.1.2	Outline of formalisms used to model data.....	115
5.1.3	Terminology	116
5.1.4	Division of text on data	116
5.2	The data kernel language	116
5.2.1	Data type definitions	116
5.2.2	Literals and parameterised operators.....	119
5.2.3	Axioms	121
5.2.4	Conditional equations.....	124
5.3	Passive use of SDL data.....	125
5.3.1	Extended data definition constructs.....	125
5.3.1.1	Special operators	126
5.3.1.2	Character string literals	127
5.3.1.3	Predefined data.....	128
5.3.1.4	Equality and noequality.....	129
5.3.1.5	Boolean axioms	129
5.3.1.6	Conditional terms	130
5.3.1.7	Errors.....	131
5.3.1.8	Ordering	132
5.3.1.9	Syntypes	132
5.3.1.9.1	Range condition.....	134
5.3.1.10	Structure sorts.....	136
5.3.1.11	Inheritance.....	137
5.3.1.12	Generators	139
5.3.1.12.1	Generator definition.....	139
5.3.1.12.2	Generator transformation.....	141
5.3.1.13	Synonyms	142
5.3.1.14	Name class literals.....	143
5.3.1.15	Literal mapping	144
5.3.2	Operator definitions.....	146
5.3.3	Use of data.....	148
5.3.3.1	Expressions.....	148
5.3.3.2	Ground expressions	149
5.3.3.3	Synonym.....	150
5.3.3.4	Indexed primary	151
5.3.3.5	Field primary	151

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.