国 际 电 信 联 盟

ITU-T Z.141

国际电信联盟
电信标准化部门

(03/2006)

Z系列：电信系统使用的语言和一般性软件情况

正式描述技巧 (FDT) — 测试和测试控制记法(TTCN)

测试和测试控制记法版本3 (TTCN-3): 列表描述格式(TFT)

ITU-T Z.141建议书

国际电信联盟

ITU-T Z系列建议书

电信系统使用的语言和一般性软件情况

正式描述技巧 (FDT)	
规范和描述语言 (SDL)	Z.100-Z.109
正式描述技巧的应用	Z.110-Z.119
信息排序表 (MSC)	Z.120-Z.129
扩展的目标描述语言 (eODL)	Z.130-Z.139
测试和测试控制记法 (TTCN)	**Z.140-Z.149**
用户要求记法 (URN)	Z.150-Z.159
编程语言	
CHILL: ITU-T 高级语言	Z.200-Z.209
人机语言	
总则	Z.300-Z.309
基本句法和对话程序	Z.310-Z.319
用于视频显示终端的扩展 MML	Z.320-Z.329
人机接口规范	Z.330-Z.349
面向数据的人机接口	Z.350-Z.359
电信网络管理使用的人机接口	Z.360-Z.379
质量	
电信软件的质量	Z.400-Z.409
涉及协议的建议书中有关质量的内容	Z.450-Z.459
方法	
认证与测试的方法	Z.500-Z.519
中间件	
分布式处理环境	Z.600-Z.609

欲了解更详细信息，请查阅*ITU-T*建议书目录。

ITU-T Z.141建议书

测试和测试控制记法版本3(TTCN-31)：
列表描述格式(TFT)

摘　要

本建议书规定 TFT，TTCN-3 的列表格式。 TFT 为 ITU-T Z.140 建议书中定义的 TTCN-3 (*测试和测试控制记法 3*) 中心语言的列表描述格式。它类似于 ITU-T X.292 建议书中规定的用于一致性测试的 TTCN-2 的外观和功能。列表格式提供了另一种方法，以显示中心语言并强调特定于标准化的一致性测试组要求的问题。虽然中心语言的使用可以独立于列表描述格式，但列表格式的使用不能缺少中心语言。列表描述格式的使用和实现必须基于中心语言。本建议书规定标准形式、语法映射、附加静态语义、操作语义限制、显示和其他属性。这些特性共同构成列表描述格式。

TFT 继承了中心语言的所有的基本特性并旨在规范与平台、测试方法、协议层和协议无关的测试组。TTCN-3 可以用于各种通信端口上所有类型的反应系统测试规范。典型的应用范围为协议测试 (包括移动和网际协议)、服务测试 (包括补充业务)、模块测试、基于 CORBA 的平台和 APIs 测试。用于物理层协议的测试组规范超出本建议书的范围。

来　源

ITU-T 第 17 研究组（2005-2008）按照 ITU-T A.8 建议书规定的程序，于 2006 年 3 月 16 日批准了 ITU-T Z.141 建议书。

前　言

国际电信联盟（ITU）是从事电信领域工作的联合国专门机构。ITU-T（国际电信联盟电信标准化部门）是国际电信联盟的常设机构，负责研究技术、操作和资费问题，并且为在世界范围内实现电信标准化，发表有关上述研究项目的建议书。

每四年一届的世界电信标准化全会（WTSA）确定 ITU-T 各研究组的研究课题，再由各研究组制定有关这些课题的建议书。

WTSA 第 1 号决议规定了批准建议书须遵循的程序。

属 ITU-T 研究范围的某些信息技术领域的必要标准，是与国际标准化组织（ISO）和国际电工技术委员会（IEC）合作制定的。

注

本建议书为简明扼要起见而使用的“主管部门”一词，既指电信主管部门，又指经认可的运营机构。

遵守本建议书的规定是以自愿为基础的，但建议书可能包含某些强制性条款（以确保例如互操作性或适用性等），只有满足所有强制性条款的规定，才能达到遵守建议书的目的。“应该”或“必须”等其它一些强制性用语及其否定形式被用于表达特定要求。使用此类用语不表示要求任何一方遵守本建议书。

知识产权

国际电联提请注意：本建议书的应用或实施可能涉及使用已申报的知识产权。国际电联对无论是其成员还是建议书制定程序之外的其它机构提出的有关已申报的知识产权的证据、有效性或适用性不表示意见。

至本建议书批准之日止，国际电联尚未收到实施本建议书可能需要的受专利保护的知识产权的通知。但需要提醒实施者注意的是，这可能并非最新信息，因此特大力提倡他们通过下列网址查询电信标准化局（TSB）的专利数据库：http://www.itu.int/ITU-T/ipr/。

© 国际电联 2006

版权所有。未经国际电联事先书面许可，不得以任何手段复制本出版物的任何部分。

目 录

ITU-T Z.141建议书

测试和测试控制记法版本3（TTCN-3）：
列表描述格式（TFT）

1 范围

本建议书规定 TTCN 版本 3(或 TTCN-3)的列表描述格式。本建议书以 ITU-T Z.140 建议书[1]中定义的 TTCN-3 中心语言为基础。

其他格式的规范超出了本建议书的范围。

2 参考文献

下列 ITU-T 建议书和其它参考文献的条款，在本建议书中的引用而构成本建议书的条款。在出版时，所指出的版本是有效的。所有的建议书和其它参考文献均会得到修订，本建议书的使用者应查证是否有可能使用下列建议书或其它参考文献的最新版本。当前有效的 ITU-T 建议书清单定期出版。本建议书引用的文件自成一体时不具备建议书的地位。

[1] ITU-T Recommendation Z.140 (2006), *Testing and Test Control Notation version 3 (TTCN-3): Core language.*

[2] ITU-T Recommendation Z.143 (2006), *Testing and Test Control Notation version 3 (TTCN-3): Operational semantics.*

3 缩写词

就本建议书而言，下列定义适用：

ASN.1	抽象语法记法一
ATS	抽象测试组
BNF	Backus-Naur 形式
MTC	主测试成分
PICS	协议实现一致性声明
PIXIT	测试的协议实现额外信息
TFT	TTCN-3 的列表描述格式
TTCN	测试和测试控制记法

4 介绍

TTCN-3(TFT)的列表描述格式为图形格式，在功能和外观上类似于早期的 TTCN 版本（面向一致性测试）。TTCN-3 的中心语言的定义见 ITU-T Z.140 建议书[1]，并提供完全基于文本的语法，静态语义并规定使用 ASN.1 语言。操作语义的定义见 ITU-T Z.143 建议书[2]。列表格式提供了另一种方法，以显示中心语言并强调特定于标准化的一致性测试组要求的问题。

