



# 中华人民共和国国家标准

GB/T 16262—1996  
ISO 8824:1990

---

## 信息处理系统 开放系统互连 抽象语法记法—(ASN.1)规范

Information processing system  
—Open systems interconnection  
—Specification of abstract syntax  
notation one (ASN.1)

1996-03-22 发布

1996-10-01 实施

---

国家技术监督局 发布



# 目 次

0	引言	( 1)
1	主题内容与适用范围	( 3)
2	引用标准	( 3)
3	术语	( 3)
4	缩写	( 6)
5	本标准中使用的记法	( 6)
6	ASN.1 记法的使用	( 8)
7	ASN.1 字符集	( 8)
8	ASN.1 项	( 9)
9	模块定义	(13)
10	引用类型和值定义	(15)
11	类型和值的赋值	(15)
12	类型和值的定义	(15)
13	布尔类型的记法	(17)
14	整数类型的记法	(17)
15	枚举类型的记法	(17)
16	实数类型的记法	(18)
17	位串类型的记法	(18)
18	八位位组串类型的记法	(19)
19	空类型的记法	(19)
20	序列类型的记法	(20)
21	单纯序列类型的记法	(20)
22	集合类型的记法	(21)
23	单纯集合类型的记法	(21)
24	值选择类型的记法	(21)
25	类型选择类型的记法	(23)
26	有标记类型的记法	(23)
27	任意类型的记法	(24)
28	客体标识符类型的记法	(24)
29	字符串类型的记法	(25)
30	第 32~35 章中定义的类型的记法	(25)
31	字符串类型的定义	(26)
32	广义时间	(27)
33	世界时间	(27)
34	外部类型	(28)

35	客体描述符类型 .....	(30)
36	子类型记法 .....	(30)
37	子类型值集合 .....	(31)
附录 A	宏记法(补充件) .....	(34)
附录 B	ISO 对 OBJECT IDENTIFIER 成分值的赋值(补充件) .....	(38)
附录 C	CCITT 对 OBJECT IDENTIFIER 成分值的赋值(补充件) .....	(39)
附录 D	联合体对 OBJECT IDENTIFIER 成分值的赋值(补充件) .....	(39)
附录 E	例子和提示(参考件) .....	(40)
附录 F	ASN.1 记法总结(参考件) .....	(50)

# 中华人民共和国国家标准

## 信息处理系统 开放系统互连 抽象语法记法—(ASN.1)规范

Information processing system  
—Open systems interconnection  
—Specification of abstract syntax  
notation one (ASN.1)

GB/T 16262—1996  
ISO 8824:1990

本标准等同采用国际标准 ISO 8824:1990《信息处理系统 开放系统互连 抽象语法记法—(ASN.1)规范》。

### 0 引言

在 GB 9387 基本参考模型的较低层中,服务原语的各个用户数据参数都表示为八位位组序列的二进制值。

在表示层中,用户数据参数的性质有变化。应用层标准要求表示层服务用户数据(见 GB/T 15695)以携带较为复杂的类型值,可能包括来自各种字符集的字符串。为了规定所携带的值,要求一种已定义的记法,但该记法不决定值的表示。这由一个或多个称作“编码规则”的算法来补充,此“编码规则”确定携带这种应用层值的会话层八位位组的值(称为“传送语法”)。表示层协议(见 GB/T 15696)可以协商使用何种传送语法。

确定一个值是为了使它区别于其他可能的值。该值与其他要区别的那些值一起的集合称为“类型”,一个特定的实例是该类型的一个值。更一般地,值或类型常被认为由几个更简单的值或类型以及它们间的关系组成。术语“数据类型”常用作类型的同义词。

