版本信息

日期	版本	描述	作者
2004-02-16	v1.0	最初版本	LDAPChina.com

Attribute Syntax Definitions
属性语法定义

1、本备忘录的状态（Status of this Memo）

本文档定义了一个用于 Internet 通讯的 Internet 标准跟踪协议，为了发展的需要讨论和建议。对于这个协议的状况和地位请参照 Internet 官方协议标准（"Internet Official Protocol Standards"（STD 1））的当前版。这个备忘录的传播是不受限制的。

版权提示（Copyright Notice）

IESG 提示（IESG Note）

本文档描述将一种同时提供读和更新访问的目录访问协议。更新访问需要安全认证，但这个文档并不强制实现任何安全认证机制。

与 RFC2026 的 4.4.1 节相同，本规范正在被 IESG 批准期间，作为被提议的标准，可能并不限于本文档所述内容。原因如下：

a、 鼓励在它们被发布前，实现和交互测试这些协议（带有或没有更新访问）；

b、 鼓励在只读的应用程序中配置和使用这些协议。（例如，在某些应用程序中，目录的更新访问使用某些其它安全的机制而不是 LDAP，而使用 LDAPv3 被作为对目录的查询语言）；

c、 避免阻碍别的 Internet 标准追踪协议的发展和发布。（这些协议需要 LDAPv3 的目录服务器的查询能力，而不是更新能力）

需要警告读者的是，直到强制的验证机制被标准化之前，根据本规范编写的客户端和服务器实现了更新功能的话，它们的互操作性可能是不可靠的，或者仅提供在认证需要极度弱化的时候的互操作性。

因此在具有强制认证的 LDAPv3 未成为一个 RFC 而被批准或发布之前，不鼓励实现者发布一个实现了更新功能的 LDAPv3 的客户端和服务器。

目 录

2、摘要（Abstract）

LDAP 协议（参考文档[1]）需要协议元素中 AttributeValue 域的内容为八进制的字符串。本文档定义了 LDAPv3 的一套语法，以及这些语法的属性值在 LDAP 协议的传输中的表示规则。在本文档中定义的语法被定义的属性类型的文档（包括本文档和其它文档）所遵从。本文档也定义了 LDAP 服务器应当支持的一套属性类型。

3、综述（Overview）

本文档为使用 LDAP 进行访问的目录定义了开发模式（schema）的框架。

模式（schema）是由下列内容组成的集合：

1、 属性类型定义；

2、 对象类（object class）定义；

3、 其它信息，服务器可以使用它来决定一个条目的属性如何匹配一个过滤和属性值判定（attribute value assertion）（在比较操作中），以及决定是否允许增加和修改操作。

第4节为属性类型、对象类、语法和匹配规则定义描述了总体需求和符号标记（notation）。

第 5 节是属性列表，第六节是语法列表，第七节是对象类列表。

其它文档定义了用目录条目表示现实世界对象的模式（schema）。

4、总体问题（General Issues）

本文档描述用于 Internet 协议的编码。

本文档中的关键字"**MUST**"，"**MUST NOT**"，"**REQUIRED**"，"**SHALL**"，"**SHALL NOT**"，"**SHOULD**"，"**SHOULD NOT**"，"**RECOMMENDED**"和"**MAY**"的含意与 RFC 2119（参考文档[4]）中描述的相同。

属性类型和对象类定义书写为 AttributeTypeDescription 和 ObjectClassDescription 数据类型的字符串表达方式，AttributeTypeDescription 和 ObjectClassDescription 在 X.501（93）（参考文档[3]）中定义。强烈建议实现者在读本文档的剩余部分之前，首先阅读 X.500 的模式（shcema）描述表示法。

4.1、公共编码方面（Common Encoding Aspects）

定义属性语法的编码规则需要使用下面的 BNF 定义。它们基于 RFC 822BNF（参考文档[13]）样式（style）。

```
a        = "a" / "b" / "c" / "d" / "e" / "f" / "g" / "h" / "i" /
           "j" / "k" / "l" / "m" / "n" / "o" / "p" / "q" / "r" /
           "s" / "t" / "u" / "v" / "w" / "x" / "y" / "z" / "A" /
           "B" / "C" / "D" / "E" / "F" / "G" / "H" / "I" / "J" /
           "K" / "L" / "M" / "N" / "O" / "P" / "Q" / "R" / "S" /
           "T" / "U" / "V" / "W" / "X" / "Y" / "Z"

d                    = "0" / "1" / "2" / "3" / "4" /
                       "5" / "6" / "7" / "8" / "9"

hex-digit        =    d / "a" / "b" / "c" / "d" / "e" / "f" /
                           "A" / "B" / "C" / "D" / "E" / "F"

k                    = a / d / "-" / ";"

p                    = a / d / "'" / "(" / ")" / "+" / "," /
                       "-" / "." / "/" / ":" / "?" / " "

letterstring     = 1*a

numericstring    = 1*d

anhstring        = 1*k

keystring        = a [ anhstring ]

printablestring = 1*p

space            = 1*" "

whsp             = [ space ]

utf8             = <any sequence of octets formed from the UTF-8 [9]
                       transformation of a character from ISO10646 [10]>

dstring          = 1*utf8

qdstring         = whsp "'" dstring "'" whsp
```

```
qdstringlist      = [ qdstring *( qdstring ) ]

qdstrings         = qdstring / ( whsp "(" qdstringlist ")" whsp )
```

下列对象标识符（OBJECT IDENTIFIER）的字符串表示的 BNF 中，descr 是对象描述符（object descriptor）的语法表示，descr 由字母和数字组成，以字母开头。一个 numericoid 格式的对象标识符开头不应该有多个 0。（例如 "0.9.3" 被允许但"0.09.3"不应该被产生）。

当在某个值里对'oid'元素编码时，descr 编码选项**应该**（SHOULD）比 numericoid 优先使用。对于一个数字对象表识符（OBJECT IDENTIFIER）来说，对象描述符是一个更可读的别名，并且这些对象标识符（被实现所分配和理解）**应该**（SHOULD）比 numericoid 更优先地使用，以得到最大的扩展（extent）可能性。在 LDAP 中的对象描述符例子是属性类型，对象类和匹配规则的名字。LDAP 中，对象描述符的例子是属性类型，对象类和匹配规则名（matching rule name）。

```
oid               = descr / numericoid

descr             = keystring

numericoid        = numericstring *( "." numericstring )

woid              = whsp oid whsp

; set of oids of either form
oids              = woid / ( "(" oidlist ")" )

oidlist           = woid *( "$" woid )

; object descriptors used as schema element names
qdescrs           = qdescr / ( whsp "(" qdescrlist ")" whsp )

qdescrlist        = [ qdescr *( qdescr ) ]

qdescr            = whsp "'" descr "'" whsp
```

