



中华人民共和国通信行业标准

YD/T 2142-2010

基于国际多语种域名体系的 中文域名总体技术要求

General technical specification for Chinese domain names

2010-12-29 发布

2011-01-01 实施

中华人民共和国工业和信息化部 发布

目次

前 言.....II

引 言.....III

1 范围.....1

2 规范性引用文件.....1

3 缩略语和术语.....1

 3.1 缩略语.....1

 3.2 术语.....1

4 解析体系.....4

 4.1 CDNA.....4

 4.2 应用要求和适用性.....5

 4.3 字符转换操作.....5

 4.4 使用 DNS 的典型应用.....9

5 注册管理机制.....10

 5.1 中文字符变体.....10

 5.2 注册管理.....10

附录 A（规范性附录） 中文异体对照表句法规定.....17

附录 B（资料性附录） 中文异体对照表示例.....19

前 言

本标准主要采用了 IETF 的标准 RFC 3490《国际化域名与应用》(英文版)和 RFC 3743《国际化域名的注册管理指南》(英文版),根据 RFC 3490 和 RFC 3743,以及我国的实际情况起草的,目的在于推动中文域名在中国的应用、发展和普及,有利于在中文用户群体中推广使用中文文化的域名。本标准在技术内容上与 RFC 3490 和 RFC 3743 保持一致,实现本标准与实现 RFC 3490 和 RFC 3743 的结果是一样的。

本标准是中文域名标准系列中的第一个标准,它包括了核心的解析体系规范以及注册管理规范,是制订其他中文域名标准的基础。该系列标准的名称和结构如下:

- 1、基于国际多语种域名体系的中文域名总体技术要求
- 2、基于国际多语种域名体系的中文域名的编码处理规范
- 3、基于国际多语种域名体系的中文域名的一系列应用协议规范和协议测试规范
- 4、基于国际多语种域名体系的中文域名的一系列软件设计规范和软件测试规范
- 5、基于国际多语种域名体系的中文域名注册实施规范

本标准的“范围”一章是根据 GB/T 1.1-2000 的规则编写的;第 3 章“术语和定义”是将 RFC3490 和 RFC3743 中的术语和定义修改归集,并增加了一些适合本标准的特定的术语和定义;第 4 章和第 5 章,分别依据 RFC3490 和 RFC3743 的基本思路,将其中有关国际化域名的规定都转换成只针对中文域名的规定,以符合中国国情,但是其技术思路并未作修改。

本标准的附录 A 是规范性附录。

本标准的附录 B 是资料性附录。

本标准由中国通信标准化协会提出并归口。

标准起草单位:中国互联网络信息中心(CNNIC)。

标准主要起草人:李晓东、钱华林、毛伟、孙国念、王艳峰、姚健康、陈卉。

引 言

互联网是一个基于开放互联协议的网络，域名是互联网上的基础服务，是用于识别和定位互联网上计算机的层次结构式的字符标识，与计算机的互联网地址相对应。基于域名可以提供WWW、E-mail、FTP等应用服务。

实现对中文域名的支持首先应实现对国际化域名的支持。该标准的引用RFC 3490是国际化域名的标准。RFC 3743是针对中日韩的文字字符存在异体的问题而设计的。本标准的第4章主要对应RFC 3490，主要规定了如何在应用程序中实现对中文域名的支持。本标准的第5章主要对应RFC 3743，主要规定了中文域名的注册指南。要实现对本标准的支持，亦可直接参照RFC 3490和RFC 3743。

随着互联网的发展，中文用户的数量不断增加，对于中文域名的使用需求也在增加。但是中文域名和传统的英文域名有较大差别，比如：域名字段分隔符（中文句点和英文句点）不同，中文字符有多种形式（包括简体、繁体、异体、古体等等）；并且中文域名的字符集比传统的英文域名的字符集大很多。为了规范中文域名的使用，让中文用户能够方便的通过中文域名来使用互联网的各种应用服务，尽快制定中文域名标准，进而推动中文域名的使用是十分重要的。