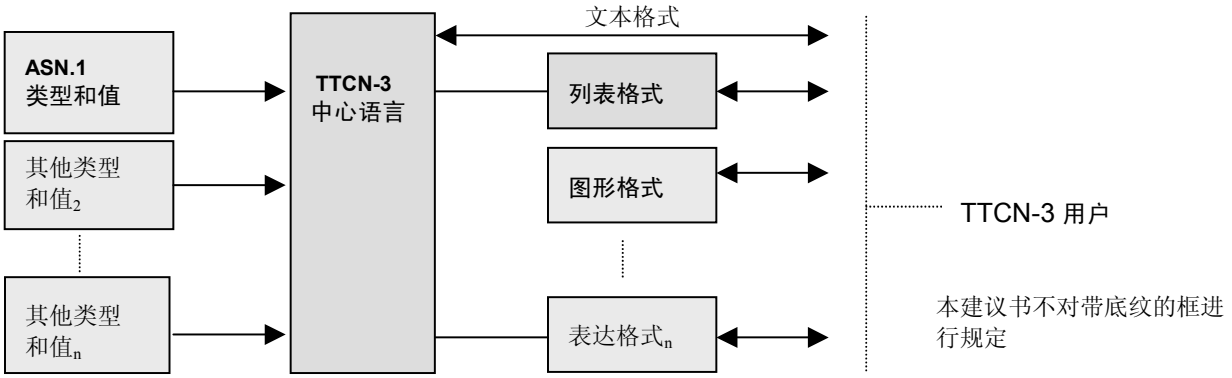

图 1/Z.141－中心语言和各种描述格式的用户角度

中心语言的使用可独立于列表描述格式。但列表格式的使用不能缺少中心语言。列表描述格式的使用和实现必须基于中心语言。

本建议书规定:

a) 标准形式;
b) 语法映射;
c) 附加静态语义;
d) 操作语义限制;
e) 显示和其他属性。

这些特性共同构成列表描述格式。

5 惯例

本节规定在定义 TTCN 标准形式和 TTCN 中心语言文法时已经采用的惯例。

5.1 语法记法

表 1 中规定的记法通常用于说明 TTCN 的扩展的 BNF 文法（此后称为 BNF）。

表 1/Z.141－TTCN.MP 语法记法

::=	将规定
abc xyz	abc 后跟 xyz
|	或
[abc]	abc 的 0 个或 1 个实例
{abc}	abc 的 0 个或多个实例
{abc}+	abc 的 1 个或多个实例
(...)	文本群
abc	非终端符号 abc
abc	一个终端符号 abc
"abc"	一个终端符号 abc

BNF 产品的规定见附件 A/Z.140[1]。

5.2 规范文本

a) **黑体文本**用于表示标准形式字段。
b) 斜体文本用于表示 TTCN-3 中心语言 BNF 产品。
c) **黑体 `courier new`** 文本用于中心语言关键词。

5.3 标准形式

a) **黑体文本**将逐字出现在每个 TTCN-3 模块的实际表中。

b) *斜体文本*不会逐字出现在 TTCN-3 模块中。此字体表示：实际文本应替代斜体的标记。实际文本的标记要求可按照标准形式的定义或参见 TTCN-3 中心语言 BNF。在*斜体文本*之前或之后的方括号表示将文本包括在标准形式的给字段域中文本之前或之后，方框是任选的。

5.4 中心语言

a) 引号中的**黑体**字符（例如,'{'）用于保留的中心语言关键词和终止。

b) *斜体文本*不会逐字出现在 TTCN-3 模块中。此字体表示：实际文本应替代斜体的标记。实际文本的标记要求可按照标准形式的定义或参见 TTCN-3 中心语言 BNF。

c) "..."符号为所有未明确表示出的内容保留位置。

5.5 普通映射规则

列表描述格式和 TTCN-3 中心语言之间的映射由一组转换组成。对于每个标准形式中的每个句法单元有一个相关的转换。转换也可以将任意中心语言模块转成列表描述。

这些转换可归为两类。第一类在列表单元和具有相同意义的中心语言构建之间直接转换。第二类在列表单元和相关中心语言构建（中心语言级上无意义）之间转换。

第一类转换的典型示例应为一个标识符字段。本字段可以直接从列表转换到中心语言且保持其意义，即，确定相同的语言单元。

第二类转换典型地为某注释格式或指令，关于语言单元将如何在描述格式中显示。这些单元在中心语言中无直接意义并采用 *WithStatement* 表示。

本建议书中规定的语法和语义特定于 ETSI 列表描述格式。为明确地识别中心语言中所采用的描述格式，下列特别显示陈述应规定为相关于 TTCN-3 中心语言模块的第一个显示陈述：

```
1: module TTCN3ModuleId    "{"
2:     ...
3: "}" with "{"
4:     display """ "presentation format" ":=" "ETSI Tabular version"
5:     MajorVersion "." MinorVersion """ ";"
6:     ...
7: "}"
```

注 — 相关于一个给定的标准形式的所有*WithStatements* 应在一个相邻的列表中组合起来。

标准形式的**群**字段不得转换为 *WithStatements*，但来源于模块规范的实际群结构。

6 标准形式

6.1 测试组控制

Test Suite Control	
Module Name	*TTCN3ModuleId*
Version	*[TabFreeText]*
Date	*[TabFreeText]*
Base Standard Ref	*[TabFreeText]*
Test Standard Ref	*[TabFreeText]*
PICS Ref	*[TabFreeText]*
PIXIT Ref	*[TabFreeText]*
Test Method(s)	*[TabFreeText]*
Encoding	*[TabFreeText]*
Comments	*[TabFreeText]*

Local Def Name	**Type**	**Initial Value**	**Comments**
[VarConstOrTimerIdentifier]	*[ConstTypeOrTimer]*	*[Expression]*	*[TabFreeText]*
...	...	...	...

Behaviour	
ModuleControlBody	
Detailed Comments	*[TabFreeText]*

图 2/Z.141－测试组控制标准形式

6.1.1 映射

测试组控制标准形式用三个部分表述。第一部分包括头字段和**详尽注释**字段，它改为相关于整个 TTCN-3 模块的 *WithStatement* 中的显示属性。**模块名**字段映射为模块标识符。

第二部分由本地恒量、变量和定时器组成（在控制部分中规定）。这些定义可以在中心语言的控制部分出现，但对标准形式，它们与模块控制体的其他部分分开并在单独的表格中显示。须保持定义的顺序，因为这些定义可互相依赖。对所有定时器，**类型**栏应置为关键词**定时器**，对所有恒量，应置为关键词**恒量**之后的恒量类型。本地定义表的**注释**改为相关于 TTCN-3 中心语言模块的控制部分的 *WithStatement* 中的显示属性。

第三部分为 TTCN-3 中心语言模块的控制部分减去本地恒量、变量和定时器。

```
 1: module TTCN3ModuleId "{"
 2:     control "{"
 3:  var Type VarIdentifier [":=" Expression] ";"
 4:  timer TimerIdentifier [":=" Expression] ";"
 5:  const Type ConstIdentifier ":=" ConstantExpression;
 6:         ModuleControlBody
 7:     "}" with "{"
 8:         { VarConstOrTimerCommentsAttribute }
 9:     "}"
10: "}" with "{"
11:     ModuleAttributes
12:  [EncodeAttribute;]
13: "}"
```

EXAMPLE:

Test Suite Control			
Module Name	Example1		
Version	1.01		
Date	19 July 2001		
Base Standards Ref	ITU-T Recommendation Q.123		
Test Standards Ref	ITU-T Recommendation Q.123.1		
PICS Ref	ITU-T Recommendation Q.123.2, Annex A		
PIXIT Ref	ITU-T Recommendation Q.123.2, Annex B		
Test Method(s)	local		
Encoding	BER		
Comments	ATS written by STF 133		
Local Def Name	**Type**	**Initial Value**	**Comments**
PI	const float	3.14	the ratio
x	float	PI * 2	double PI
t1	timer	15	a 15 second timer
Behaviour			

/* group1/ */
/* group1_1/ */
execute(test1);
execute(test2);
/* group1_2/ */
execute(test3);
execute(test4);
/* group2/ */
execute(test5);			