4.2、属性类型（Attribute Types）

为描述子模式（subschema）"attributeTypes"属性，下面范例值展示了属性类型，它使用 AttributeTypeDescription 语法书写。然而，这种折行显示的方式只是为了使用它具有可读性，当这些值在协议中传输时将不包含折行。

AttributeTypeDescription 按照下面的 BNF 进行编码，其中 oid，qdescrs 和 qdstring 的说明已经在 4.1 节中给出。实现者应该注意在本文档未来的版本中可能会扩展该 BNF 以包含附加的术语。以字符"X-"开头的术语将为用户保留，并且它**必须**（MUST）紧跟一个<qdstrings>。

```
AttributeTypeDescription = "(" whsp
      numericoid whsp                  ; AttributeType identifier
    [ "NAME" qdescrs ]                 ; name used in AttributeType
    [ "DESC" qdstring ]                ; description
    [ "OBSOLETE" whsp ]
    [ "SUP" woid ]                     ; derived from this other
                                       ; AttributeType
    [ "EQUALITY" woid                  ; Matching Rule name
    [ "ORDERING" woid                  ; Matching Rule name
    [ "SUBSTR" woid ]                  ; Matching Rule name
    [ "SYNTAX" whsp noidlen whsp ] ; see section 4.3
    [ "SINGLE-VALUE" whsp ]            ; default multi-valued
    [ "COLLECTIVE" whsp ]              ; default not collective
    [ "NO-USER-MODIFICATION" whsp ]; default user modifiable
    [ "USAGE" whsp AttributeUsage ]; default userApplications
    whsp ")"

AttributeUsage =
    "userApplications"     /
    "directoryOperation"   /
    "distributedOperation" / ; DSA-shared
    "dSAOperation"           ; DSA-specific, value depends on server
```

服务器不需要在它们维护的子模式（subschema）值的描述部分提供相同的文本或其它任意的文本。服务器**应该**（SHOULD）为每一个 AttributeTypeDescription 提供至少"SUP"字段和"SYNTAX"字段中的一个。

服务器**必须**（MUST）实现在 5.1 节、5.2 节和 5.3 节中引用的所有属性类型。

服务器**可以**（MAY）识别未列在本文档中的其它命名和属性，并且如果它们这样做了，则**必须**（MUST）在它们的子模式（subschema）条目的 attributeTypes 属性中发布该类型的定义。

Schema 开发者**绝不**（MUST NOT）能创建与目前 LDAP 标准追踪 RFC（standards-track RFCs）中定义的属性名称相冲突的属性。

一个 AttributeDescription 可以被用来作为 AttributeTypeDescription 的 NAME 部分的值。应注意它们是区分大小写的。

注意 AttributeTypeDescription 没有列出可以在扩展匹配（extensibleMatch）查询过滤器中与属性类型一同使用的匹配规则。这方面内容将在 4.5 节中的 matchingRuleUse 属性中描述。

本文档细化了 X.501 中的模式（schema）描述，本文档要求为了 LDAP 字符串语法定义，AttributeTypeDescription 中的语法域（syntax field）是一个对象标识符（OBJECT IDENTIFIER）的字符串表达，同时也是该属性的某个值的最大长度的可选指示标志（在 4.3.2 节中定义）。

4.3、语法（Syntaxes）

本节为 LDAP 属性值语法编码定义了总体要求。其它所有文档定义的与 LDAP 相关的属性语法编码都应该与这些要求一致。

为一个给定的属性语法定义的编码规则必须产生八进制字符串。为了达到最大的扩展可能性，已编码的八进制字符串为了显示的目的，应该使用本地语言编码形式。特别需要指出的是，为属性语法定义的非二进制值的编码规则产生的字符串，在 LDAP 客户端显示时应该做到尽量少的翻译或根本不需要翻译。然而，有极少数的情况（例如，音频），某些属性语法并不能产生可打印的表达形式，客户端**绝不**（MUST NOT）能主观地认为某个不识别的语法是一个字符串表达形式。

当对任意字符串（该字符串不是一个 DN，而是被用当作一个更大的生成物的一部分，并且也不是一个 DN 的一部分）进行编码时，反斜线（\）符号机制被用于对后面的单个字符（如"\"，"$"或"#"）进行转义（escape）。反斜线后面紧跟一对 16 进制数字，该数字代表下一个字符。字符串中的一个反斜线字符本身总是使用'\5C'和'\5c'来代表。在 6.27 节中将给出一个例子。

对于判定值语法与属性值语法不同的配匹规则，语法也对它们进行了定义。

4.3.1、值的二进制传换（Binary Transfer of Values）

如果客户端需要对一个属性进行二进制编码，或者如果属性语法名是"1.3.6.1.4.1.1466.115.121.1.5"，则使用该编码规则。LDAP 的 AttributeValue 域（field）或 AssertionValue 域的内容或者是该属性值的 BER 编码实例，或者是匹配规则判定值 ASN.1 数据类型（用于与 X.500 一起使用）。（八进制字符串封装（OCTET STRING wrapper）内部的第一个字节是一个标签字节（tag octet）。然而，该八进制字符串仍编码成原始形式。）

如果属性类型可被识别并且属性语法名是二进制（Binary）的属性语法名时，所有的服

务器在查询响应，以及在增加、比较和修改请求中解析属性值时，**必须**（MUST）实现这种语法形式。请求从条目中返回所有属性的客户端，**必须**（MUST）准备以二进制形式接收值（例如下列属性：userCertificate，binary），并且**不应该**（SHOULD NOT）只简单地给用户显示二进制值或不识别的值。

4.3.2、语法对象标识符（Syntax Object Identifiers）

用于 LDAP 的语法通过对象标识符（OBJECT IDENTIFIER）命名，它是点-数（dotted-decimal）字符串。不要将它们显示给用户。

noidlen = numericoid ["{" len "}"]

len = numericstring

下表列出了一些迄今为止已经为 LDAP 定义了的语法。H-R 列指出在那个语法中的值是否是人类可识别字符串。客户和服务器不需要实现所有列出的语法，并且**可以**（MAY）实现其它的语法。

其它文档定义可以定义更多的语法。然而，强烈地反对定义任意其它的语法，因为它对互操作性起负面作用。当今的客户端和服务端通常不具备动态识别新语法的能力。绝大多数情况下，属性将使用为目录字符串而定义得语法。