基于国际多语种域名体系的中文域名总体技术要求

1 范围

本标准规定了在互联网体系上使用中文域名的总体技术要求,规定了使用中文域名的解析体系要求,并对中文域名的注册管理提出相应的规范。

本标准适用于各级域名注册管理机构、域名注册服务提供商以及软件厂商开发支持中文域名的应用或者服务等。

2 规范性引用文件

下列文件中的条款通过本标准的引用而成为本标准的条款。凡是注日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本标准。然而,鼓励根据本标准达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本标准。

IETF RFC 1034	域名的概念与应用
IETF RFC 1035	域名的实现与规范
IETF RFC 1122	互联网主机传输层要求
IETF RFC 1123	互联网主机的应用与支持要求
IETF RFC 2234	语法规则的扩展巴科斯范式: ABNF
IETF RFC 3454	国际化字符串预处理
IETF RFC 3490	国际化域名与应用
IETF RFC 3491	国际化域名预处理
IETF RFC 3492	一种适用于国际化域名应用的对统一码的编码方法: Punycode
IETF RFC 3743	国际化域名的注册管理指南
IETF Draft-xdlee-idn-cdnadmin-05	中文域名注册管理指南

3 缩略语和术语

3.1 缩略语

下列缩略语适用于本标准。

CDL	Chinese Domain Label	中文域名字段
CDN	Chinese Domain Name	中文域名
CDNA	Chinese Domain Names in Applications	中文域名与应用
DNS	Domain Name System	域名系统
LDH	Letters Digits Hyphen	字母、数字、连接符

3.2 术语

下列术语和定义适用于本标准。

3.2.1

LDH-DNS

根据RFC1034《域名的概念与应用》和RFC 1035《域名的实现与规范》的定义，只允许使用26个英文字母，数字0到9以及连接符“-”作为字符的域名系统。本标准在后续的阐述均将这种域名系统称为LDH-DNS系统，LDH即Letter、Digit和Hyphen三个英文单词的首字母简写。

3.2.2

通用字符编码

通用字符编码根据其位置或码位来识别字符，给每个字符提供的一个惟一的数字。比如说，U+4E96指的是在通用字符集中位于4E96处的字符。本标准的通用字符编码采用了ISO/IEC 10646-1: 2000，通用字符编码的集合称为通用字符集。

3.2.3

分隔符 Delimiter

LDH-DNS域名中的英文句点或中文域名中的中文句点。

3.2.4

域名字段 Label

域名中由分隔符隔开的几个部分。例如：对于一个完整的域名“www.cnnic.cn.”，“www”、“cnnic”、“cn”分别是三个域名字段。

3.2.5

中文域名字段 Chinese Domain Label

含有中文字符的域名字段。

3.2.6

中文域名 Chinese Domain Name

含有中文域名字段的域名。

3.2.7

解析器 Resolver

解析器负责接受应用程序的域名查询请求，最终由它来向域名服务器发送域名查询请求，并负责接受域名服务器的返回信息，再将结果发给应用程序，从而完成整个查询过程。

3.2.8

中文域名与应用 CDNA

关于如何在应用程序以及某些应用环境中使用或者实现中文域名的协议，它允许使用某些 LDH 字符（以特殊的前缀开始）来表示非 LDH 字符；不改变现有的域名服务器、解析器、或是协议单元；与下层协议无关，不需要改变现有的网络结构。

3.2.9

域名槽 Domain Name Slot

被设计用来携带域名的协议元素、函数参数、函数返回值等，例如：域名查询中的QNAME字段。

3.2.10

ZONE

是用于存储域名的数据库，它是域名空间树状结构的一部分，域名服务器是以ZONE为单位来管理域名空间的，ZONE中的数据保存在管理它的域名服务器中。当在现有的域中添加子域时，该子域既应可包含在现有的ZONE中，也应可为它创建一个新ZONE或包含在其他的ZONE中。一个域名服务器应可管理一个或多个ZONE，一个ZONE应可保存在多个域名服务器上。