为了正确地解释值所代表的内容(纸上的记号或是通信线路中的位),必须知道(通常从上下文中)所表示的值的类型。因此,类型的标识是本标准的一个重要部分。

定义复杂类型的非常普通的技术是先定义少量“简单类型”,这些简单类型通过定义它们的所有可能值来获得,然后用不同方式组合这些简单类型。下面是几种定义新类型的方式:

a) 给出已有类型的(有序)表,从已有类型中取出每个值,这些值的(有序)序列形成一个值;按这种方法得到的所有的值的集合是一个新类型(若表中的已有类型都是不同的,则允许省略某些值以扩充这个机制);

b) 给出(不同的)已有类型的表,从已有类型中取出每个值,这些值的(无序)集形成一个值;按此法得到的所有可能值的集合是一个新类型(通过省略某些值也能扩展该机制);

c) 给出单个已有类型,其值的(有序)序列或零个、一个或几个已有类型的值的(无序)集形成一个值;按此法得到的所有可能值的(无限)集合是一个新类型;

d) 给出(不同)类型的表,选择其中任何一个类型的值作为一个值;按此法得到的所有可能值的集合是一个新类型;

e) 给出一个类型,通过使用它的值的某些结构和序关系,它的一个子集可以形成一个新类型;

按上述方法定义的类型称为结构类型。

对每个用本标准的记法定义的类型指派了一个“标记”。这个标记由本标准或者该记法的使用者定义。

相同的标记指派给不同的类型是常见的,这些特定的类型由标记的上下文来识别。

该记法的使用者可以选择指派不同的标记给某单个类型的两次出现,从而产生两个不同的类型。如同上述 d) 情况下,需要区别所作哪一个值选择时,这是必要的。

在该记法中规定了四种类别。

第一类是“通用”类。通用类标记仅用于本标准中,每个标记指派给:

- a) 单个类型;或
- b) 一个构造机制。

第二类是“应用”类。应用类标记由其他标准指派一些类型。在一个特定标准中,一个应用类标记只指派给一个类型。

第三类型是“专用”类。专用类标记不由标准指派。其用法由使用者决定。

最后一个标记类是“上下文指定”类。在该记法的任何使用中自由指派,并按照它使用的上下文来解释。

标记主要为机器使用,对于本标准中的与人有关的记法,它是不必出现的。但是,要求区别某些类型时,则要使它们有不同的标记。因此标记的分配是使用该记法的一个重要部分。

注:① 本标准的记法中的所有类型都有一个标记,给定任何类型,该记法的使用者能用不同的标记定义新类型。

② 对某类型的值的标记,编码规则总是显式或隐式地有该类型的标记,该记法的使用上的限制是为了保证:在假定适用的类型定义可用时,确保标记可无二义地确定实际类型。

本标准规定的记法,既能定义复杂的类型,也能规定这些类型的值,但并不决定传送时一个实际类型将如何表示(通过八位位组的序列)。具有这样的特性的记法称为“用于抽象语法定义的记法”。

本标准的目的是规定一个用于抽象语法定义的记法,叫做“抽象语法记法一”,或“ASN.1”,也用作定义协议的半形式化工具。该记法不必预先作无二义的说明,该记法的使用者有责任确保它们的规范是无二义的。

本标准由规定“编码规则”的标准支持。对 ASN.1 定义的类型值,使用编码规则后产生该值的表示的完整规范(传送语法)。

本标准与 CCITT 建议 X.208(1988)在技术上是相容的。

本标准的第 7 章至第 30 章(包含这两章)定义 ASN.1 支持的简单类型,并描述用于引用简单类型及定义结构类型的一些记法。第 7 章至第 30 章也描述了 ASN.1 定义的类型值的记法。

