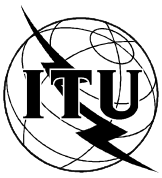


INTERNATIONAL TELECOMMUNICATION UNION

CCITT

THE INTERNATIONAL
TELEGRAPH AND TELEPHONE
CONSULTATIVE COMMITTEE

Z.100 Annex C1

(11/1988)

SERIES Z: LANGUAGES AND GENERAL SOFTWARE ASPECTS FOR TELECOMMUNICATION SYSTEMS

Functional specification and description language (SDL)
Criteria for using formal description techniques (FDTs)

CONCRETE GRAPHICAL SYNTAX SUMMARY

Reedition of CCITT Recommendation Z.100 Annex C1
published in the Blue Book, Fascicle X.1 (1988)

NOTES

1 CCITT Recommendation Z.100 Annex C1 was published in Fascicle X.1 of the *Blue Book*. This file is an extract from the *Blue Book.* While the presentation and layout of the text might be slightly different from the *Blue Book* version, the contents of the file are identical to the *Blue Book* version and copyright conditions remain unchanged (see below).

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

ANNEX C1
(To Recommendation Z.100)

Concrete graphical syntax summary

C1.1 Introduction

C1.1.1 *Metalanguage*

For the graphical grammar the metalanguage described in SDL Rec § 1.5.2 is extended with the following metasymbols:

a) **contains**
b) **is associated with**
c) **is followed by**
d) **is connected to**
e) **set**

The **set** metasymbol is a postfix operator operating on the immediately preceding syntactic elements within curly brackets, and indicating an (unordered) set of items. Such items may be any syntactic element, in which case it must be applied before the **set** metasymbol. Example:

{{<system text area>}* {<macro diagram>}* <block interaction area>} **set**

is a set of zero or more <system text area>s, zero or more <macro diagram>s and one <block interaction area>.

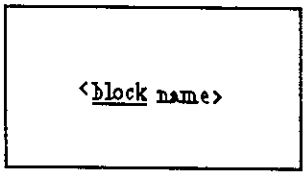
All the other metasymbols are infix operators, having a graphical non-terminal symbol as the left-hand argument. The right-hand argument is either a group of syntactic elements within curly brackets or a single syntactic element. If the right-hand side of a production rule has a graphical non-terminal symbol as the first element and contains one or more of these infix operators, then the graphical non-terminal symbol is the left-hand argument of each of these infix operators. A graphical non-terminal symbol is a non-terminal having the word "symbol" immediately before the greater than sign >.

The metasymbol **contains** indicates that its right-hand argument should be placed within its left-hand argument and the attached <text extension symbol>, if any. Example:

<block reference> ::=
<block symbol> **contains** <block name>

<block symbol> ::=

means the following



The metasymbol **is associated with** indicates that its right-hand side argument is logically associated with its left-hand argument (as if it were "contained" in that argument, the unambiguous association is ensured by appropriate drawing rules).

The metasymbol **is followed by** means that its right-hand argument follows (both logically and in drawing) its left-hand argument.

The metasymbol **is connected to** means that its right-hand argument is connected (both logically and in drawing) to its left-hand argument.

C1.1.2 *General rules*

C1.1.2.1 *Partitioning of diagrams*

The following definition of diagram partitioning is not part of the *Concrete graphical grammar*, but the same meta-language is used.

<page>::=
<frame symbol> **contains**
<heading area> <page number area>
{<syntactical unit>}*

<heading area> ::=
<implicit text symbol> **contains** <heading>

<page number area> ::=
<implicit text symbol> **contains** [<page number> [(<number of pages>)]]

<page number> ::=
<literal name>

<number of pages> ::=
<natural literal name>

<page> is a starting non-terminal, therefore it is not referred to in any production rule. A diagram may be partitioned into a number of <page>s, in which case the <frame symbol> delimiting the diagram and the diagram <heading> are replaced by a <frame symbol> and a <heading> for each page.

The <implicit text symbol> is not shown, but implied, in order to have a clear separation

between <heading area> and <page number area>. <heading area> is placed at the upper left corner of the <frame symbol>. <page number area> is placed at the upper right corner of the <frame symbol>. <heading> and <syntactical unit> depends on the type of diagram.