3.2.11

Punycode

一种编码转换规则。运用这种规则应可实现通用字符编码字符串和LDH字符串的相互转换。

3.2.12

LDH编码前缀

由两个 LDH 字符后跟着两个连字符（其中字母不区分大小写）进行表示的前缀。用于中文域名的LDH 编码前缀是“xn--”。

3.2.13

DNS资源记录 DNS Resource Record

用于描述DNS ZONE信息的基本组成结构，包括：资源记录所有者（Owner）、记录类型（Type）、协议类型（Class）、生存时间（TTL）、记录数据（Rdata）等。详见RFC 1034和RFC 1035。

3.2.14

存储字符串 Stored String

用在协议标识符和名称实体中的字符串。

3.2.15

查询字符串 Query String

用于和已保存的标示符相匹配或者相比较的字符串，例如域名查询中的域名部分。

3.2.16

异体对照表 Language Variant Table

在域名注册和管理时用到的字符表。表的格式分为三栏，第一栏为有效字栏，第二栏为建议字栏，第三栏为异体字栏。异体对照表是域名注册和管理的基础。

3.2.17

有效字 Valid Code Point

在异体对照表中，位于第一栏的码位。这一栏中所有字符的集合构成了特定语言的有效字集合，这个集合用来检查用户注册域名的合法性，只有域名所有码位均属于这个集合，域名才被认为是合法的。

3.2.18

建议字 Preferred Variant

在异体对照表中，位于第二栏的码位。这一栏对应有效字栏。建议字表明了有效字在特定语言或者特定语言书写环境中的建议值，它构成的变体域名字段通常被注册在ZONE文件中。异体对照表中的建议字栏用来生成建议CDL。

3.2.19

异体字 Character Variant

在异体对照表中，位于第三栏的码位。这一栏对应于有效字在特定语言或者特定语言书写环境中的异体。异体字与建议字不同，基于异体字的替换通常被保留而并未注册（激活）。异体对照表中的异体字栏用来生成异体CDL。

3.2.20

CDL包 CDL Package

在特定的ZONE内成功注册CDL时产生含有CDL的包。包内的所有CDL应被“保留”；除了包的持有者外，包内的所有CDL不能被任何人注册。这些被保留的CDL可能被激活写入一个ZONE文件中。CDL包同时也包括与注册过程相关语言的标识。初始CDL及其包内所有CDL构成了一个单一的原子单元。

3.2.21

字符串预处理 Stringprep

按照 RFC 3454《国际化字符串预处理》的规定，对字符串进行处理的过程。

3.2.22

名字预处理 Nameprep

按照RFC 3491《国际化域名预处理》的规定，对字符串进行处理，其目的是使其可以用于域名字段，或者检查其是否可以用于域名字段。

4 解析体系

4.1 CDNA

本章确立了一个如何在应用程序中实现中文域名或者在应用环境中使用中文域名的协议框架，该框架最核心的内容是规定了如何在整个DNS解析体系中支持中文域名，该框架称为中文域名与应用（CDNA）。

LDH-DNS只能使用字母、数字和连接符组成的编码，而中文域名允许使用更大的字符集。CDNA采用向后兼容的表示法，用于解决在域名中使用中文字符集的问题。

只要某种应用（包括 DNS ZONE 文件和解析器接口）支持 LDH 字符，应可使用 CDNA 来支持 CDN。CDNA 完全包括在具体应用中。需要使用域名解析时，应在应用和解析器之间使用 CDNA；需要在域名写入 DNS ZONE 时，应在域名放入 ZONE 之前使用 CDNA。本标准规定了两种操作：