Detailed Comments	detailed comments		

映射为:

```
  1: module Example1 {
  2:     control {
  3:         const float PI := 3.14;
  4:         var float x := PI * 2;
  5:         timer t1 := 15;
  6:
  7:         /* group1/ */
  8:             /* group1_1/ */
  9:                 execute(test1());
 10:                 execute(test2());
 11:             /* group1 2/ */
 12:                 execute(test3());
 13:                 execute(test4());
 14:             /* group2/ */
 15:                 execute(test5());
 16:     } with {
 17:         display (PI) "comments := the ratio";
 18:         display (x) "comments := double PI";
 19:         display (t1) "comments := a 15 second timer";
 20:     }
 21: } with {
 22:     display "presentation format := ETSI Tabular version 1.0";
 23:     display "module version := 1.01";
 24:     display "module date := 19 July 2001";
 25:     display "module base standards ref := ITU-T Recommendation Q.123";
 26:     display "module test standards ref := ITU-T Recommendation Q.123";
 27:     display "module pics ref := ITU-T Recommendation Q.123, Annex A";
 28:     display "module pixit ref := ITU-T Recommendation Q.123, Annex A";
 29:     display "module test method := local";
 30:     display "module comments := ATS written by STF 133";
 31:     display "module detailed comments := detailed comments";
 32:     encode "BER";
 33: }
```

6.2 测试组参数

Test Suite Parameters				
Name	**Type**	**Initial Value**	**PICS/PIXIT Ref**	**Comments**
. *ModuleParIdentifier* .	. *ModuleParType* .	. *[ConstantExpression]* .	. *[TabFreeText]* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*			

图 3/Z.141－测试组参数标准形式

6.2.1 映射

测试组参数标准形式中的所有条目映射为相关于 TTCN-3 模块的 *ModuleParameterDefs* 中的 *ModuleParLists*。如果有多个 *ModuleParameterDef*，则收集所有 *ModuleParLists* 并在一个**测试组参数标准**形式中显示。

PICS/PIXITref 和**注释**映射为相关于装入 *ParamDef* 的 *WithStatements* 中的参数标识符限定的显示属性。**详尽注释**字段映射为相关于装入 *ParamDef* 的 *WithStatement* 中的显示属性。

```
 1: module TTCN3ModuleId "{"
 2: parameters "{" ModuleParList "}"
 3:  with "{"
 4:  [ModuleParPicsPixitRefAttribute ";"]
 5:  [ModuleParComments ";"]
 6:  [DetailedComments ";"]
 7: "}"
 8: "}"
```

EXAMPLE:

Test Suite Parameters				
Name	**Type**	**Initial Value**	**PICS/PIXIT Ref**	**Comments**
CAP_1	boolean	true	A.1.3	option 1 implemented
Tall	float	600.0	A.1.4	overall module timer
Detailed Comments	detailed comments			

映射为:

```
 1: module MyModule{
 2:  parameters { boolean CAP 1 := true, float Tall := 600.0 }
 3:  with {
 4:  display (CAP_1) "pics/pixit ref := A.1.3";
 5:  display (CAP_1) "comments := option 1 implemented";
 6:  display (Tall) "pics/pixit ref := A.1.4";
 7:  display (Tall) "comments := overall module timer";
 8:  display "detailed comments := detailed comments"
 9:  }
10: }
```

6.3 模块输入

Imports	
Source Name	*GlobalModuleId [**recursive**]*
Source Language	*[LanguageSpec]*
Group	*[GroupReference]*
Source Ref	*[TabFreeText]*
Encoding	*[TabFreeText]*
Comments	*[TabFreeText]*

Type	Name
. *[ImportType]* .	. *ImportSpecification* .
Detailed Comments	*[TabFreeText]*

图 4/Z.141－输入标准形式

6.3.1 映射

输入标准形式映射为 TTCN-3 中心语言中的 *ImportDef* 陈述。**源名称**、**源语言**、**类型**和**名称**字段直接用于相应的中心语言 *ImportDef* 陈述中。**源参考**、**注释**和**详尽注释**字段表述为相关于 *ImportDef* 陈述的 *WithStatement* 中的显示属性。**编码**字段表述为相关于 *ImportDef* 陈述的 *WithStatement* 中的编码属性。

如果一个模块的所有定义为输入，则 *ImportType* 将是空的且 *ImportSpecification* 将采用关键词 **all**。

```
 1: module TTCN3ModuleId "{"
 2:  ImportDef
 3:     with "{"
 4:  [ImportsSourceRefAttribute ";"]
 5:  [CommentsAttribute ";"]
 6:  [ImportsSourceDefinitionCommentsAttribute ";"]
 7:  [DetailedCommentsAttribute ";"]
 8:  [EncodeAttribute ";"]
 9:  "}"
10: "}"
```

EXAMPLE:

Imports		
Source Name	ModuleA recursive	
Source Language	ASN.1:1997	
Group		
Source Ref	EN 800 900 version 2	
Encoding	BER	
Comments	importing declarations from ATS	
Type	**Name**	**Comments**
constant	all except foobar	
Type	MyType	foobar
Group	AtoU_CTR	
Detailed Comments	detailed comments	

映射为:

```
 1: module MyModule {
 2:  import from ModuleA recursive language "ASN.1997" {
 3:  const all except foobar;
 4:  type MyType;
 5:  Group AtoU_CTR;
 6:  } with {
 7:  display "imports source ref := EN 800 900 version 2";
 8:  display "comments := importing declarations from ATS";
 9:  display "detailed comments := detailed comments";
10:  encode "BER";
11:  }
12: }
```

6.4 简单类型

Simple Types			
Group	*[GroupReference]*		
Name	**Definition**	**Encoding**	**Comments**
. *SubTypeIdentifier* .	. *Type [ArrayDef] [SubTypeSpec]* .	. *[TabFreeText]* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*		

图 5/Z.141－简单类型标准形式

6.4.1 映射

简单类型标准形式映射为相同群级上简单类型定义陈述系列。简单类型定义为全部 *SubTypeDef* 类型定义。

详尽注释字段映射为相关于装入群或模块的 *WithStatement* 内的显示属性。在相关于各自简单类型定义的 *WithStatement* 中，**编码**和**注释**字段分别映射为编码和显示属性。

```
 1: module TTCN3ModuleId "{"
 2:  type Type SubTypeIdentifier [ArrayDef] [SubTypeSpec] with "{"
 3:  [EncodeAttribute ";"]
 4:  [CommentsAttribute ";"]
 5: "}" with "{"
 6:  [SimpleTypesDetailedCommentsAttribute ";"]
 7: "}"
```

EXAMPLE:

Simple Types			
Group	SimpleTypes/		
Name	**Definition**	**Encoding**	**Comments**
EQ_NUMBER	integer (1 .. 20)	PER	God knows
Detailed Comments	detailed comments		

映射为:

```
 1: module MyModule {
 2:  group SimpleTypes {
 3:  type integer EQ NUMBER (1..20) with {
 4:  encode "PER";
 5:  display "comments := God knows";
 6:  }
 7:  } with {
 8:  display "simple types detailed comments := detailed comments";
 9:  }
10: }
```

6.5 构建类型

Structured Type			
Name	*StructTypeIdentifier[StructDefFormalParList]*		
Group	*[GroupReference]*		
Structure	*StructureType*		
Encoding	*[TabFreeText]*		
Comments	*[TabFreeText]*		
Field Name	**Field Type**	**Field Encoding**	**Comments**
.	.	.	.
FieldIdentifier	*Type [ArrayDef]*	*[TabFreeText]*	*[TabFreeText]*
.	*[SubTypeSpec]*	.	.
.	*[OptionalKeyword]*	.	.
.	.	.	.
Detailed Comments	*[TabFreeText]*		