C1.1.2.2 *Comment*

<comment area> ::=
<comment symbol> **contains** <text>
is connected to <dashed association symbol>

<comment symbol> ::=

<dashed association symbol> ::=

One end of the <dashed association symbol> must be connected to the middle of the vertical segment of the <comment symbol>.

A <comment symbol> is connected to any graphical symbol by means of a <dashed association symbol>. The <comment symbol> is considered as a closed symbol by completing (in imagination) the rectangle. It contains comment text related to the graphical symbol.

C1.1.2.3 *Text extension*

<text extension area> ::=
<text extension symbol> **contains** <text>
is connected to <solid association symbol>

<text extension symbol> ::=
<comment symbol>

<solid association symbol> ::=

———————

A <text extension symbol> is connected to any graphical symbol by means of a <solid association symbol>. The <text extension symbol> is considered as a closed symbol by completing (in imagination) the rectangle.

One end of the <solid association symbol> must be connected to the middle of the vertical segment of the <text extension symbol>.

The text contained in the <text extension symbol> is a continuation of the text within the graphical symbol and is considered to be contained in that symbol.

C1.2 *System definition*

<concrete system definition> ::=
{<system definition> | <system diagram>} {<remote definition>}*

<remote definition> ::=
<definition>
| <diagram>

<diagram> ::=
<block diagram>
| <process diagram>
| <procedure diagram>
| <block substructure diagram>
| <channel substructure diagram>
| <service diagram>
| <macro diagram>

C1.3 *System diagram*

<system diagram> ::=
<frame symbol> **contains**
{<system heading>
{ {<system text area>}*
{<macro diagram>}*
<block interaction area> }**set** }

<frame symbol> ::=

<system heading> ::=
SYSTEM <system name>

<system text area> ::=
<text symbol> **contains**
{<signal definition>
| <signal list definition>
| <data definition>
| <macro definition>
| <select definition>}*

<text symbol> ::=

<block interaction area> ::=
{<block area> | <channel definition area>}+

<block area> ::=
<graphical block reference>
| <block diagram>

<graphical block reference> ::=
<block symbol> **contains** <block name>

<block symbol> ::=

<channel definition area> ::=
<channel symbol>
is associated with {<channel name>
{ [{<channel identifier> | <block identifier>}]
<signal list area>
[<signal list area>] }**set** }
is connected to { <block area>
{<block area> | <frame symbol>}
[<channel substructure association area>] }set

The <channel identifier> identifies an external channel connected to the <block substructure diagram> delimited by the <frame symbol>. The <block identifier> identifies an external block being a channel endpoint for the <channel substructure diagram> delimited by the <frame symbol>.

<channel symbol> ::=
<channel symbol 1>
| <channel symbol 2>
| <channel symbol 3>

<channel symbol 1> ::=

<channel symbol 2> ::=

<channel symbol 3> ::=

<signal list area> ::=
<signal list symbol> **contains** <signal list>

<signal list symbol> ::=

C1.4 *Block diagram*

<block diagram> ::=
<frame symbol>
contains {<block heading>
{ {<block text area>}*
{<macro diagram>}*
[<process interaction area>]
[<block substructure area>] }**set** }
is associated with {<channel identifier>}*

The <channel identifier> identifies a channel connected to a signal route in the <block diagram>. It is placed outside the <frame symbol> close to the endpoint of the signal route at the <frame symbol>. If the <block diagram> does not contain a <process interaction area>, then it must contain a <block substructure area>.

<block heading> ::=
BLOCK {<block name> | <block identifier> }

<block text area> ::=
<system text area>

<process interaction area> ::=
{ <process area>
| <create line area>
| <signal route definition area> }+

<process area> ::=
<graphical process reference>
| <process diagram>

<graphical process reference> ::=
<process symbol> **contains** {<process name> [<number of instances>]}

<process symbol> ::=

<create line area> ::=
<create line symbol>
is connected to {<process area> <process area>}

<create line symbol> ::=

<signal route definition area> ::=
<signal route symbol>
is associated with {<signal route name>
{ [<channel identifier>]
<signal list area>
[<signal list area>] }**set** }
is connected to
{<process area> {<process area> | <frame symbol>} }**set**

When the <signal route symbol> is connected to the <frame symbol>, then the <channel identifier> identifies a channel to which the signal route is connected.