本标准的第 31 章定义另外的类型(字符集串),通过对字符集的应用编码规则,该类型可以等同于八位位组串类型。

本标准的第 32 章至第 35 章(包含这两章)定义某些结构类型,这些类型被认为是通用的,但它们不需要另外的编码。

注:希望增加这些章使之包括其他普通数据类型,如诊断、授权信息、计费信息、安全参数等。

值记法和在这些章中定义的类型语义定义是从使用 ASN.1 记法的类型定义导出的。此类型定义能被定义编码规则的标准引用,对这些类型规定编码。

本标准的第 36 章至第 37 章定义了一个记法,它可以使子类型从双亲类型的值定义得到。

附录 A(补充件),描述一个基本 ASN.1 记法的扩展记法,称为宏业务。

附录 B(补充件),ISO 支持的管理部门对客体标识符树的定义。

附录 C(补充件),CCITT 支持的管理部门对客体标识符树的定义。

附录 D(补充件),ISO 和 CCITT 联合使用对客体标识符树的定义。

附录 E(补充件),提供使用 ASN.1 表示的例子和提示。

附录 F(参考件),用第五章的记法给出 ASN.1 的提要。

在本标准的末尾给出了包括关键字、类型和主要术语的索引。

本标准文本,特别是附录 B(补充件)到附录 D(补充件),是以联合 ISO-CCITT 的协定为主题的。

## 1 主题内容与适用范围

本标准规定一个用于抽象语法的记法,叫做抽象语法记法一(ASN.1)。

本标准定义了一些简单类型及其标记,并规定引用这些类型和这些类型值的记法。

本标准定义了从较基本类型构造新类型的机制,并规定定义这些结构类型和给它们指派标记,以及规定这些类型值的记法。

本标准(通过引用其他的标准)定义在 ASN.1 范围使用的字符集。

本标准定义一些有用的类型(使用 ASN.1),它们可供 ASN.1 使用者使用。

无论何时需要定义信息的抽象语法都可应用 ASN.1 记法。它尤其用于应用层协议标准,但并不仅仅如此。

ASN.1 记法也被其他表示层标准引用。这些标准为简单类型、结构类型、字符串类型和 ASN.1 的其他有用类型定义编码规则。

## 2 引用标准

GB 2659 世界各国和地区名称代码

ISO 6523 数据交换 组织标记的结构

GB/T 7408 数据元和交换格式 信息交换 日期和时间表示法

GB 9387 信息处理系统 开放系统互连 基本参考模型

GB/T 12054 数据处理 转义序列的登记规程

GB/T 15695—1995 信息处理系统 开放系统互连 面向连接的表示服务定义

GB/T 15696—1995 信息处理系统 开放系统互连 面向连接的表示协议规范

GB/T 16263—1996 信息处理系统 开放系统互连 用于抽象语法记法一(ASN.1)基本编码规则规范

CCITT X.208(1988) 抽象语法记法 1(ASN.1)规范

CCITT X.121(1988) 公用数据网用的国际编号方案

## 3 术语

本标准使用 GB 9387 中的术语。

### 3.1 值 value

值集合中可区分的成员。

### 3.2 类型 type

已命名的值集合。

### 3.3 简单类型 simple type

通过直接规定类型值集合而定义的类型。

### 3.4 结构类型 structured type

通过引用一个或多个其他类型而定义的类型。

### 3.5 成分类型 component type

定义结构类型时所引用的类型之一。

### 3.6 标记 tag

与每个 ASN.1 类型关联的类型记号。

### 3.7 置标记 tagging

用规定的标记替换某个类型现存的(可能是默认的)标记。

### 3.8 ASN.1 字符集 ASN.1 character set

在第7章中规定的用于ASN.1记法的字符集。

### 3.9 项 items

取自ASN.1字符集的已命名的字符序列,用于形成ASN.1记法,在第8章中规定。

### 3.10 类型(或值)引用名 type(or value)reference name

在某个上下文中唯一与一个类型(或值)相联系的名字。

注:引用名指派给本标准中定义的类型,在ASN.1中这是普遍有效的。其他引用名在其他标准中定义,并只在定义它的标准的上下文中出现。

### 3.11 ASN.1 编码规则 ASN.1 encoding rules

在传送任何ASN.1类型值期间规定其表示的规则;ASN.1编码规则使得传送的信息能够被接受者识别为特定ASN.1类型的特定值。

### 3.12 字符串类型 character string type

其值是取自某个已定义字符集的字符串的类型。