图 6/Z.141－构建类型标准形式

6.5.1 映射

构建类型标准形式映射为 TTCN-3 中的构建类型定义陈述。下列类型将采用此标准形式:*RecordDef, UnionDef* 和 *SetDef*。

注释和**详尽注释**字段映射为对应 *WithStatement* 中的显示属性，**编码**字段映射为对应 *WithStatement* 中的编码属性。每个字段单元的**注释**和**字段编码**字段分别映射为显示和编码属性，由对应 *WithStatement* 中的 *FieldIdentifier* 限定。

```
 1: module TTCN3ModuleId "{"
 2:  type StructureType StructTypeIdentifier [StructDefFormalParList] "{"
 3:  {Type FieldIdentifier [ArrayDef] [SubtypeSpec] [OptionalKeyword]}
 4:  "}" with "{"
 5:  [EncodeAttribute ";"]
 6:  [CommentsAttribute ";"]
 7:  {FieldCommentsAttribute ";"}
 8:  {FieldEncodeAttribute ";"}
 9:  [DetailedCommentsAttribute ";"]
10:  "}"
11: "}"
```

EXAMPLE:

Structured Type			
Name	routing_label(SLSel_Type)		
Group			
Structure	record		
Encoding	BER		
Comments	header for routing info		
Element Name	**Type Definition**	**Field Encoding**	**Comments**
DestPC	BIT_14		destination point code
OrigPC	BIT_14		origination point code
SLSel	SLSel_Type	PER	signalling link selection
Detailed Comments	overrides previous definitions		

映射为：

```
 1: module MyModule {
 2:  type record routing_label(SLSel_Type) {
 3:  BIT 14 DestPC,
 4:  BIT 14 OrigPC,
 5:  SLSel Type SLSel
 6:  } with {
 7:  encode "BER";
 8:  display "comments := header for routing info";
 9:  display (DestPC) "comments := destination point code";
10:  display (OrigPC) "comments := origination point code";
11:  display (SLSel) "comments := signalling link selection";
12:  encode (SLSel) "PER";
13:  display "detailed comments := overrides previous definition";
14:  }
15: }
```

6.6 SequenceOf 类型

SequenceOf Types					
Group	*[GroupReference]*				
Name	**Type**	**Kind**	**Length**	**Encoding**	**Comments**
. *StructTypeIdentifier* .	. *Type [SubTypeSpec]* .	. *RecordOrSet* .	. *[StringLength]* .	. *[TabFreeText]* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*				

图 7/Z.141－SequenceOf 类型标准形式

6.6.1 映射

SequenceOf 类型标准形式映射为相同群级上的 sequenceof 类型定义陈述系列。本标准形式将用于 *RecordOfDef* 和 *SetOfDef* 类型定义。

详尽注释字段映射为相关于装入群或模块的 *WithStatement* 中的显示属性。**编码**和**注释**字段分别映射为相关于各自 SequenceOf 类型定义的 *WithStatement* 中的编码和显示属性。

```
 1: module TTCN3ModuleId "{"
 2:  type record of [StringLength] Type StructTypeIdentifier [SubTypeSpec]
 3:  with {
 4:  [EncodeAttribute ";"]
 5:  [CommentsAttribute ";"]
 6:  }
 7:  type set of [StringLength] Type StructTypeIdentifier [SubTypeSpec]
 8:  with {
 9:  [EncodeAttribute ";"]
10:  [CommentsAttribute ";"]
11:  }
12: } with {
13:  [SequenceOfTypesDetailedCommentsAttribute ";"]
14: }
```

EXAMPLE:

SequenceOf Types					
Group	SequenceOfTypes/				
Name	**Type**	**Kind**	**Length**	**Encoding**	**Comments**
RecordOfIntegers	integer(1..10)	record	10	BER	ten integers
SetOfBooleans	boolean	set	3	PER	three booleans
Detailed Comments	example sequenceof types				

映射为:

```
 1: module MyModule {
 2:  group SequenceOfTypes {
 3:  type record of length(10) integer RecordOfIntegers(1..10) with {
 4:  encode "BER";
 5:  display "comments := ten integers";
 6:  }
 7:  type set of length(3) boolean SetOfBooleans with {
 8:  encode "PER";
 9:  display "comments := three booleans";
10:  }
11:  } with {
12:  display "sequenceof types detailed comments := example sequenceof types";
13:  }
14: }
```

6.7 列举类型

Enumerated Type			
Name	*EnumTypeIdentifier*		
Group	*[GroupReference]*		
Encoding	*[TabFreeText]*		
Comments	*[TabFreeText]*		
Enumeration Name		**Enumeration Value**	**Comments**
. *EnumerationIdentifier* .		. *[Number]* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*		

图 8/Z.141－列举类型标准形式

6.7.1 映射

列举类型标准形式映射为 TTCN-3 中心语言中的列举类型定义陈述。**注释**和**详尽注释**字段映射为对应 *WithStatement* 中的显示属性，**编码**字段映射为对应 *WithStatement* 中的编码属性。每个列举的**注释**字段映射为显示属性，由对应 *WithStatement* 中的 *EnumerationIdentifier* 限定。

```
 1: module TTCN3ModuleId "{"
 2:  type enumerated EnumTypeIdentifier "{"
 3:  EnumerationIdentifier ["(" Number ")"]
 4:  {"," EnumerationIdentifier ["(" Number ")"]}
 5:  } with {
 6:  [EncodeAttribute ";"]
 7:  [CommentsAttribute ";"]
 8:  {NamedValueCommentsAttribute ";"}
 9:  [DetailedCommentsAttribute ";"]
10:  }
11: }
```

EXAMPLE:

Enumerated Type			
Name	Weekdays		
Group			
Encoding	BER		
Comments	days of the week		
Enumeration Name		**Enumeration Value**	**Comments**
Monday		1	
Tuesday		2	
Wednesday		3	half way there
Thursday		4	
Friday		5	TGIF
Saturday		6	
Sunday		7	
Detailed Comments	wish it were Friday		

映射为:

```
 1: module MyModule {
 2:  type enumerated Weekdays {
 3:  Monday(1), Tuesday(2), Wednesday(3), Thursday(4), Friday(5),
 4:  Saturday(6), Sunday(7)
 5:  } with {
 6:  encode "BER";
 7:  display "comments := days of the week";
 8:  display (Wednesday) "comments := half way there";
 9:  display (Friday) "comments := TGIF";
10:  display "detailed comments := wish it were Friday";
11:  }
12: }
```

6.8 端口类型

Port Type			
Name	*PortTypeIdentifier*		
Group	*[GroupReference]*		
Communication Model	*PortModelType*		
Comments	*[TabFreeText]*		
Type/Signature		**Direction**	**Comments**
. *TypeOrSignature* .		. *InOutOrInout* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*		

图 9/Z.141－端口类型标准形式

6.8.1 映射

端口类型标准形式映射为 TTCN-3 中心语言中的端口类型定义。**注释**和**详尽注释**字段映射为对应 *WithStatement* 中的显示属性。类型和签名表的**注释**字段映射为对应 *WithStatement* 中的显示属性，由类型或签名标识符限定。通常每类型或签名一行。

如果模块中定义的所有类型或所有程序签名可以通过通信端口，则**类型/签名**字段置为关键词 **all**。

```
1: module TTCN3ModuleId "{"
2:  type port PortTypeIdentifier PortModelType "{"
3:  PortTypeDef
4:  "}" with "{"
5:  [CommentsAttribute ";"]
6:  {TypeOrSignatureCommentsAttribute ";"}
7:  [DetailedCommentsAttribute ";"]
8:  "}"
9: "}"
```

EXAMPLE:

Port Type			
Name	MyPortType		
Group			
Communication Model	message		
Comments	example port type		
Type/Signature		**Direction**	**Comments**
MsgType1		in	first comment
MsgType2		in	second comment
MsgType3		out	
Detailed Comments	detailed comment		

映射为：

```
 1: module MyModule {
 2:  type port MyPortType message {
 3:  in MsgType1;
 4:  in MsgType2;
 5:  out MsgType3;
 6:  } with {
 7:  display "comments := example port type";
 8:  display (MsgType1) "comments := first comment";
 9:  display (MsgType2) "comments := second comment";
10:  display "detailed comments := detailed comment";
11:  }
12: }
```

6.9 成分类型

Component Type			
Name	*ComponentTypeIdentifier*		
Group	*[GroupReference]*		
Comments	*[TabFreeText]*		
Local Def Name	**Type**	**Initial Value**	**Comments**
. *VarConstOrTimerIdentifier* .	. *TypeOrTimer [ArrayDef]* .	. *[ConstantExpression | Expression]* .	. *[TabFreeText]* .
Port Name	**Port Type**		**Comments**
. *PortIdentifier* .	. *PortType[ArrayDef]* .		. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*		

图 10/Z.141－成分类型标准形式

6.9.1 映射

成分类型标准形式映射为 TTCN-3 中心语言中的成分类型定义。标准形式表述为三部分。

第一部分包括头**注释**和**详尽注释**字段，改为相关于成分类型定义的 *WithStatement* 中的显示属性。

第二部分包括成分类型中规定的本地恒量、变量和定时器。这些定义可出现在中心语言的成分类型定义中，但对于标准形式，它们与端口实例分开并显示在单独的表中。这些定义的顺序将保持，因为定义可以相互依赖。**类型**栏将被置为关键词 **timer**（对于所有定时器），和置为关键词 **const** 之后的恒量类型（对于所有恒量）。通常每个恒量、变量或定时器占一行。本表中的**注释**栏改为显示属性，由相关于成分类型定义的 *WithStatement* 中的本地定义的标识符限定。

第三部分包括在成分类型中规定的端口实例。任何排列定义附加到端口类型。通常每端口实例一行。本表的**注释**栏改为显示属性，由相关于成分类型定义的 *WithStatement* 中的 *PortIdentifier* 限定。

```
 1: module TTCN3ModuleId "{"
 2:  type component ComponentTypeIdentifier "{"
 3:  var Type VarIdentifier [":=" Expression] ";"
 4:  timer TimerIdentifier [":=" Expression] ";"
 5:  const Type ConstIdentifier ":=" ConstantExpression ";"
 6:  PortList
 7:  "}" with "{"
 8:  [CommentsAttribute ";"]
 9:  {PortCommentsAttribute ";"}
10:  [DetailedCommentsAttribute ";"]
11:  "}"
12: "}"
```

EXAMPLE:

Component Type			
Name	MyComponentType		
Group			
Comments	an example component type		
Local Def Name	**Type**	**Initial Value**	**Comments**
PI	const float	3.14	the ratio
x	float	PI * 2	double PI
t1	timer	15 min	a 15 second timer
Port Name	**Port Type**		**Comments**
PCO1	MyMessagePortType		first comment
PCO2	MyProcedurePortType		second comment
Detailed Comments	detailed comments		

映射为:

```
 1: module MyModule {
 2:  type component MyComponentType {
 3:  const float PI := 3.14;
 4:  var float x := PI * 2;
 5:  timer t1 := 15;
 6:  port MyMessagePortType PCO1;
 7:  port MyProcedurePortType PCO2;
 8:  } with {
 9:  display "comments := an example component type";
10:  display (PI) "comments := the ratio";
11:  display (x) "comments := double PI";
12:  display (t1) "comments := a 15 second timer";
13:  display (PCO1) "comments := first comment";
14:  display (PCO2) "comments := second comment";
15:  display "detailed comments := detailed comments";
16:  }
17: }
```

6.10 恒量

Constants			
Group	*[GroupReference]*		
Name	**Type**	**Value**	**Comments**
. *ConstIdentifier* | *ExtConstIdentifier* .	. *Type [ArrayDef]* .	. *ConstantExpression* | **external**	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*		

图 11/Z.141－恒量标准形式

6.10.1 映射

恒量标准形式映射为相同群级上恒量和外部恒量定义陈述系列。**详尽注释**字段映射为相关于装入群或模块的 *WithStatement* 中的显示属性。**注释**字段映射为相关于各自恒量定义的 *WithStatement* 中的显示属性。对于外部恒量，**值**字段置为关键词 **external**。

```
1:    module TTCN3ModuleId "{"
2:     const Type ConstIdentifier[ArrayDef] ":=" ConstantExpression with "{"
3:     [CommentsAttribute ";"]
4:     "}"
5:     external const Type ConstIdentifier with "{"
6:     [CommentsAttribute ";"]
7:     "}"
8:    "}" with "{"
9:     [ConstantsDetailedCommentsAttribute ";"]
10:   "}"
```

EXAMPLE:

Constants			
Group	Constants1		
Name	**Type**	**Value**	**Comments**
TOTO	integer	external	defined somewhere else
SEL2	boolean	(5 + TOTO) < 10	TOTO limit reached
T1	integer[1..3]	{1,3,2}	
Detailed Comments	detailed comments		

映射为:

```
 1: module MyModule {
 2:  group Constants1 {
 3:  external const integer TOTO with {
 4:  display "comments := defined somewhere else";
 5:  }
 6:  const boolean SEL2 := (5 + TOTO) < 10 with {
 7:  display "comments := TOTO limit reached";
 8:  }
 9:  const integer T1[1..3] := {1,3,2};
10:  } with {
11:  display "detailed comments := detailed comments";
12:  }
13: }
```

6.11 信号定义

Signature Definition		
Name	*SignatureIdentifier*([*SignatureFormalParList*])	
Group	*[GroupReference]*	
Return Type	*[Type]* | **noblock**	
Comments	*[TabFreeText]*	
Exception Type		**Comments**
. *[ExceptionType]* .		. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*	

图 12/Z.141－信号定义标准形式

6.11.1 映射

信号定义标准形式映射为 TTCN-3 中心语言中的信号定义。**注释**和**详尽注释**字段映射为对应 *WithStatement* 中的显示属性。例外表的**注释**字段映射为显示属性，由对应 *WithStatement* 中的例外类型限定。非成组程序将关键词 **noblock** 规定为返回类型。

```
 1: module TTCN3ModuleId "{"
 2:  signature SignatureIdentifier "(" [SignatureFormalParList] ")"
 3:  [return Type | noblock]
 4:  [exception "(" ExceptionTypeList ")"]
 5:  with "{"
 6:  [CommentsAttribute ";"]
 7:  [ExceptionCommentsAttribute ";"]
 8:  [DetailedCommentsAttribute ";"]
 9:  "}"
10: "}"
```

EXAMPLE:

Signature Definition		
Name	read(integer fields, inout charstring buf, integer nbyte)	
Group		
Return Type	integer	
Comments	reads from a file	
Exception Type		**Comments**
integer		error code
MyException		user defined
Detailed Comments	required: unistd.h	

映射为:

```
 1: module MyModule {
 2:  signature read syscall(in integer fields,
 3:  inout charstring buf,
 4:  in integer nbyte)
 5:  return integer
 6:  exception (integer)
 7:  with {
 8:  display "comments := reads from a file";
 9:  display (integer) "comments := error code of system call";
10:  display "detailed comments := required: unistd.h";
11:  }
12: }
```