<signal route symbol> ::=
<signal route symbol 1>
| <signal route symbol 2>

<signal route symbol 1> ::=

<signal route symbol 2> ::=

C1.5 *Process diagram*

<process diagram> ::=
<frame symbol>
contains {<process heading>
{ {<process text area>}*
{<procedure area>}*
{<macro diagram> }*
{<process graph area> | <service interaction area> } }**set** }
[**is associated with** {<signal route identifier>}+]

The <signal route identifier> identifies an external signal route connected to a signal route in the <process diagram>. It is placed outside the <frame symbol> close to the endpoint of the signal route at the <frame symbol>.

<process heading> ::=
PROCESS {<process name> | <process identifier>}
[<number of instances> <end>] [<formal parameters>]

<process text area> ::=
<text symbol> **contains** [<valid input signal set>]
{<signal definition>
| <signal list definition>
| <variable definition>
| <view definition>
| <import definition>
| <data definition>
| <macro definition>
| <timer definition>
| <select definition> }*

<graphical procedure reference> ::=
<procedure symbol> **contains** <procedure name>

<procedure symbol> ::=

<process graph area> ::=
<start area> { <state area> |<in-connector area> }*

<start area> ::=
<start symbol> **is followed by** <transition area>

<start symbol> ::=

<transition area> ::=
[<transition string area>] **is followed by**
{<state area>
| <nextstate area>
| <decision area>
| <stop symbol>
| <merge area>
| <out-connector area>
| <return symbol>
| <transition option area> }

<merge area> ::=
<merge symbol> **is connected to** <flow line symbol>

<merge symbol> ::=
<flow line symbol>

<flow line symbol> ::=

<transition string area> ::=
{<task area>
| <output area>
| <priority output area>
| <set area>
| <reset area>
| <export area>
| <create request area>
| <procedure call area> }
[**is followed by** <transition string area>]

<task area> ::=
<task symbol> **contains** <task body>

<task symbol> ::=

<output area> ::=
<output symbol> **contains** <output body>

<output symbol> ::=

<create request area> ::=
<create request symbol> **contains** <create body>

<create request symbol> ::=

<procedure call area> ::=
<procedure call symbol> **contains** <procedure call body>

<procedure call symbol> ::=

<state area> ::=
<state symbol> **contains** <state list> **is associated with**
{<input association area>
| <priority input association area>
| <continuous signal association area>
| <save association area> }*

<state symbol> ::=

<input association area> ::=
<solid association symbol> **is connected to** <input area>

<input area> ::=
<input symbol> **contains** <input list>
is followed by { [<enabling condition area>] <transition area>}

<input symbol> ::=

<save association area> ::=
<solid association symbol> **is connected to** <save area>

<save area> ::=
<save symbol> **contains** <save list>

<save symbol> ::=

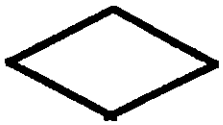
<in-connector area> ::=
<in-connector symbol> **contains** <connector name>
is followed by <transition area>

<in-connector symbol> ::=

<nextstate area> ::=
<state symbol> **contains** <nextstate body>

<decision area> ::=
<decision symbol> **contains** <question>
is followed by
{ {<graphical answer part> <graphical else part>} **set**
| {<graphical answer part> {<graphical answer part>}+ [<graphical else part>] } set }

<decision symbol> ::=



<graphical answer> ::=
<answer> | (<answer>)

<graphical answer part> ::=
<flow line symbol> **is associated with** <graphical answer>
is followed by <transition area>

<graphical else part> ::=
<flow line symbol> **is associated with** ELSE
is followed by <transition area>

<set area>::=
<task symbol> **contains** <set>

<reset area>
<task symbol> **contains** <reset>

<stop symbol> ::=

<out-connector area> ::=
<out-connector symbol> **contains** <connector name>

<out-connector symbol> ::=
<in-connector symbol>

C1.6 *Procedure diagram*

<procedure diagram> ::=
<frame symbol> **contains** {<procedure heading>
{ {<procedure text area>}*
{ <procedure area>}*
{<macro diagram> }*
<procedure graph area> }**set** }

<procedure heading> ::=
PROCEDURE {<procedure name> | <procedure identifier> }
[<procedure formal parameters>]

<procedure area> ::=
<graphical procedure reference>
| <procedure diagram>

<procedure text area> ::=
<text symbol> **contains**
{<variable definition>
| <data definition>
| <macro definition>
| <select definition> }*