### 3.13 布尔类型 boolean type

具有两个可区分值的简单类型。

### 3.14 真 true

布尔类型中的可区分值的一个。

### 3.15 假 false

布尔类型中的可区分值的另一个。

### 3.16 整数类型 integer type

具有可区分值的简单类型,值是正整数或负整数,包括零(作为单一值)。

注:特定的编码规则限定整数的范围,但应使选择的这种范围不致影响ASN.1的任何用户。尽管如此,ASN.1的特定用户可以使用注释表达更严格的限制范围。

### 3.17 枚举类型 enumerated type

一个简单类型,其值是做为类型记法一部分的给定不同标识符。

### 3.18 实数类型 real type

一个简单类型,其可区分值(16.2中规定)是实数集合的一个成员。

### 3.19 位串类型 bitstring type

其可区分值是零个、一个或多个二进制位的有序序列的简单类型。

注:编码规则不限制位串中的位数。但是,ASN.1的特定用户可以使用注释表达位数的上限或下限。

### 3.20 八位位组串类型 octetstring type

其可区分值是零个、一个或多个八位位组的有序序列的简单类型。每个八位位组是八个二进制位的有序序列。

注:编码规则不限制八位位组串中的八位位组数,但ASN.1的特定用户可以使用注释表达八位位组数的上限或下限。

### 3.21 空类型 null type

由一个单一值组成的简单类型,也称为空。

注:空值通常用于有几个可能的候选值但又都不适用的情况。

### 3.22 序列类型 sequence type

通过引用一个固定的有序的类型表(其中某些类型可以申明为可选的)而定义的结构类型;新类型的每个值是一个有序的值表,表中各个值取自相应的成分类型。

注:当一个成分类型申明为可选时,新类型的值不必包括该成分类型的值。

### 3.23 单纯序列类型 sequence-of type

通过引用单个现存类型而定义的结构类型；新类型的每个值是该现存类型的零个、一个或多个值的有序成分表。

注：编码规则中不限制单纯序列值中的值的数目。但是，ASN.1 的特定用户可以使用注释来表达值的数目的上限或下限。

### 3.24 集合类型 set type

通过引用固定的无序的不同类型表(其中某些类型可以申明为可选的)而定义的结构类型；新类型的每个值是一个无序的值表，表中各个值取自相应的成分类型。

注：当一个成分类型说明为可选时，新类型的值不必包括该成分类型的值。

### 3.25 单纯集合类型 set-of type

通过引用单个现存类型定义的结构类型；新类型的每个值是该现存类型的零个、一个或多个值的无序表。

注：编码规则不限制单纯序列类型中的值的数目。但是，ASN.1 的用户可以使用注释表达值的数目的上限或下限，或要求各个值彼此不同。

### 3.26 已标记类型 tagged type

通过引用单个现存类型和一个标记来定义的类型；新类型与该现存类型同构，但并不等同。

### 3.27 值选择类型 choice type

通过引用固定的无序的不同类型而定义的结构类型；新类型的每个值是成分类型之一的值。

### 3.28 类型选择类型 selection type

通过引用值选择类型的一个成分类型而定义的结构类型。

### 3.29 任意类型 any type

未规定其成分类型的值选择类型，但限定其成分类型为能用 ASN.1 定义的类型集合。

### 3.30 外部类型 external type

其可区分值不能从其外部特征导出，但能够从这样一个值的编码导出，这些值可以但不必用 ASN.1 描述，因而其编码可以但不必与 ASN.1 编码规则一致。

### 3.31 信息客体 information object

信息中定义或规范中已定义的部分，为在通信实例中标识其使用而要求有一个名字。

### 3.32 客体标识符 object identifier

一个与信息客体相联系的值(区别于所有其他同类值)。

### 3.33 客体标识符类型 object identifier type

其可区分值为按照本标准规则分配的所有客体标识符的集合。

注：本标准的这条规则允许各种机构独立地将客体标识符与信息客体相联系。

### 3.34 客体描述符类型 object descriptor type

其可区分值是提供对信息客体简要描述的人可读的文本的类型。

注：客体描述符值通常但不总是与某个单一的信息客体相联系。只有客体标识符才无歧义地标识信息客体。