6.12 简单模板

Simple Templates					
Group	*[GroupReference]*				
Name	**Type**	**Derived**	**Value**	**Encoding**	**Comments**
. *Template Identifier* .	. *BaseTemplate* .	. *[DerivedDef]* .	. *TemplateBody* .	. *[TabFreeText]* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*				

图 13/Z.141－简单模板标准形式

6.12.1 映射

简单模板标准形式映射为相同群级上简单模板定义陈述系列。简单模板定义为将 *SimpleSpec* 或 *ArrayValueOrAttrib* 作为 *TemplateBody* 的所有模板定义。在简单类型中规定对应类型、SequenceOf Type 和列举类型标准形式。

详尽注释字段映射为相关于装入群或模块的 *WithStatement* 中的显示属性。**注释**和**编码**字段映射为显示和编码属性，由相关于各自的简单模板定义陈述的 *WithStatement* 中的 *TemplateIdentifier* 限定。

```
1: module TTCN3ModuleId "{"
2:  template BaseTemplate[DerivedDef] := TemplateBody with "{"
3:  [EncodeAttribute ";"]
4:  [CommentsAttribute ";"]
5:  "}"
6: "}" with "{"
7:  [SimpleTemplatesDetailedCommentsAttribute ";"]
8: "}"
```

EXAMPLE:

Simple Templates					
Group	SimpleTemplates1				
Name	**Type**	**Derived**	**Value**	**Encoding**	**Comments**
MyTemplate1	MyType1		3	BER	foobar
MyTemplate11(integer index)	MyType1	MyTemplate1	3*index	PER	the current index
Detailed Comments	an example				

映射为:

```
 1: module MyModule {
 2:  group SimpleTemplates {
 3:  template MyType1 MyTemplate1 with {
 4:  encode "BER";
 5:  display "comments := foobar";
 6:  }
 7:  template MyType1 MyTemplate11(integer index)
 8:  modifies MyTemplate1 := 3 * index
 9:  with {
10:  encode "PER";
11:  display "comments := the current index";
12:  }
13:  } with {
14:  display "simple templates detailed comments := an example";
15:  }
16: }
```

6.13 构建模板

Structured Template			
Name	*TemplateIdentifier[(TemplateFormalParList)]*		
Group	*[GroupReference]*		
Type/Signature	*TypeIdentifier | SignatureIdentifier*		
Derived From	*[TemplateRef]*		
Encoding	*[TabFreeText]*		
Comments	*[TabFreeText]*		
Element Name	**Element Value**	**Element Encoding**	**Comments**
. *FieldReference* .	. *FieldValueOrAttrib* .	. *[TabFreeText]* .	. *[TabFreeText]* .
Detailed Comments	*[TabFreeText]*		

图 14/Z.141－构建模板标准形式

6.13.1 映射

构建模板标准形式映射为 TTCN-3 构建模板定义陈述。构建模板定义为将 *FieldSpecList* 作为模板体的全部模板定义。对应的类型在构建类型标准形式中规定。

注释和**详尽注释**字段映射为相关于构建模板定义的 *WithStatement* 中的显示属性，**编码**字段映射为相关于构建模板定义的 *WithStatement* 中的编码属性。

单元表的**注释**字段映射为显示属性，由相关于构建模板定义的 *WithStatement* 中的字段参考限定。**单元编码**字段映射为编码属性，由相关于构建模板定义的 *WithStatement* 中的字段参考限定。

```
module TTCN3ModuleId "{"
 template BaseTemplate [DerivedDef] ":=" TemplateBody with "{"
 [EncodeAttribute ";"]
 [CommentsAttribute ";"]
 [FieldEncodeAttribute ";"]
 [FieldCommentsAttribute ";"]
 [DetailedCommentsAttribute ";"]
 "}"
"}"
```

EXAMPLE:

Structured Template			
Name	MyStructuredTemplate11(integer para1, boolean para2)		
Group			
Type/Signature	MyStructuredType		
Derived From	MyStructuredTemplate1		
Encoding	BER		
Comments	example structured template		
Element Name	**Element Value**	**Element Encoding**	**Comments**
field1	13		first field
field2	para2	PER	second field
field3	para1		third field
Detailed Comments	detailed comments		

映射为:

```
module MyModule {
 template MyStructuredType MyStructuredTemplate11(integer para1,
 boolean para2)
 modifies MyStructuredTemplate1 := {
 field1 := 13,
 field2 := para2,
 field3 := para1
 } with {
 encode "BER";
 display "comments := example structured template";
 display (field1) "comments := first field";
 encode (field2) "PER";
 display (field2) "comments := second field";
 display (field3) "comments := third field";
 display "detailed comments := detailed comments";
 }
}
```

6.14 功能

Function			
Name	*FunctionIdentifier([FunctionFormalParList])*		
Group	*[GroupReference]*		
Runs On	*[ComponentType]*		
Return Type	*[Type]*		
Comments	*[TabFreeText]*		
Local Def Name	**Type**	**Initial Value**	**Comments**
. *VarConstOrTimerIdentifier* .	. *TypeOrTimer* .	. *[Expression | ConstantExpression]*	. *[TabFreeText]* .
Behaviour			
. *FunctionStatement* | **external** .			
Detailed Comments	*[TabFreeText]*		

图 15/Z.141－功能标准形式

6.14.1 映射

功能标准形式映射为 TTCN-3 功能定义陈述或外部功能定义。它表述为三部分。

第一部分包括头字段。**注释**和**详尽注释**字段映射为相关于功能定义的 *WithStatement* 中的显示属性。

第二部分包括功能定义中规定的本地恒量、变量和定时器。这些定义可出现在中心语言的功能体中，但对于标准形式，它们与功能体的其他部分分开并在单独的表中显示。定义的顺序将保持，因为定义可以相互依赖。**类型**栏将被置为关键词 **timer**（对所有定时器），以及置为关键词 **const** 之后的恒量类型（对所有恒量）。**注释**字段改为显示属性，由相关于功能定义的 *WithStatement* 中的本地标识符限定。

第三部分由 TTCN-3 中心语言的功能体减去本地恒量、变量和定时器组成。

对于外部功能，行为只包括关键词 **external**。

```
 1: module TTCN3ModuleId "{"
 2:  function FunctionIdentifier "(" [FunctionFormalParList] ")"
 3:  [runs on ComponentType]
 4:  [return Type] "{"
 5:  var Type VarIdentifier [":=" Expression] ";"
 6:  timer TimerIdentifier [":=" Expression]";"
 7:  const Type ConstIdentifier ":=" ConstantExpression ";"
 8:  {FunctionStatement}
 9:  "}" with "{"
10:  [CommentsAttribute ";"]
11:  [VarConstOrTimerCommentsAttribute ";"]
12:  [DetailedCommentsAttribute ";"]
13:  "}"
14: "}"
```

EXAMPLE:

Function			
Name	MyFunction(integer para1)		
Group			
Runs On	MyComponentType		
Return Type	boolean		
Comments	example function definition		
Local Def Name	**Type**	**Initial Value**	**Comments**
MyLocalVar	boolean	false	local variable
MyLocalConst	const float	60	local constant
MyLocalTimer	timer	15 * MyLocalConst	local timer
Behaviour			

if (para1 == 21) {
 MyLocalVar := true;
}
if (MyLocalVar) {
 MyLocalTimer.start;
 MyLocalTimer.timeout;
}
return (MyLocalVar);			