Value being represented	H-R	OBJECT IDENTIFIER
ACI Item	N	1.3.6.1.4.1.1466.115.121.1.1
Access Point	Y	1.3.6.1.4.1.1466.115.121.1.2
Attribute Type Description	Y	1.3.6.1.4.1.1466.115.121.1.3
Audio	N	1.3.6.1.4.1.1466.115.121.1.4
Binary	N	1.3.6.1.4.1.1466.115.121.1.5
Bit String	Y	1.3.6.1.4.1.1466.115.121.1.6
Boolean	Y	1.3.6.1.4.1.1466.115.121.1.7
Certificate	N	1.3.6.1.4.1.1466.115.121.1.8
Certificate List	N	1.3.6.1.4.1.1466.115.121.1.9
Certificate Pair	N	1.3.6.1.4.1.1466.115.121.1.10
Country String	Y	1.3.6.1.4.1.1466.115.121.1.11
DN	Y	1.3.6.1.4.1.1466.115.121.1.12
Data Quality Syntax	Y	1.3.6.1.4.1.1466.115.121.1.13
Delivery Method	Y	1.3.6.1.4.1.1466.115.121.1.14
Directory String	Y	1.3.6.1.4.1.1466.115.121.1.15
DIT Content Rule Description	Y	1.3.6.1.4.1.1466.115.121.1.16
DIT Structure Rule Description	Y	1.3.6.1.4.1.1466.115.121.1.17
DL Submit Permission	Y	1.3.6.1.4.1.1466.115.121.1.18

DSA Quality Syntax	Y	1.3.6.1.4.1.1466.115.121.1.19
DSE Type	Y	1.3.6.1.4.1.1466.115.121.1.20
Enhanced Guide	Y	1.3.6.1.4.1.1466.115.121.1.21
Facsimile Telephone Number	Y	1.3.6.1.4.1.1466.115.121.1.22
Fax	N	1.3.6.1.4.1.1466.115.121.1.23
Generalized Time	Y	1.3.6.1.4.1.1466.115.121.1.24
Guide	Y	1.3.6.1.4.1.1466.115.121.1.25
IA5 String	Y	1.3.6.1.4.1.1466.115.121.1.26
INTEGER	Y	1.3.6.1.4.1.1466.115.121.1.27
JPEG	N	1.3.6.1.4.1.1466.115.121.1.28
LDAP Syntax Description	Y	1.3.6.1.4.1.1466.115.121.1.54
LDAP Schema Definition	Y	1.3.6.1.4.1.1466.115.121.1.56
LDAP Schema Description	Y	1.3.6.1.4.1.1466.115.121.1.57
Master And Shadow Access Points	Y	1.3.6.1.4.1.1466.115.121.1.29
Matching Rule Description	Y	1.3.6.1.4.1.1466.115.121.1.30
Matching Rule Use Description	Y	1.3.6.1.4.1.1466.115.121.1.31
Mail Preference	Y	1.3.6.1.4.1.1466.115.121.1.32
MHS OR Address	Y	1.3.6.1.4.1.1466.115.121.1.33
Modify Rights	Y	1.3.6.1.4.1.1466.115.121.1.55
Name And Optional UID	Y	1.3.6.1.4.1.1466.115.121.1.34
Name Form Description	Y	1.3.6.1.4.1.1466.115.121.1.35
Numeric String	Y	1.3.6.1.4.1.1466.115.121.1.36
Object Class Description	Y	1.3.6.1.4.1.1466.115.121.1.37
Octet String	Y	1.3.6.1.4.1.1466.115.121.1.40
OID	Y	1.3.6.1.4.1.1466.115.121.1.38
Other Mailbox	Y	1.3.6.1.4.1.1466.115.121.1.39
Postal Address	Y	1.3.6.1.4.1.1466.115.121.1.41
Protocol Information	Y	1.3.6.1.4.1.1466.115.121.1.42
Presentation Address	Y	1.3.6.1.4.1.1466.115.121.1.43
Printable String	Y	1.3.6.1.4.1.1466.115.121.1.44
Substring Assertion	Y	1.3.6.1.4.1.1466.115.121.1.58
Subtree Specification	Y	1.3.6.1.4.1.1466.115.121.1.45
Supplier Information	Y	1.3.6.1.4.1.1466.115.121.1.46
Supplier Or Consumer	Y	1.3.6.1.4.1.1466.115.121.1.47
Supplier And Consumer	Y	1.3.6.1.4.1.1466.115.121.1.48
Supported Algorithm	N	1.3.6.1.4.1.1466.115.121.1.49
Telephone Number	Y	1.3.6.1.4.1.1466.115.121.1.50
Teletex Terminal Identifier	Y	1.3.6.1.4.1.1466.115.121.1.51
Telex Number	Y	1.3.6.1.4.1.1466.115.121.1.52
UTC Time	Y	1.3.6.1.4.1.1466.115.121.1.53

一个基于字符串的语法的值的最大字符数上限，或者所有其它语法的值的最大字节数的上限，都可以通过下面的方式标明：即在属性类型描述中，在语法名的对象标识符（OBJECT

IDENTIFIER）后加上大花括号，并在大花括号中标明边界数字。例如，"1.3.6.4.1.1466.0{64}"指出服务器应该允许一个最长为 64 个字符的字符串，虽然该服务器可以允许更长的字符串。注意在目录字符串语法中的单个字符，可能在编码后超过一个字节，这是由于 UTF-8 是变长编码（variable-length encoding）方式。

4.3.3、语法描述（Syntax Description）

下面的 BNF 可以用来将一个短的语法描述与语法对象标识符联系起来。实现者应当注意本文档未来的版本中可能会扩充该定义以包含附加的术语。标识符以"X-"开头的术语保留给个人实验用，并且后面**必须**（MUST）紧跟有<qdstrings>。

```
SyntaxDescription = "(" whsp
    numericoid whsp
    [ "DESC" qdstring ]
    whsp ")"
```

4.4、对象类（Object Classes）

表达对象类的格式在 X.501（参考文档[3]）中定义。在通常情况下，每个条目将包含下列内容：一个抽象类（abstract class）（"top"或"alias"）、至少一个结构对象类（structural object class）、以及零个或多个辅助对象类（auxiliary object class）。当给一个对象类标识符赋值时（译者注：即新建一个对象类时），该对象类是抽象对象类、还是结构对象类、还是属于辅助将被定义。在没有给某个对象类赋予一个新标识符时，该对象类定义不应该被更改。