<procedure graph area> ::=
<procedure start area> {<state area> | <in-connector area> }*

<procedure start area> ::=
<procedure start symbol> **is followed by** <transition area>

<procedure start symbol> ::=

<return symbol> ::=

C1.7 *Block substructure*

<block substructure area> ::=
<graphical block substructure reference>
| <block substructure diagram>

<graphical block substructure reference> ::=
<block substructure symbol> **contains** <block substructure name>

<block substructure symbol> ::=
<block symbol>

<block substructure diagram> ::=
<frame symbol>
contains {<block substructure heading>
{ {<block substructure text area>}*
{<macro diagram>}*
<block interaction area> }**set** }
is associated with {<channel identifier>}*

The <channel identifier> identifies a channel connected to a subchannel in the <block substructure diagram>. It is placed outside the <frame symbol> close to the endpoint of the subchannel at the <frame symbol>.

<block substructure heading> ::=
SUBSTRUCTURE {<block substructure name> | <block substructure identifier>}

<block substructure text area> ::=
<system text area>

C1.8 *Channel substructure*

<channel substructure association area> ::=
<dashed association symbol>
is connected to <channel substructure area>

<channel substructure area> ::=
<graphical channel substructure reference>
| <channel substructure diagram>

<graphical channel substructure reference> ::=
<channel substructure symbol> **contains** <channel substructure name>

<channel substructure symbol> ::=
<block symbol>

<channel substructure diagram> ::=
<frame symbol>
contains {<channel substructure heading>
{ {<channel substructure text area>}*
{<macro diagram>}*
<block interaction area> }**set** }
is associated with {<block identifier> | ENV}+

The <block identifier> or ENV identifies an endpoint of the partitioned channel. The <block identifier> is placed outside the <frame symbol> close to the endpoint of the associated subchannel at the <frame symbol>.

<channel substructure heading> ::=
SUBSTRUCTURE { < channel substructure name>
| < channel substructure identifier>}

<channel substructure text area> ::=
<system text area>

C1.9 *Macro*

9.1 *Macro diagram*

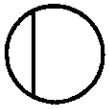
<macro diagram> ::=
<frame symbol> **contains** <macro heading> <macro body area>

<macro heading> ::=
MACRODEFINITION <macro name> [<macro formal parameters>]

<macro body area> ::=
{ {<any area> }*
<any area> [**is connected to** <macro body port1>] }**set**
|{ <any area> **is connected to** <macro body port2>
<any area> **is connected to** <macro body port2>
{ <any area> [**is connected to** <macro body port2>]}*}**set**

<macro inlet symbol>::=

<macro outlet symbol>::=



<macro body port1> ::=
<outlet symbol> [**is associated with** <macro label>]
is connected to {<frame symbol>
| <macro inlet symbol>
| <macro outlet symbol>}

<macro body port2> ::=
<outlet symbol> **is associated with** <macro label>
is connected to {<frame symbol>
| <macro inlet symbol>
| <macro outlet symbol>}

<macro label> ::=
<name>

<outlet symbol> ::=
<dummy outlet symbol>
| <flow line symbol>
| <channel symbol>
| <signal route symbol>
| <solid association symbol>
| <dashed association symbol>
| <create line symbol>

<dummy outlet symbol> ::=
<solid association symbol>

<any area> ::=
<system text area>
| <block interaction area>
| <signal list area>
| <block area>
| <block text area>
| <process interaction area>
| <graphical procedure reference>
| <procedure area>
| <process text area>
| <process graph area>
| <merge area>
| <transition string area>
| <state area>
| <input area>
| <save area>
| <text extension area>
| <channel substructure association area>
| <channel substructure area>
| <block substructure area>
| <priority input area>
| <continuous signal area>
| <in-connector area>
| <nextstate area>
| <process area>
| <channel definition area>
| <create line area>
| <signal route definition area>
| <graphical process reference>
| <process diagram>
| <start area>
| <output area>
| <set area>
| <reset area>
| <export area>
| <priority output area>
| <task area>
| <create request area>
| <procedure call area>
| <decision area>

| <out-connector area>
| <procedure text area>
| <procedure graph area>
| <procedure start area>
| <block substructure text area>
| <block interaction area>
| <service area>
| <service signal route definition area>
| <service text area>
| <service graph area>
| <service start area>
| <comment area>
| <macro call area>