Detailed Comments	detailed comments		

映射为：

```
 1: module MyModule {
 2:  function MyFunction(in integer para1)
 3:  runs on MyComponentType
 4:  return boolean {
 5:  var boolean MyLocalVar := false;
 6:  const float MyLocalConst := 60;
 7:  timer MyLocalTimer := 15 * MyLocalConst;
 8:
 9:  if (para1 == 21) {
10:  MyLocalVar := true;
11:  }
12:  if (MyLocalVar) {
13:  MyLocalTimer.start;
14:  MyLocalTimer.timeout;
15:  }
16:  return (MyLocalVar);
17:  } with {
18:  display "comments := example function definition";
19:  display (MyLocalVar) "comments := local variable";
20:  display (MyLocalConst) "comments := local constant";
21:  display (MyLocalTimer) "comments := local timer";
22:  display "detailed comments := detailed comments";
23:  }
24: }
```

6.15 Altstep

Altstep			
Name	*AltstepIdentifier([AltstepFormalParList])*		
Group	*[GroupReference]*		
Purpose	*[TabFreeText]*		
Runs On	*[ComponentType]*		
Comments	*[TabFreeText]*		
Local Def Name	**Type**	**Initial Value**	**Comments**
. *VarConstOrTimerIdentifier* .	. *TypeOrTimer [ArrayDef]* .	. *[Expression | ConstantExpression]* .	. *[TabFreeText]* .
Behaviour			
. *AltGuardList* .			
Detailed Comments	*[TabFreeText]*		

图 16/Z.141－Altstep 标准形式

6.15.1 映射

Altstep 标准形式映射为 TTCN-3 altstep 定义陈述。它表述为三个部分。

第一部分包括头字段。**目的**、**注释**和**详尽注释**字段映射为相关于 altstep 定义的 *WithStatement* 中的显示属性。

第二部分包括 altstep 定义中规定的本地恒量、变量和定时器。这些定义可出现在中心语言的 altstep 体中，但对于标准形式，它们与 altstep 体的其他部分分开并在单独的表中显示。定义的顺序保持，因为定义可以相互依赖。**类型**栏将被置为关键词 **timer**（对所有定时器），以及置为关键词 **const** 之后的恒量类型（对所有恒量）。**注释**字段改为显示属性，由相关于 altstep 定义的 *WithStatement* 中的本地标识符限定。

第三部分包括 TTCN-3 中心语言的 altstep 的 *AltGuardList*。

```
module TTCN3ModuleId "{"
 teststep AltstepIdentifier "(" [AltstepFormalParList] ")"
 [runs on ComponentType] "{"
 AltGuardList
 "}" with "{"
 [PurposeAttribute ";"]
 [CommentsAttribute ";"]
 [VarConstOrTimerCommentsAttribute ";"]
 [DetailedCommentsAttribute ";"]
 "}"
"}"
```

EXAMPLE:

Altstep			
Name	MyAltstep(integer para1)		
Group			
Runs On	MyComponentType		
Purpose	to do something		
Comments	example altstep definition		
Local Def Name	**Type**	**Initial Value**	**Comments**
MyLocalVar	boolean	false	local variable
MyLocalConst	const float	60	local constant
MyLocalTimer	timer	15 * MyLocalConst	local timer
Behaviour			

[] PCO1.receive(MyTemplate(para1, CompVar) {
verdict.set(inconc);
}
[] PCO2.receive {
repeat;
}
[] CompTimer.timeout {
verdict.set(fail);
stop;
}			

Detailed Comments	detailed comments		

映射为:

```
module MyModule {
 altstep MyTeststep(integer para1) runs on MyComponentType {
 var boolean MyLocalVar := false;
 const float MyLocalConst := 60;
 timer MyLocalTimer := 15 * MyLocalConst;

 [] PCO1.receive(MyTemplate(para1, CompVar)) {
 verdict.set(inconc);
 }
 [] PCO2.receive {
 repeat;
 }
 [] CompTimer.timeout {
 verdict.set(fail);
 stop;
 }
 } with {
 display "purpose := to do something";
 display "comments := example altstep definition";
 display "detailed comments := detailed comments";
 }
}
```

6.16 Testcase

Testcase			
Name	*TestcaseIdentifier([TestcaseFormalParList])*		
Group	*[GroupReference]*		
Purpose	*[TabFreeText]*		
System Interface	*[ComponentType]*		
MTC Type	*ComponentType*		
Comments	*[TabFreeText]*		
Local Def Name	**Type**	**Initial Value**	**Comments**
. *VarConstOrTimerIdentifier* .	. *TypeOrTimer* .	. *[Expression | ConstantExpression]*	. *[TabFreeText]* .
Behaviour			
. *FunctionStatement* .			
Detailed Comments	*[TabFreeText]*		

图 17/Z.141－Testcase 标准形式

6.16.1 映射

Testcase 标准形式映射为 TTCN-3 testcase 定义陈述。它表述为三个部分。

第一部分包括头字段。**目的**、**注释**和**详尽注释**字段映射为相关于测试情况定义的 *WithStatement* 中的显示属性。

第二部分包括 testcase 定义中规定的本地恒量、变量和定时器。这些定义可出现在中心语言的 testcase 体中，对于标准形式，他们与 testcase 体的其他部分分开并显示在单独的表中。定义的顺序将保持，因为定义可以相互依赖。**类型**栏将被置为关键词 **timer**（对所有定时器），以及置为关键词 **const** 之后的恒量类型（对所有恒量）。**注释**字段改为显示属性，由相关于 testcase 定义的 *WithStatement* 中的本地标识符限定。

第三部分包括 TTCN-3 中心语言的 testcase 体减去本地恒量、变量和定时器。

```
 1: module TTCN3ModuleId "{"
 2:  testcase TestcaseIdentifier [TestcaseFormalParList]
 3:  [runs on ComponentType]
 4:  [system ComponentType] "{"
 5:  var Type VarIdentifier [":=" Expression] ";"
 6:  timer TimerIdentifier [":=" Expression] ";"
 7:  const Type ConstIdentifier ":=" ConstantExpression;
 8:  {FunctionStatement}
 9:  "}" with "{"
10:  [CommentsAttribute ";"]
11:  [PurposeAttribute ";"]
12:  [VarConstOrTimerCommentsAttribute ";"]
13:  [DetailedCommentsAttribute ";"]
14:  "}"
15: "}"
```

EXAMPLE:

Testcase			
Name	MyTestcase(integer para1)		
Group			
Purpose	do something useful		
System Interface	MyComponentType		
MTC Type	MyComponentType		
Comments	example testcase definition		
Local Def Name	**Type**	**Initial Value**	**Comments**
MyLocalVar	boolean	false	local variable
MyLocalConst	const float	60	local constant
MyLocalTimer	timer	15 * MyLocalConst	local timer

Behaviour

```
default.activate { [expand] OtherwiseFail(); }; /* Default activation */
ISAP1.send(ICONreq {}); /* Inline template definition */
alt {
    [] MSAP2.receive(Medium_Connection_Request()) { /* use of a template */
        MSAP2.send(MDATreq Medium_Connection_Confirmation());
        alt {
            [] ISAP1.receive(ICONconf {} ); {
                ISAP1.send(Data_Request(TestSuitePar) );
                alt {
                    [] MSAP2.receive(Medium_Data_Transfer()) {
                        MSAP2.send(MDATreq cmi_synch1());
                        ISAP1.send(IDISreq {});
                    }
                    [] ISAP1.receive(IDISind {}) {
                        verdict.set(inconclusive);
                        stop();
                    }
                }
            }
            [] MSAP2.receive(MDATind_Connection_Request()) {
                verdict.set(inconclusive);
                stop();
            }
            [] ISAP1.receive(IDISind {}) {
                verdict.set(inconclusive);
                stop();
            }
        }
    }
    [] ISAP1.receive(IDISind {}) {
        verdict.set(inconclusive);
        stop();
    }
}
```