### 3.35 递归定义 recursive definitions

ASN.1 的定义的一个集合，不能对这些定义重新排序，使得构造中使用的所有 ASN.1 类型在该定义构造之前定义。

注：ASN.1 允许递归定义；本记法的用户有责任确保使用的值(或结果类型)具有有限的表示。

### 3.36 模块 module

用于类型定义和值定义的 ASN.1 记法的一个或多个使用实例，用 ASN.1 模块记法(见第 9 章)定界。

### 3.37 产生式 production

用来规定 ASN.1 形式记法的一部分，其中允许的项序列与一个名字相联系，该名字可以用来在新的允许的序列集合定义中引用它所代表的那些序列。

**3.38 国际协调时 Coordinate Universal Time(UTC)**

国际时间局所保持的时标,构成标准频率和时间信号协调传播的基础。

注:① 此定义来源于国际无线电咨询委员会(CCIR)的 160-2 建议。CCIR 也用 UTC 作为世界协调时的缩写。

② UTC 也称为格林尼治标准时间,并定时地广播适当的时间信号。

**3.39 (ASN.1 的)用户 user(of ASN.1)**

使用 ASN.1 定义一段特定信息的抽象语法部分的个人或组织。

**3.40 (双亲类型的)子类型 subtype(of a parent type)**

一个类型,其值是作为一些其他类型(双亲类型)的值的子集合规定的。

**3.41 (子类型的)双亲类型 parent type(of a subtype)**

用于定义子类型的类型。

注:双亲类型自己本身可能是一些其他类型的子类型。

**3.42 子类型规范 subtype specification**

能与一个类型记法一起用于定义此类型的子类型的记法。

**3.43 子类型值集合 subtype value set**

形成子类型规范,规定将包括在子类型中的双亲类型值集合的记法。

本标准使用 GB/T 15695 定义的下列术语:

- a) 表示数据值;
- b) (一个)抽象语法;
- c) 抽象语法名;
- d) 传送语法名。

本标准也使用 GB/T 15696 中定义的下列术语表示上下文标识符。

本标准使用 ISO 6523 中定义的下列术语:

- a) 通行组织;
- b) 组织代码;
- c) 国际代码指示符。

**4 缩写**

ASN.1	抽象语法记法一
UTC	世界协调时
ICD	国际代码指示符
DCC	数据国家代码
DNIC	数据网络识别码
RPOA	认识的私用操作机构

**5 本标准中使用的记法**

ASN.1 记法包含一个字符串序列,这些字符取自第 7 章规定的 ASN.1 字符集。

每次使用 ASN.1 记法包括从 ASN.1 字符集中抽出字符并组合为项。第 8 章规定组成 ASN.1 项的字符串,并给每个项命名。

在第 9 章(以及以下几章)中,ASN.1 记法的规定是通过对组成 ASN.1 表示有效实例的项的序列之集的规定,及对每个序列语义的规定来实现的。

为了说明这样的集合,本标准使用下面各条中定义的形式记法。

**5.1 产生式**

一个新的(或更复杂的)ASN.1 序列的集合是通过产生式来定义的。它使用本标准中定义的序列集

合的名字,并通过以下规定之一来形成新的序列集合:

- a) 新序列集由任何原有集合的任何原有序列组成,或者
- b) 新序列由任何这样的序列组成:它通过从每个集合中仅取一个序列,并按特定的次序组合而成。每个产生式由下面几个部分组成,占一行或几行,次序是:

- a) 新序列集合的名字;
- b) 字符

::=

- c) 一个或多个 5.2 中所定义序列的候选集,使用下面字符分隔

一个序列若在一个或多个候选集中出现,则它在新集中出现。在本标准中,新集合用上面 a) 中的名字引用。

注:若一个序列出现在多个候选中,结果表示中任何语义上的二义性由整个 ASN.1 序列的其他部分解决。

## 5.2 候选集

在“一个或多个候选集”中,序列的每一个候选集通过一组名字来描述。每一个名字或者是一个项名,或者是一个本标准中产生式定义序列集合的名字。

候选集定义的序列集合由所有这样获得的序列组成,取任何一个带第一名字序列(或项),(然后)和任何一个带第二个名字的序列(或项)组合,(然后)和任何一个带第三个名字的序列(或者项)组合,等等,直到包括候选集中最后的一个名字(或者项)。

## 5.3 产生式的例子

位串值 ::= b 串 |  
          h 串 |  
          {标识符表}

是一个产生式,它的名字为“位串值”,序列如下:

- a) 任何 b 串(单项);和
- b) 任何 h 串(单项);和
- c) 任何有标识符表的序列,用{开始,并用}结束。

注:{和}是含有单字符{和}的项名(见第 8 章)。

在这个例子中,“标识符表”由进一步的产生式定义,可以在定义“位串值”的产生式之前或者之后。

## 5.4 格式

在本标准中使用的每个产生式前面和后面都有一个空行,产生式中没有空行。产生式或者在一行上或者分布在几行上。格式是不重要的。

## 5.5 递归

本标准中的产生式通常是递归的。在这种情况下,只要有新的序列产生,产生式就要继续重复。

注:在许多情况下,这样的重复导致了容许序列的无界集,某些或所有这些序列本身就是无界的,这没有错。

## 5.6 序列集合的引用

本标准通过引用产生式的第一个名字(在 ::= 之前)来引用一个序列集合(ASN.1 表示的一部分);这个名字用“括起来,把它和自然语言文本区分开来,否则,它就作为产生式的一部分出现。

## 5.7 项的引用

本标准通过引用项名来定义一个项。项名用“括起来,以与自然语言正文区别开来,否则,它将作为产生式的一部分而出现。

## 5.8 标记

一个标记是通过给出它的类型和类型中的号码来说明的,类型是下列中之一:

通用  
应用

专用  
上下文规定

号码是一个非负整数,用十进制记法。

ASN.1 用户在指派标记时的限制在第 24 章中说明。

通用类型的标记是这样指派的:对结构类型,最高层结构可以由标记导出;对简单类型,类型可由标记导出。表 1 总结了在本标准中说明的通用类型中标记的指派。

表 1 通用类型标记指派

通用类型 1	布尔类型
通用类型 2	整数类型
通用类型 3	位串类型
通用类型 4	八位位串类型
通用类型 5	空类型
通用类型 6	客体标识符类型
通用类型 7	客体描述符类型
通用类型 8	外部类型
通用类型 9	实数类型
通用类型 10	枚举类型
通用类型 11~15	为本标准的将来版本保留
通用类型 16	序列和单纯序列类型
通用类型 17	集合和单纯集合类型
通用类型 18~22,25~27	字符串类型
通用类型 23~24	时间类型
通用类型 28~……	为本标准的将来版本保留

注:通用类型另外的标记保留给本标准以后的增加。

## 6 ASN.1 记法的使用

6.1 类型定义的 ASN.1 表示为“类型”(见 12.1)。

6.2 类型值的 ASN.1 表示为“值”(见 12.6)。

注:在不知道类型的有关内容时,通常不能解释值的记法。

6.3 将类型赋给一个类型引用名的 ASN.1 记法为“类型赋值”(见 11.1)。

6.4 将值赋给一个值引用名的 ASN.1 记法为“值赋值”(见 11.2)。

6.5 “类型赋值”记法和“值赋值”记法仅用于“模块定义”中(见 9.1)。

## 7 ASN.1 字符集

7.1 除 7.2 和 7.3 说明外,一个 ASN.1 项由表 2 中字符的序列组成。

表 2 ASN.1 字符集

A	到	Z
a	到	z
0	到	9
:	=	, { } < .
(	) [ ] - ' "	

注:① 附加字符 > 和 | 在宏定义中使用(见附录 A(补充件))。

② 等价的有关标准由我国的标准化组织给出,附加字符可能在以下的项中出现(最后 5 个在附录 A(补充件)定义):

类型引用 (8.2.1)

标识符 (8.3)

值引用	(8.4)
模块引用	(8.5)
宏引用	(A2.1)
产生式引用	(A2.2)
局部类型引用	(A2.3)
局部值引用	(A2.4)
a 串	(A2.7)