对象类描述根据下面的 BNF 书写。实现者应该注意，本文档未来的版本中可能扩展该定义来包含附加的术语。标识符以"X-"开头的术语保留给个人实验用，并且后面**必须**（MUST）紧跟有<qdstrings>。

```
ObjectClassDescription = "(" whsp
    numericoid whsp          ; ObjectClass identifier
    [ "NAME" qdescrs ]
    [ "DESC" qdstring ]
    [ "OBSOLETE" whsp ]
    [ "SUP" oids ]           ; Superior ObjectClasses
    [ ( "ABSTRACT" / "STRUCTURAL" / "AUXILIARY" ) whsp ]
                             ; default structural
    [ "MUST" oids ]          ; AttributeTypes
    [ "MAY" oids ]           ; AttributeTypes
whsp ")"
```

上面的范例值为实现了 LDAP 模式（schema）的服务器描述子模式（subschema）

"objectClasses"属性。然而，这种折行显示的方式只是为了使用它具有可读性，当这些值在协议中传输时将不包含折行。

服务器**应该**（SHOULD）实现第 7 节涉及到的所有对象类，除了可选的对象类 extensibleObject。服务器**可以**（MAY）实现未在本文档列出的更多的对象类。如果服务器这样做的话，则**必须**（MUST）将这些类定义公布在子模式（subschema）条目的 objectClasses 属性中。

模式（schema）开发者**绝不**（MUST NOT）能创建与现有标准追踪 RFC 文档中定义的对象类名称冲突的对象类。

4.5、匹配规则（Matching Rules）

在查询和比较操作时，服务器使用匹配规则来比较属性值和判定值（assertion value）。在修改条目时，匹配规则也用于识别要被修改或要被删除的值，并且也用于条目名称与期望 DN 进行比较。

在本文档中定义的绝大多数属性已经定义了相等匹配规则（equality matching rule）。

匹配规则描述根据下面的 BNF 书写。实现者应该注意，本文档未来的版本中可能扩展该 BNF 以包含附加的术语。标识符以"X-"开头的术语保留给个人实验用，并且后面**必须**（MUST）紧跟有<qdstrings>。

```
MatchingRuleDescription = "(" whsp
      numericoid whsp    ; MatchingRule identifier
      [ "NAME" qdescrs ]
      [ "DESC" qdstring ]
      [ "OBSOLETE" whsp ]
      "SYNTAX" numericoid
whsp ")"
```

MatchingRuleUse 的值列出了适合与扩展匹配规则一起使用的属性。

支持匹配规则和 extensible

```
MatchingRuleUseDescription = "(" whsp
      numericoid whsp    ; MatchingRule identifier
      [ "NAME" qdescrs ]
      [ "DESC" qdstring ]
      [ "OBSOLETE" ]
     "APPLIES" oids      ; AttributeType identifiers
whsp ")"
```

Match 的服务器**应该**（SHOULD）实现第 8 节中描述的所有匹配规则。

服务器**可以**（MAY）实现未在本文档列出的的更多匹配规则，如果服务器这样做的话，

则**必须**（MUST）将这些匹配规则定义公布在子模式（subschema）条目的 matchingRules 属性中。如果服务器支持 extensibleMatch，则该服务器**必须**（MUST）在 matchingRuleUse 中公布这种匹配规则和属性之间的关系。

例如，某个服务器在字符值（String-valued）属性上实现自定义的类似匹配（sound-alike matches），该服务器在子模式（subschema）条目中将包含如下信息：（1.2.3.4.5 仅是一个例子，实际的匹配规则的 OID 将与其不同）：

matchingRule: (1.2.3.4.5 NAME 'soundAlikeMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15)

如果该匹配规则能与属性 2.5.4.41 和 2.5.4.15 一起使用，将出现如下信息：

matchingRuleUse: (1.2.3.4.5 APPLIES (2.5.4.41 $ 2.5.4.15))

那么，客户端可以发送一个使用该匹配规则的查询操作，在该查询操作中，过滤器使用 extensibleMatch 选项，matchingRule 域是"soundAlikeMatch"，并且 type 域是"2.5.4.41"和"2.5.4.15"。

5、属性类型（Attribute Types）

所有 LDAP 服务器**必须**（MUST）可以识别在本节定义的所有属性类型。

服务器也**应该**（SHOULD）可以识别所有的来自参考文档[12]中第 5 节的属性类型。

5.1、标准操作性属性（Standard Operational Attributes）

服务器**必须**（MUST）维护这些属性的值，以保持与 X.501(93)中定义的一致。

5.1.1、createTimestamp

该属性**应该**（SHOULD）出现在使用 Add 操作创建的条目中。

(2.5.18.1 NAME 'createTimestamp' EQUALITY generalizedTimeMatch
ORDERING generalizedTimeOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE NO-USER-MODIFICATION USAGE directoryOperation)

5.1.2、modifyTimestamp

该属性**应该**（SHOULD）出现在使用 Modify 操作修改过的条目中。

(2.5.18.2 NAME 'modifyTimestamp' EQUALITY generalizedTimeMatch
ORDERING generalizedTimeOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE NO-USER-MODIFICATION USAGE directoryOperation)

5.1.3、creatorsName

该属性**应该**（SHOULD）出现在使用 Add 操作创建的条目中。

(2.5.18.3 NAME 'creatorsName' EQUALITY distinguishedNameMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE NO-USER-MODIFICATION USAGE directoryOperation)

5.1.4、modifiersName

该属性**应该**（SHOULD）出现在使用 Modify 操作修改过的条目中。

(2.5.18.4 NAME 'modifiersName' EQUALITY distinguishedNameMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE NO-USER-MODIFICATION USAGE directoryOperation)

5.1.5、subschemaSubentry

该属性的值是一个子模式条目（subschema entry）（如果服务器是基于 X.500(93)则是子条目（subentry））的名称，在子模式条目中服务器允许有效的属性指定模式（schema）。

(2.5.18.10 NAME 'subschemaSubentry'
EQUALITY distinguishedNameMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 NO-USER-MODIFICATION
SINGLE-VALUE USAGE directoryOperation)

5.1.6、attributeTypes

该属性一般位于子模式（subschema）条目中。

(2.5.21.5 NAME 'attributeTypes'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.3 USAGE directoryOperation)

5.1.7、objectClasses

该属性一般位于子模式（subschema）条目中。

(2.5.21.6 NAME 'objectClasses'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.37 USAGE directoryOperation)

5.1.8、matchingRules

该属性一般位于子模式（subschema）条目中。

(2.5.21.4 NAME 'matchingRules'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.30 USAGE directoryOperation)

5.1.9、matchingRuleUse

该属性一般位于子模式（subschema）条目中。

(2.5.21.8 NAME 'matchingRuleUse'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.31 USAGE directoryOperation)