Detailed Comments	detailed comments

映射为：

```
1:    module MyModule {
2:        testcase MyTestcase(in integer para1)
3:            runs on MyComponentType
4:            system MyComponentType {
5:            var boolean MyLocalVar := false;
6:            const float MyLocalConst := 60;
7:            timer MyLocalTimer := 15 * MyLocalConst;
8:            var default MyDefault := activate(OtherwiseFail());
9:
10:           ISAP1.send(ICONreq:{}); /* Inline template definition */
11:           alt {
12:               /* use of a template */
13:               [] MSAP2.receive(Medium_Connection_Request()) {
14:       MSAP2.send(MDATreq:Medium Connection Confirmation());
15:                   alt {
16:                       [] ISAP1.receive(ICONconf:{}) {
17:                           ISAP1.send(Data Request(TestSuitePar));
18:                           alt {
19:                               [] MSAP2.receive(Medium_Data_Transfer()) {
20:                                   MSAP2.send(MDATreq:cmi synch1());
21:                                   ISAP1.send(IDISreq:{});
22:                               }
23:                               [] ISAP1.receive(IDISind:{}) {
24:                                   verdict.set(inconc);
25:                                   stop;
26:                               }
27:                           }
28:                       }
29:                       [] MSAP2.receive(MDATind_Connection_Request()) {
30:                           verdict.set(inconc);
31:                           stop;
32:                       }
33:                       [] ISAP1.receive(IDISind:{}) {
34:                           verdict.set(inconc);
35:                           stop;
36:                       }
37:                   }
38:               }
39:               [] ISAP1.receive(IDISind:{}) {
40:                   verdict.set(inconc);
41:                   stop;
42:               }
43:           }
44:       } with {
45:           display "purpose := do something useful";
46:           display "comments := example testcase definition";
47:           display (MyLocalVar) "comments := local variable";
48:           display (MyLocalConst) "comments := local constant";
49:           display (MyLocalTimer) "comments := local timer";
50:           display "detailed comments := detailed comments";
51:       }
      }
```

7 BNF 产品

1. TabFreeText ::= [ExtendedAlphaNum]

2. GroupReference ::= {GroupIdentifier "/"}+

3. EncRuleIdentifier ::= Identifier

4. CommentsAttribute :: = **display** """ "comments" ":=" TabFreeText """

5. DetailedCommentsAttribute :: = **display** """ "detailed comments" ":=" TabFreeText """

6. TTCN3ModuleId ::= ModuleIdentifier [DefinitiveIdentifier]

```
7.  ModuleAttributes ::= TabularPresentationFormatAttribute ";"
    ModuleVersionAttribute ";"
    ModuleDateAttribute ";"
    ModuleBaseStandardRefAttribute ";"
    ModuleTestStandardRefAttribute ";"
    ModulePICSRefAttribute ";"
    ModulePIXITRefAttribute ";"
    ModuleTestMethodAttribute ";"
    ModuleCommentsAttribute ";"
    ModuleDetailedCommentsAttribute ";"

8.  TabularPresentationFormatAttribute ::=
    display """ "presentation format := ETSI Tabular version" MajorVersion "." MinorVersion """

9.  MajorVersion ::= Number

10. MinorVersion ::= Number

11. ModuleVersionAttribute ::=
    display """ "module version" ":=" TabFreeText """

12. ModuleDateAttribute ::=
    display """ "module date" ":=" TabFreeText """

13. ModuleBaseStandardRefAttribute ::=
    display """ "module base standards ref" ":=" TabFreeText """

14. ModuleTestStandardRefAttribute ::=
    display """ "module test standards ref" ":=" TabFreeText """

15. ModulePICSRefAttribute ::=
    display """ "module pics ref" ":=" TabFreeText """

16. ModulePIXITRefAttribute ::=
    display """ "module pixit ref" ":=" TabFreeText """

17. ModuleTestMethodAttribute ::=
    display """ "module test method" ":=" TabFreeText """

18. ModuleCommentsAttribute ::=
    display """ "module comments" ":=" TabFreeText """

19. ModuleDetailedCommentsAttribute ::=
    display """ "module detailed comments" ":=" TabFreeText """

20. ModuleParPicsPixitRefAttribute ::=
    display "(" ModuleParIdentifier ")"
    """ "pics/pixit ref" ":=" TabFreeText """

21. ModuleParComments ::=
    display "(" ModuleParIdentifier ")"
    """ "comments" ":=" TabFreeText """

22. ImportsSourceRefAttribute ::=
    display """ "imports source ref" ":=" TabFreeText """

23. ImportsSourceDefinitionCommentsAttribute ::=
    display "(" ImportIdentifier ")"
    """ "comments" ":=" TabFreeText """

24. ImportSpecification ::= ( (Identifier | FullGroupIdentifier) | AllKeyword ) [ ExceptionsDef ]
    /* STATIC SEMANTIC: FullGroupIdentifier shall only be used for group imports. */

25. EncodeAttribute ::= encode """ TabFreeText """

26. SimpleTypesDetailedCommentsAttribute ::=
    display """ "simple types detailed comments" ":=" TabFreeText """

27. StructureType ::= record | union | set

28. FieldCommentsAttribute ::=
    display "(" FieldIdentifier ")" """ "comments" ":=" TabFreeText """

29. FieldEncodeAttribute ::=
    encode "(" FieldIdentifier ")" """ TabFreeText """

30. SequenceOfTypesDetailedCommentsAttribute ::=
    display """ "sequenceof types detailed comments" ":=" TabFreeText """
```

```
31. NamedValueCommentsAttribute ::=
    display "(" NamedValueIdentifier ")"
    """ "comments" ":=" TabFreeText """

32. TypeOrSignatureCommentsAttribute ::=
    display "(" TypeOrSignatureIdentifier ")"
    """ "comments" ":=" TabFreeText """

33. PortCommentsAttribute ::=
    display "(" PortIdentifier ")"
    """ "comments" ":=" TabFreeText """

34. ConstantsDetailedCommentsAttribute ::=
    display """ "simple types detailed comments" ":=" TabFreeText """

35. ExceptionCommentsAttribute ::=
    display "(" Type ")"
    """ "comments" ":=" TabFreeText """

36. VarConstOrTimerCommentsAttribute ::=
    display "(" VarConstOrTimerIdentifier ")"
    """ "comments" ":=" TabFreeText """

37. PurposeAttribute ::= display """ "purpose" ":=" TabFreeText """

38. SimpleTemplatesDetailedCommentsAttribute ::= display """ "simple templates detailed comments"
    ":=" TabFreeText """
```

ITU-T 系列建议书

A系列	ITU-T工作的组织
D系列	一般资费原则
E系列	综合网络运行、电话业务、业务运行和人为因素
F系列	非话电信业务
G系列	传输系统和媒质、数字系统和网络
H系列	视听及多媒体系统
I系列	综合业务数字网
J系列	有线网络和电视、声音节目及其它多媒体信号的传输
K系列	干扰的防护
L系列	电缆和外部设备其它组件的结构、安装和保护
M系列	电信管理，包括TMN和网络维护
N系列	维护：国际声音节目和电视传输电路
O系列	测量设备的技术规范
P系列	电话传输质量、电话设施及本地线路网络
Q系列	交换和信令
R系列	电报传输
S系列	电报业务终端设备
T系列	远程信息处理业务的终端设备
U系列	电报交换
V系列	电话网上的数据通信
X系列	数据网、开放系统通信和安全性
Y系列	全球信息基础设施、互联网协议问题和下一代网络
Z系列	**用于电信系统的语言和一般软件问题**

瑞士印刷
2006 年，日内瓦