当附加字符用在一种大小写无区别的语言时,由以上某些 ASN.1 项第一个字符的不同情况导致语义的区别将用别的办法处理。

7.2 在用该记法规定字符串类型时,ASN.1 记法中可以出现已定义字符集中的所有字符,括以双引号(见 8.11)。

7.3 其他的字符可以出现于“注释”项中(见 8.6)。

7.4 字符的形式、大小、色彩、亮度及其他显示特性无关紧要。

7.5 大写字母和小写字母是有区别的。

## 8 ASN.1 项

### 8.1 一般规则

8.1.1 下列各条规定 ASN.1 项中的字符,在每种情况下,都给出项名和形成项的字符序列的定义。

注:附录 A(补充件)给出用于宏记法的项。

8.1.2 下列各条规定的每个项要出现在一行上,没有空格(除掉“注释”项和列在表 4 中的以外)。

8.1.3 每行的长度没有限制。

8.1.4 本标准(ASN.1 记法)规定的序列的项可以出现在一行或者几行上,可由一个或多个间隔或空行分隔。

8.1.5 若后续项的起始字符(或多个字符)是前项的末尾字符集包括的容许字符(或多个字符)时,项和后续项间要用空格分隔或写在不同行内。

### 8.2 类型引用

#### 项名 — 类型引用

8.2.1 “类型引用”由任意个(一个或多个)字母、数字或连字符组成。以大写字母开头。不可用连字符结尾。一个连字符不能紧接另一个连字符。

注:有关连字符的规定是为了避免与(可能后随的)注释混淆。

8.2.2 “类型引用”不应是表 3 中列出的保留字符串中之一。

表 3 保留字符序列

BOOLEAN
INTEGER
BIT
STRING
OCTET
NULL
SEQUENCE
OF
SET
IMPLICIT
CHOICE
ANY
EXTERNAL

续表 3

OBJECT
IDENTIFIER
OPTIONAL
DEFAULT
COMPONENTS
TRUE
FALSE
BEGIN
END

注：A2.9 规定了宏定义中的保留字符序列。

### 8.3 标识符

项名 —— 标识符

一个“标识符”由任意个(一个或多个)字母、数字和连字符组成,以小写字母开头,不能以连字符结尾,一个连字符不能紧接另一个连接号。

注:有关连接号的规则是为避免与(可能有的)注释混淆。

### 8.4 值引用

项名 —— 值引用

“值引用”由上述 8.3 中规定为“标识符”的字符序列组成。分析记法的实例时,“值引用”由其出现的上下文与“标识符”区别。

### 8.5 模块引用

项名 —— 模块引用

“模块引用”由 8.2 中规定为“类型引用”的字符序列组成。分析记法的使用实例时,“模块引用”通过出现的上下文与“标识符”区别。

### 8.6 注释

项名 —— 注释

8.6.1 ASN.1 记法的定义中不引用“注释”。但它可以在任意时候出现在别的 ASN.1 项中,无关紧要。

8.6.2 “注释”是从两个相连的连字符开始并以下一对相连的连字符或行尾为结束,不论哪一个在前面。注释除了开始的一对连字符及结尾的一对连字符(若有的话)外,不能有两个相连连字符。注释中可以包括不在 7.1 字符集中的字符(见 7.3)。

### 8.7 空项

项名 —— 空

“空”不包含任何字符,该记法第五章中用它来规定候选序列集,以指示所有的候选都缺省是可能的。

### 8.8 数项

项名 —— 数

“数”是由一个或多个数字组成。除非“数”是单个数字,否则其第一位数字不能是 0。

### 8.9 二进制数串项

项名 —— b 串

“b 串”由任意个 0 和 1 组成。在它之前放置‘,后随一对字符’B。

例:‘01101100’B。

### 8.10 十六进制数串项

项名 —— h 串

8.10.1 “h 串”由下列字符中任意个组成(可以零个)。

A B C D E F 0 1 2 3 4 5 6 7 8 9

前置单个‘,后随’H。

例:‘ABC196’H

8.10.2 代表十六进制的每个字符表示 4 位的值。

### 8.11 字符串项

项名 —— c 串

“c 串”由任意个(可以为零个)字符组成,前后置”,这些字符来自某个字符集类型所用的字符集。若字符集中包含”,则该字符要由一对”来表示,采用的字符不限于表 2 列出的字符,而由以 c 串为值(见 7.2)的类型决定。

例:“宛姐蛇怡”

### 8.12 赋值项

项名 —— “::=”

该项由序列::=组成。

注:该项不包含任何空格字符。(见 8.1.2)

### 8.13 单字符项

项名 —— {  
 }  
 <  
 ,  
 .  
 (  
 )  
 [  
 ]  
 —(连字符)  
 ;”