5.2、LDAP 操作性属性（LDAP Operational Attributes）

这些属性仅出现在根 DSE 中。（参见参考文档[1]和[3]）。

服务器**必须**（MUST）识别这些属性名称，但是当属性对应的功能服务器没有实现时，服务器可以不必为这些属性提供值。

5.2.1、namingContexts

该属性的值反映服务器所掌管的或映像（shadow）得命名上下文（naming context：或称为命名空间）。如果服务器不掌管任何信息，（例如该服务器是一个连接到 X.500 目录的 LDAP 网关）不填写该属性。如果服务器相信它掌管整个目录，该属性将是单值的，并且该值是空字符串（声明根（root）是空的 DN）。该属性允许一个客户端在连接服务器后，选择合适的基准对象（base object）来进行查询。

(1.3.6.1.4.1.1466.101.120.5 NAME 'namingContexts'

SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 USAGE dSAOperation)

5.2.2、altServer

该属性的值是其它服务器的 URL，这些其它的服务器在该服务器不可用时，允许客户端连接它们。如果该服务器不知道其它服务器是否可以使用，不填写该属性。万一客户端连接的 LDAP 服务器变得不可使用，客户端可以缓存该信息。

(1.3.6.1.4.1.1466.101.120.6 NAME 'altServer'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 USAGE dSAOperation)

5.2.3、supportedExtension

该属性的值是标识服务器已支持的扩展操作的对象标识符（OBJECT IDENTIFIER）。

如果服务器不支持任何扩展，不填写该属性。

(1.3.6.1.4.1.1466.101.120.7 NAME 'supportedExtension'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 USAGE dSAOperation)

5.2.4、supportedControl

该属性的值是标识服务器支持的控制的对象标识符（OBJECT IDENTIFIER）。如果服务器不支持任何控制，不填写该属性。

(1.3.6.1.4.1.1466.101.120.13 NAME 'supportedControl'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 USAGE dSAOperation)

5.2.5、supportedSASLMechanisms

该属性的值是服务器支持的 SASL 安全机制的名称。果如服务器不支持任何安全机制，不填写该属性。

(1.3.6.1.4.1.1466.101.120.14 NAME 'supportedSASLMechanisms'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 USAGE dSAOperation)

5.2.6、supportedLDAPVersion

该属性的值是服务器实现的 LDAP 协议的版本。

(1.3.6.1.4.1.1466.101.120.15 NAME 'supportedLDAPVersion'

SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 USAGE dSAOperation)

5.3、LDAP 子模式属性（LDAP Subschema Attribute）

该属性一般位于子模式（subschema）条目中。

5.3.1、ldapSyntaxes

服务器**可以**（MAY）使用该属性来列出已实现了的语法。每个值对应一个语法。

(1.3.6.1.4.1.1466.101.120.16 NAME 'ldapSyntaxes'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.54 USAGE directoryOperation)

5.4、X.500 子模式属性（X.500 Subschema attributes）

这些属性位于子模式（subschema）条目中。虽然一般仅 X.500 服务器才实现它们的功能，但所有服务器都**应该**（SHOULD）识别它们的名称。

5.4.1、dITStructureRules

(2.5.21.1 NAME 'dITStructureRules' EQUALITY integerFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.17 USAGE directoryOperation)

5.4.2、nameForms

(2.5.21.7 NAME 'nameForms'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.35 USAGE directoryOperation)

5.4.3、ditContentRules

(2.5.21.2 NAME 'dITContentRules'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.16 USAGE directoryOperation)

6、语法（Syntaxes）

服务器**应该**（SHOULD）识别所有的在本节描述的语法。

6.1、属性类型描述（Attribute Type Description）

(1.3.6.1.4.1.1466.115.121.1.3 DESC 'Attribute Type Description')

该语法中的值根据 4.2 节中的开始部分描述的 BNF 进行编码。例如：

```
( 2.5.4.0 NAME 'objectClass'
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )
```

6.2、二进行（Binary）

(1.3.6.1.4.1.1466.115.121.1.5 DESC 'Binary')

该语法中的值以 4.3.1 节中描述的方式进行编码。

6.3、二进制字符串（Bit String）

(1.3.6.1.4.1.1466.115.121.1.6 DESC 'Bit String')

该语法中的值应当根据下面的 BNF 进行编码：

```
bitstring = "'" *binary-digit "'B"

binary-digit = "0" / "1"
```

例如：

```
'0101111101'B
```

6.4、布尔（Boolean）

(1.3.6.1.4.1.1466.115.121.1.7 DESC 'Boolean')

该语法中的值应当根据下面的 BNF 进行编码：

```
boolean = "TRUE" / "FALSE"
```

如果逻辑上是真，则 Boolean 值的编码是"TRUE"，否则是"FALSE"。

6.5、证书（Certificate）

(1.3.6.1.4.1.1466.115.121.1.8 DESC 'Certificate')

由于 X.509(1988)和 X.509(1993)的改变，以及为了支持证书的扩展，ASN.1 定义而产生的附加改变，所以没有定义字符串表达方式（译者注：即该属性不能为字符串），并且该语法中的值**必须**（MUST）仅能使用二进制编码进行传递，在进行传递时，使用"userCertificate;binary"或"caCertificate;binary"属性进行请求和返回。在 RFC 1778 中的"User Certificate"的 BNF 符号不被建议使用。

6.6、证书列表（Certificate List）

(1.3.6.1.4.1.1466.115.121.1.9 DESC 'Certificate List')

由于 X.509(1988)和 X.509(1993)的撤销（或称吊销）列表（revocation list）定义不兼容，所以该语法中的值**必须**（MUST）仅能使用二进制编码进行传递，在进行传递时，使用"certificateRevocationList;binary"或"authorityRevocationList;binary"属性请求或返回。在 RFC 1778 中的"Authority Revocation List"的 BNF 符号不被建议使用。

6.7、证书对（Certificate Pair）

(1.3.6.1.4.1.1466.115.121.1.10 DESC 'Certificate Pair')

由于 Certificate（证书）以二进制方式传递，所以该语法中的值**必须**（MUST）也仅能使用二进制编码进行传递，进行传递时，使用"crossCertificatePair;binary"属性请求或返回。在 RFC 1778 中的"Certificate Pair"的 BNF 符号不被建议使用。

6.8、国家字符串（Country String）

(1.3.6.1.4.1.1466.115.121.1.11 DESC 'Country String')

该语法中的值与目录字符串语法（Directory String syntax）的值使用相同的编码规则。注意该语法的值被严格限制仅能使用两个可打印字符，这些可打印字符定义于 ISO 3166 [14]中。

CountryString = p p

例如：

US

6.9、分辨名（DN）

(1.3.6.1.4.1.1466.115.121.1.12 DESC 'DN')