C1.9.2 *Macro call*

<macro call area> ::=
<macro call symbol> **contains** {<macro name> [<macro call body>]}
[is connected to
{<macro call port1> | <macro call port2> {<macro call port2>}+}]

<macro call symbol> ::=

<macro call port1> ::=
<inlet symbol> **[is associated with** <macro label>]
is connected to <any area>

<macro call port2> ::=
<inlet symbol> **is associated with** <macro label>
is connected to <any area>

<inlet symbol> ::=
<dummy inlet symbol>
| <flow line symbol>
| <channel symbol>
| <signal route symbol>
| <solid association symbol>
| <dashed association symbol>
| <create line symbol>

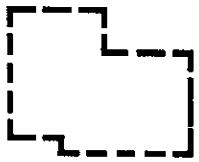
<dummy inlet symbol> ::=
<solid association symbol>

C1.10 *Generic systems*

C1.10.1 *Optional definition*

<option area> ::=
<option symbol> **contains**
{ SELECT IF (<boolean simple expression>)
{<block area>
| <channel substructure area>
| <system text area>
| <block text area>
| <process text area>
| <procedure text area>
| <block substructure text area>
| <channel substructure text area>
| <service text area>
| <macro diagram>
| <option area>
| <process area>
| <signal route definition area>
| <create line area>
| <procedure area>
| <service area>
| <service signal route definition area> }+ }

The <option symbol> is a dashed polygon having solid corners, for example:



C1.10.1 *Optional transition*

<transition option area> ::=
<transition option symbol> **contains** {<alternative question>}
is followed by {<option outlet1> {<option outlet1> | <option outlet2> }
{ <option outlet1> }* }**set**

<transition option symbol> ::=

<option outlet1> ::=
<flow line symbol> **is associated with** <graphical answer>
is followed by {<transition area> | <merge area>}

<option outlet2> ::=
<flow line symbol> **is associated with** ELSE
is followed by {<transition area> | <merge area>}

C1.11 *Service*

C1.11.1 *Service decomposition*

<service interaction area> ::=
{ <service area> | <service signal route definition area> }+

<service area> ::=
<graphical service reference>
| <service diagram>

<graphical service reference> ::=
<service symbol> **contains** <service name>

<service symbol> ::=

<service signal route definition area> ::=
<signal route symbol>
is associated with {<service signal route name> [<signal route identifier>]
<signal list area> [<signal list area>]}**set**
is connected to {<service area>
{<service area> | <frame symbol>} }**set**

When the <signal route symbol> is connected to the <frame symbol>, then the <signal route identifier> identifies an external signal route to which the signal route is connected.

C1.11.2 *Service diagram*

<service diagram> ::=
<frame symbol> **contains**
{<service heading>
{ {<service text area>}*
{<procedure area>}*
{<macro diagram>}*
<service graph area> }**set** }

<service heading> ::=
SERVICE { <service name> | <service identifier>}

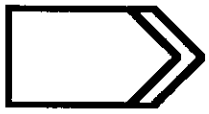
<service text area> ::=
<text symbol> **contains**
{ <variable definition>
| <view definition>
| <import definition>
| <data definition>
| <macro definition>
| <timer definition>
| <select definition> }*

<service graph area> :::=
<process graph area>

<priority input association area> ::=
<solid association symbol> **is connected to** <priority input area>

<priority input area> ::=
<priority input symbol> **contains** <priority input list>

<priority input symbol> ::=



<priority output area> ::=
<priority output symbol> **contains** <priority output body>

<priority output symbol> ::=

C1.12 *Continuous signal*

<continuous signal association area> ::=
<solid association symbol> **is connected to** <continuous signal area>

<continuous signal area> ::=
<enabling condition symbol>
contains {<boolean expression> [<end> PRIORITY <integer literal name>]}
is followed by <transition area>

C1.13 *Enabling condition*

<enabling condition area> ::=
<enbling condition symbol> **contains** <boolean expression>

<enabling condition symbol> ::=

C1.14 *Export*

<export area> ::=
<task symbol> **contains** <export>

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
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 and Internet protocol aspects
Series Z	**Languages and general software aspects for telecommunication systems**

Printed in Switzerland

Geneva, 2010