具有上述列出的任意一个名字项由形成名字的单字符组成。

注:①项“1”在 A2.5 中定义。

②项>在 A2.6 中定义。

### 8.14 关键字项

项名 —— BOOLEAN  
 INTEGER  
 BIT  
 STRING  
 OCTET  
 NULL  
 SEQUENCE  
 OF  
 SET  
 IMPLICIT  
 CHOICE  
 ANY  
 EXTERNAL

OBJECT  
IDENTIFIER  
OPTIONAL  
DEFAULT  
COMPONENTS  
UNIVERSAL  
APPLICATION  
PRIVATE  
TRUE  
FALSE  
BEGIN  
END  
DEFINITIONS  
EXPLICIT  
ENUMERATED  
EXPORTS  
IMPORTS  
REAL  
INCLUDES  
MIN  
MAX  
SIZE  
FROM  
WITH  
COMPONENT  
PRESENT  
ABSENT  
DEFINE  
BY  
PLUS—INFINITY  
MINUS--INFINITY  
TAGS

有上述名字的项应由名字中字符序列组成。

注：① 这些序列中无空格。

② 对以上字符序列中不在 8.2.2 中的保留字符序列中列出的几个，它们由上下文区别于包含同样字符的项。

#### 8.15 变化范围分隔符

项名 — ..

此项应由字符序列组成。

注：此序列不包含任何空格符。

#### 8.16 省略号

项名 ...

此项应由字符序列组成。

注：此序列不包含任何空格符。

## 9 模块定义

### 9.1 “模块定义”由下列产生式规定

模块定义 ::= 模块标识符

DEFINITIONS

标记缺省

“::=”

BEGIN

模块体

END

模块标识符 ::= 模块引用

已赋值标识符

已赋值标识符 ::= 客体标识符值 |

空值

标记缺省 ::= EXPLICIT TAGS |

IMPLICIT TAGS |

空值

模块体 ::= 外部 内部 赋值表 | 空值

外部 ::= EXPORTS 外部符号; | 空值

外部符号 ::= 符号表 | 空值

内部 ::= IMPORTS 内部符号; | 空值

外部符号 ::= 来自模块表的符号 | 空值

来自模块表的符号 ::= 来自模块的符号 |

来自模块表的符号 来自模块的符号

来自模块表的符号 ::= 符号表 FROM 模块标识符

符号表 ::= 符号 | 符号表, 符号

符号 ::= 类型引用 | 值引用

赋值表 ::= 赋值 | 赋值表 赋值

赋值 ::= 类型赋值 | 值赋值

注：① 附录 A(补充件)规定一个也可以出现在“赋值表”中的“宏定义”序列。由宏定义产生的记法可以在同一模块中的其他地方出现。

② 在个别情况下(但不赞成这样),对于例子和通用类标记的定义,“模块体”可以在“模块定义”之外使用。

③ “类型定义”和“值赋值”产生式在第 11 章中规定。

④ 将 ASN.1 数据类型组成模块时,不必为表示上下文定义将表示数据信息变为命名抽象语法。

⑤ 模块定义的“标记缺省”值只影响那些在模块中显式定义的类型。它并不影响内部类型的表示。

⑥ 一个“宏引用”(见附录 A)也可作为一个“符号”。

### 9.2 如果“标记缺省”是“空值”,它取“EXPLICIT TAGS”。

9.3 出现在“模块定义”产生式中的“模块引用”称为模块名。模块名的选择用来保证在具有这个模块名的所有“模块定义”序列的“模块体”中出现的所有序列的相容性和完整性。如果对其中出现的每个“类型引用”和“值引用”,只有一个“类型赋值”或“值赋值”(相应的)与类型或值的名字(相应的)相关连,则“赋值”序列的集合是相容和完整的。

### 9.4 模块名在模块定义的应用范围内只能使用一次(除 9.4 中说明以外)。

注：建议在标准中定义的模块有如下形式的模块名:ISO XXXX-YYYY 其中 XXXX 是标准序号,YYYY 是标准内的首字母缩写(如:JTM,FTEM 或 CCR)。在其他标准中可使用类似的约定。