分辨名语法（Distinguished Name）中的值被编码成参考文档[5]中定义的的表达形式。注意这种表达形式不可再还原成在 X.500 中为 DN 使用的 ASN.1 编码，因为在 RDN 中的任何 DirectoryString（目录字符串）元素的 CHOICE（选择）无法再被识别。

Examples (from [5]):

CN=Steve Kille,O=Isode Limited,C=GB
OU=Sales+CN=J. Smith,O=Widget Inc.,C=US
CN=L. Eagle,O=Sue\, Grabbit and Runn,C=GB
CN=Before\0DAfter,O=Test,C=GB
1.3.6.1.4.1.1466.0=#04024869,O=Test,C=GB
SN=Lu\C4\8Di\C4\87

6.10、目录字符串（Directory String）

(1.3.6.1.4.1.1466.115.121.1.15 DESC 'Directory String')

该语法中的字符串使用 ISO 10646（一个 Unicode 扩展集）的 UF8 形式进行编码。服务器和客户端**必须**（MUST）准备接收任意的 Unicode 字符的编码，包括现在仍未出现在任何字符集中的字符。

对于 PrintableString（可打印字符串）形式的字符，该值被编码成字符串值本身。

如果该字符串是 TeletexString 形式，则使用 UniversalString 形式将它翻译成它的等同字符，并且使用 UTF-8 编码。

如果该字符串是 UniversalString 或 BMPString 形式（参考文档[10]），则使用 UTF-8 对它们进行编码。

注意：除非属性值以二进制的方式进行传递，否则 DirectoryString 形式将不在协议中声明。面向 DAP 的服务器必须选择一种合适的形式。服务器**绝不**（MUST NOT）能仅因为它们包含了超出了可打印的 ASCII 范围的合法 Unicode 字符就拒绝这些值。

例如：

This is a string of DirectoryString containing #!%#@

6.11、DIT Content Rule Description

(1.3.6.1.4.1.1466.115.121.1.16 DESC 'DIT Content Rule Description')

该语法中的值根据下面的 BNF 进行编码。实现者应该注意本文档未来的版本中可能会扩展该 BNF 以包括附加的术语。

```
DITContentRuleDescription = "("
      numericoid     ; Structural ObjectClass identifier
      [ "NAME" qdescrs ]
      [ "DESC" qdstring ]
      [ "OBSOLETE" ]
      [ "AUX" oids ]       ; Auxiliary ObjectClasses
      [ "MUST" oids ]      ; AttributeType identifiers
      [ "MAY" oids ]       ; AttributeType identifiers
      [ "NOT" oids ]       ; AttributeType identifiers
     ")"
```

6.12、Facsimile Telephone Number

(1.3.6.1.4.1.1466.115.121.1.22 DESC 'Facsimile Telephone Number')

该语法中的值根据下面的 BNF 进行编码：

```
fax-number       = printablestring [ "$" faxparameters ]

faxparameters = faxparm / ( faxparm "$" faxparameters )

faxparm = "twoDimensional" / "fineResolution" /
                "unlimitedLength" /
                "b4Length" / "a3Width" / "b4Width" / "uncompressed"
```

上面第一个 printablestring（可打印字符串）是电话号码，基于 E.123（参考文档[15]），并且 faxparm 代表传真参数。

6.13、Fax

(1.3.6.1.4.1.1466.115.121.1.23 DESC 'Fax')

如果该语法中的值是包含 Group 3 Fax images 的八进制字符串，则使用参考文档[7]中定义的方式进编码。

6.14、Generalized Time

(1.3.6.1.4.1.1466.115.121.1.24 DESC 'Generalized Time')

该语法中的值被编码成可打印字符串，使用 X.208 中定义的表达方式。应注意必须指定

时区（time zone）。强烈建议使用 GMT 时间。例如：

199412161032Z

6.15、IA5 String

(1.3.6.1.4.1.1466.115.121.1.26 DESC 'IA5 String')

该语法中的值的编码是字符串值自身。

6.16、INTEGER

(1.3.6.1.4.1.1466.115.121.1.27 DESC 'INTEGER')

该语法中的值被编码成它们值的十进制表达方式，每个十进值数使用用它们对应的同等字符表示。例如：数字 1321 被表示成字符串"1321"。

6.17、JPEG

(1.3.6.1.4.1.1466.115.121.1.28 DESC 'JPEG')

该语法中的值使用 JPEG 文件交换格式（JFIF：JPEG File Interchange Format）被编码成包含 JPEG 图像的字符串。JFIF 在参考文档[8]中描述。

6.18、匹配规则描述（Matching Rule Description）

(1.3.6.1.4.1.1466.115.121.1.30 DESC 'Matching Rule Description')

matchingRules 类型的值根据 4.5 节给出的 BNF，被编码成字符串。

6.19、匹配规则使用描述（Matching Rule Use Description）

(1.3.6.1.4.1.1466.115.121.1.31 DESC 'Matching Rule Use Description')

matchingRuleUse 类型的值根据 4.5 节给出的 BNF，被编码成字符串。

6.20、MHS OR Address

(1.3.6.1.4.1.1466.115.121.1.33 DESC 'MHS OR Address')

该语法中的值根据参考文档[11]中定义的格式，被编码成字符串。

6.21、Name And Optional UID

(1.3.6.1.4.1.1466.115.121.1.34 DESC 'Name And Optional UID')

该语法中的值根据下面的 BNF 进行编码：

NameAndOptionalUID = DistinguishedName ["#" bitstring]

虽然'#'字符可以出现在代表一个分辨名的字符串中，但不允许引用其它特殊字符。该语法在 RFC 1778.之后被增加。

例如：

1.3.6.1.4.1.1466.0=#04024869,O=Test,C=GB#'0101'B

6.22、Name Form Description

(1.3.6.1.4.1.1466.115.121.1.35 DESC 'Name Form Description')

该语法中的值根据下面的 BNF 进编码。实现者应该注意本文档未来的版本中可能会扩展该 BNF 以包含附加的术语。

```
NameFormDescription = "(" whsp
      numericoid whsp    ; NameForm identifier
      [ "NAME" qdescrs ]
      [ "DESC" qdstring ]
      [ "OBSOLETE" whsp ]
      "OC" woid                ; Structural ObjectClass
      "MUST" oids              ; AttributeTypes
      [ "MAY" oids ]       ; AttributeTypes
whsp ")"
```

6.23、Numeric String

(1.3.6.1.4.1.1466.115.121.1.36 DESC 'Numeric String')

该语法中的字符串编码是字符串值自身。例如：

1997

6.24、对象类描述（Object Class Description）

(1.3.6.1.4.1.1466.115.121.1.37 DESC 'Object Class Description')

该语法中的值根据 4.4 节中的 BNF 进行编码。

6.25、OID

(1.3.6.1.4.1.1466.115.121.1.38 DESC 'OID')

对象标识符语法中的值根据 4.1 节中的"oid"的 BNF 进行编码。

例如：

1.2.3.4
cn

6.26、Other Mailbox

(1.3.6.1.4.1.1466.115.121.1.39 DESC 'Other Mailbox')

该语法中的值根据下面的 BNF 进编码。

otherMailbox = mailbox-type "$" mailbox

mailbox-type = printablestring

mailbox = <an encoded IA5 String>

上面描述中，mailbox-type 表示 mailbox 所在邮件系统的类型，例如"MCIMail"；mailbox 是位于由 mailbox-type 定义的邮件系统中的实际邮件箱。

6.27、Postal Address

(1.3.6.1.4.1.1466.115.121.1.41 DESC 'Postal Address')

该语法中的值根据下面的 BNF 进行编码。

postal-address = dstring *("$" dstring)

上面描述中，一个邮寄地址值的每个 dstring 组成都使用目录字符串（Directory String）语法进行编码。字符'\'和'$'，如果出现在组成部分中，可以象 4.3 节中所描述的一样标记出来。许多服务器限制邮寄地址为 6 行，每行最多 30 个字符。

例如：

1234 Main St.$Anytown, CA 12345$USA
\241,000,000 Sweepstakes$PO Box 1000000$Anytown, CA 12345$USA

6.28、Presentation Address

(1.3.6.1.4.1.1466.115.121.1.43 DESC 'Presentation Address')

该语法中的值使用在 RFC 1278（参考文档[6]）中的描述表达。

6.29、可打印字符串（Printable String）

(1.3.6.1.4.1.1466.115.121.1.44 DESC 'Printable String')

该语法中值的编码是字符串值自身。PrintableString（可打印字符串）限制为 4.1 节中的 p。

例如：

This is a PrintableString

6.30、Telephone Number

(1.3.6.1.4.1.1466.115.121.1.50 DESC 'Telephone Number')

该语法的值使用与可打印字符串相同的编码方式。电话号码在 X.520 中建议为国际形式，在 E.123（参考文档[15]）中的描述。

例如：

+1 512 305 0280

6.31、UTC Time

(1.3.6.1.4.1.1466.115.121.1.53 DESC 'UTC Time')

该语法中的值使用与可打印字符串相同的编码方式，该字符串包含一个 UTCTime 值。它已成为历史；新的属性定义**应该**（SHOULD）使用 GeneralizedTime 代替。

6.32、LDAP Syntax Description

(1.3.6.1.4.1.1466.115.121.1.54 DESC 'LDAP Syntax Description')

该语法中的值根据 4.3.3 节中的 BNF 进行编码。

6.33、DIT Structure Rule Description

(1.3.6.1.4.1.1466.115.121.1.17 DESC 'DIT Structure Rule Description')

该语法中的值根据下面的 BNF 进行编码。

```
DITStructureRuleDescription = "(" whsp
      ruleidentifier whsp                    ; DITStructureRule identifier
      [ "NAME" qdescrs ]
      [ "DESC" qdstring ]
      [ "OBSOLETE" whsp ]
      "FORM" woid whsp                       ; NameForm
      [ "SUP" ruleidentifiers whsp ] ; superior DITStructureRules
")"

ruleidentifier = integer

ruleidentifiers = ruleidentifier |
      "(" whsp ruleidentifierlist whsp ")"

ruleidentifierlist = [ ruleidentifier *( ruleidentifier ) ]
```

7、对象类（Object Classes）

服务器**应该**（SHOULD）识别参考文档[12]中第 7 节中的所有标准类的名称。

7.1、扩展对象类（Extensible Object Class）

如果一个条目中出现了 extensibleObject 对象类，那么该条目允许任意拥有任何属性。该对象类的属性列表可以隐式的是所有属性的集合。

(1.3.6.1.4.1.1466.101.120.111 NAME 'extensibleObject' SUP top AUXILIARY)

该条目中的其它对象类的强制属性仍然需要填写。（译者注：强制属性也可以称为必选属性，即如果你使用了某个对象类，则该对象类所定义的必选属性也必须填写）

注意，不是所有的服务器都实现了该对象类，并且对于那些未实现该对象类的服务器而言，它将拒绝含有该对象类的条目的添加请求，或者向条目中添加这个对象类的修改请求。

7.2、子模式（subschema）

该对象类在子模式条目中使用。

(2.5.20.1 NAME 'subschema' AUXILIARY
MAY (dITStructureRules $ nameForms $ ditContentRules $
objectClasses $ attributeTypes $ matchingRules $
matchingRuleUse))

ldapSyntaxes 作为操作性属性（operational attribute）也出现在子模式条目中。

8、匹配规则（Matching Rules）

实现 extensibleMatch（扩展匹配）过滤器的服务器**应该**（SHOULD）允许本节所列的所有匹配规则可以在 extensibleMatch 中使用。总的来说，只要匹配规则的判定语法（assertion syntax）与属性的值语法相同，那么这些服务器**应该**（SHOULD）允许匹配规则与该服务器已知属性类型一起使用。

服务器**可以**（MAY）实现附加的匹配规则。

8.1、Matching Rules used in Equality Filters

服务器**应该**（SHOULD）有能力执行下面的匹配规则。

对于所有的这些规则，它们的判定语法与值语法相同。

(2.5.13.0 NAME 'objectIdentifierMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38)

如果客户端提供了一个使用 objectIdentifierMatch 的过滤器，而 objectIdentifierMatch 的 matchValue 的 oid 是"descr"形式，但该 oid 并不能被服务器识别，则该过滤器为 Undefined（未定义）。

(2.5.13.1 NAME 'distinguishedNameMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12)

(2.5.13.2 NAME 'caseIgnoreMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15)

(2.5.13.8 NAME 'numericStringMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.36)

(2.5.13.11 NAME 'caseIgnoreListMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.41)

(2.5.13.14 NAME 'integerMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27)

(2.5.13.16 NAME 'bitStringMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.6)

(2.5.13.20 NAME 'telephoneNumberMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.50)

(2.5.13.22 NAME 'presentationAddressMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.43)

(2.5.13.23 NAME 'uniqueMemberMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.34)

(2.5.13.24 NAME 'protocolInformationMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.42)

(2.5.13.27 NAME 'generalizedTimeMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24)

(1.3.6.1.4.1.1466.109.114.1 NAME 'caseExactIA5Match'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26)

(1.3.6.1.4.1.1466.109.114.2 NAME 'caseIgnoreIA5Match'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26)

当执行 caseIgnoreMatch，caseIgnoreListMatch，telephoneNumberMatch，caseExactIA5Match 和 caseIgnoreIA5Match 时，多个连在一起的空格字符被当作一个空格，并且头和尾的空格被忽略。

客户端**绝不**（MUST NOT）能主观认为服务器能够翻译 Unicode 值。

8.2、Matching Rules used in Inequality Filters

服务器**应该**（SHOULD）有能力执行下列匹配规则，这些匹配规则在过 greaterOrEqual（大于等于）和 lessOrEqual（小于等于）滤器中使用。

(2.5.13.28 NAME 'generalizedTimeOrderingMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24)

(2.5.13.3 NAME 'caseIgnoreOrderingMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15)

每个服务器的具体实现（译者注：即每个 LDAP 服务器厂商开发的服务器）可以定义

不同的 caseIgnoreOrderingMatch 匹配规则的排序顺序。

8.3、Syntax and Matching Rules used in Substring Filters

Substring 判定语法仅用作扩展匹配的判定值语法。它不作为属性语法或 substring（子串）过滤器使用。

(1.3.6.1.4.1.1466.115.121.1.58 DESC 'Substring Assertion')

Substring 判定根据下面的 BNF 进行编码：

```
substring = [initial] any [final]
initial = value
any = "*" *(value "*")
final = value
```

<value>的产生物是 UTF-8 编码的字符串。'\'和'*'字符应该出现在<value>的产生物中，如何引用它们在 4.3 节中描述。

服务器**应该**（SHOULD）有能力执行下列匹配规则，这些匹配规则 substring 过滤器中使用。

(2.5.13.4 NAME 'caseIgnoreSubstringsMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.58)

(2.5.13.21 NAME 'telephoneNumberSubstringsMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.58)

(2.5.13.10 NAME 'numericStringSubstringsMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.58)

8.4、Matching Rules for Subschema Attributes

允许客户端修改子模式（subschema）条目的服务器**必须**（MUST）支持下列匹配规则，因为这些匹配规则是某些子模式属性（subschema attributes）的相等判定匹配规则。

(2.5.13.29 NAME 'integerFirstComponentMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27)

(2.5.13.30 NAME 'objectIdentifierFirstComponentMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38)

实现者应该注意虽然某些属性将上述这些匹配规则作为相等匹配规则，但这些匹配规则的判定语法（INTEGER 或 OID）却与这些属性的值语法不同。

如果客户端提供了一个使用 objectIdentifierFirstComponentMatch 的过滤器，而 objectIdentifierFirstComponentMatch 的匹配值是"descr"形式，并且 OID 不能被服务器识别，则过滤器为 Undefined（未定义）。

9、安全考虑（Security Considerations）

9.1、泄密（Disclosure）

目录条目的属性被用于提供有关现实世界对象的描述信息：它们能是人、组织或设备。绝大多数国家都有针对关于人的信息公开的隐私法。

9.2、在安全应用程序中属性值的使用（Use of Attribute Values in Security Applications）

一个 AttributeValue 的值，从它的 X.501 格式到 LDAP 字符串形式的转换，并不总是可以反向转换回同样的 BER 或 DER 形式。一个例子是：需要一个 DN 的 DER 形式的情形是 X.509 证书的验证。

例如，由带有一个 AVA 的 RDN 组成的分辨名，在该分辨名中，属性类型是 commonName，值是 TeletexString 类型的字符'Sam'，它在 LDAP 中表示为字符串 CN=Sam。另一个分辨名值仍然是'Sam'，但是值是 PrintableString 类型，它将具有相同的表示 CN=Sam。

需要重构值的 DER 形式的应用程序，在将一个分辨名转换成 LDAP 格式时**不应该**（SHOULD NOT）使用字符串表达的属性语法。相应的，它**应该**（SHOULD）使用二进制（binary）语法。

10、感谢（Acknowledgements）

本文档充分地基于 RFC 1778，它由 Tim Howes，Steve Kille，Wengyik Yeong 和 Colin Robbins 撰写。

许多在本文档和相关文档定义的属性语法是从 QUIPU 和 IC R3 X.500 实现中使用的属性语法所修改而成的。对这两个文档中的语法定义作出贡献的作者表示诚挚的感谢。

11、作者地址（Authors' Addresses）

Mark Wahl
Critical Angle Inc.
4815 West Braker Lane #502-385
Austin, TX 78759
USA
Phone: +1 512 372-3160
EMail: M.Wahl@critical-angle.com

Andy Coulbeck
Isode Inc.
9390 Research Blvd Suite 305
Austin, TX 78759
USA
Phone: +1 512 231-8993
EMail: A.Coulbeck@isode.com

Tim Howes
Netscape Communications Corp.
501 E. Middlefield Rd, MS MV068
Mountain View, CA 94043
USA
Phone: +1 650 937-3419
EMail: howes@netscape.com

Steve Kille
Isode Limited
The Dome, The Square
Richmond
TW9 1DT
UK
Phone: +44-181-332-9091
EMail: S.Kille@isode.com

12、参考书目（Bibliography）

[1] Wahl, M., Howes, T., and S. Kille, "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997.

[2] The Directory: Selected Attribute Types. ITU-T Recommendation X.520, 1993.

[3] The Directory: Models. ITU-T Recommendation X.501, 1993.

[4] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[5] Wahl, M., Kille, S., and T. Howes, "Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names", RFC 2253, December 1997.

[6] Kille, S., "A String Representation for Presentation Addresses", RFC 1278, November 1991.

[7] Terminal Equipment and Protocols for Telematic Services – Standardization of Group 3 facsimile apparatus for document transmission. CCITT, Recommendation T.4.

[8] JPEG File Interchange Format (Version 1.02). Eric Hamilton, C-Cube Microsystems, Milpitas, CA, September 1, 1992.

[9] Yergeau, F., "UTF-8, a transformation format of Unicode and ISO 10646", RFC 2044, October 1996.

[10] Universal Multiple-Octet Coded Character Set (UCS) - Architecture and Basic Multilingual Plane, ISO/IEC 10646-1 : 1993 (With amendments).

[11] Hardcastle-Kille, S., "Mapping between X.400(1988) / ISO 10021 and RFC 822", RFC 1327, May 1992.

[12] Wahl, M., "A Summary of the X.500(96) User Schema for use with LDAPv3", RFC 2256, December 1997.

[13] Crocker, D., "Standard of the Format of ARPA-Internet Text Messages", STD 11, RFC 822, August 1982.

[14] ISO 3166, "Codes for the representation of names of countries".

[15] ITU-T Rec. E.123, Notation for national and international telephone numbers, 1988.

13、完整的版权声明（Full Copyright Statement）

略