1) LDH 字符转换操作，应该用于将 CDN 发送给仅支持 LDH 字符的设备（比如解析器）之前，或是把 CDN 写入仅支持 LDH 字符的文件（比如 DNS ZONE 文件）之前使用。

2) 通用字符编码转换操作，用于向用户显示域名时使用，例如：将 DNS ZONE 中取出的域名显示给用户。

两种操作中，LDH 字符转换操作可能会失败；如果对某个域名使用 LDH 字符转换操作失败，则此域名不应作为中文域名使用，相应的有一些错误处理。

在进行两种操作前，域名的编码格式应是通用字符编码；如果域名使用其他编码方式，应先转成通用字符编码（LDH 码位也属于通用字符编码码位）。

CDNA 应对输入字符串进行“nameprep”（详见 RFC 3491），“nameprep”是“stringprep”（详见 RFC 3454）在 CDNA 中使用的具体表述，之后进行 punycode（详见 RFC 3492）转换。实现 CDNA 应实现“nameprep”和“punycode”，两者缺一不可。

4.2 应用要求和适用性

4.2.1 应用要求

CDNA 规定了以下四个应用要求:

1) 当点作为域名字段分隔符时, 以下这些字符也应等同于点: U+002E(英文句号“.”), U+3002(中文句号“。”), U+FF0E(全角句号“.”), U+FF61(半角中文句号“.”)。

2) 如果域名进入不支持 CDN 的域名槽时, 此域名应只包含 LDH 码, 应通过 LDH 字符转换操作来获取 LDH 形式的域名, 如果点用作域名字段分隔符, 则应把所有这些域名字段分隔符转成 U+002E。

3) 如果用户环境能处理非 LDH 形式的域名, 宜对用户隐藏从域名槽获得的 LDH 域名。当不能确定环境是否支持非 LDH 形式时, 应用应可使用非 LDH 形式(可能失败, 比如不能正确显示), 也应可使用 LDH 形式。中文域名的非 LDH 形式的等同域名应可通过通用字符编码转换操作对每个域名字段进行操作来获得。要求 2 和要求 3 同时需要满足时, 要求 2 优先。

4) 当比较两个域名字段时, 应比较它们的 LDH 形式在不区分字母大小写时是否相同。当比较两个域名时, 只要相应的域名字段都相同, 无论是否使用同样的分隔符, 都应认为整个域名相同。

4.2.2 适用性

如果没有特别指出不使用中文域名, CDNA 对所有域名槽中的中文域名都应是适用的。以下两种情况需要注意。

1) DNS 资源记录

除非在将来有特殊规定, 对于所有类别值(Class)不是 IN(Internet)的 DNS 资源记录, 其中的 Name 和 Rdata 字段, 不应使用 CDNA。

DNS 资源记录能否应用 CDNA, 完全由类别值决定, 而不是记录类型(Type), 没有需要特别注意的。除非有强制的特殊限制, 以后新定义的记录类型的 DNS 资源记录也应可应用 CDNA。

2) 域名中存储的非域名类型数据

本标准没有规定存储在域名里的非域名数据是否可以使用非 LDH 字符。

4.3 字符转换操作

中文域名应可通过转换, 以 LDH 形式放入不能识别 CDN 的域名槽中, 或者以通用字符编码形式显示给用户。

LDH 字符转换操作或通用字符编码转换操作的输入是一串特定的通用字符编码。本部分规定了转换的步骤。

转换流程示意如图 1 所示。

详细转换步骤如下所述:

步骤 1: 判断域名是“stored string”还是“query string”, 如果是“query string”, 标记上“AllowUnassigned”(详见 RFC 3491)。

步骤 2: 以分隔符为界, 应将域名分割成几个独立的域名字段, 域名字段不包括分隔符。

步骤 3: 对每一个域名字段, 应决定是否在主机名中限制只能使用 LDH 码。如果需要限制, 应为此字段标记上“UseSTD3ASCIIRules”(详见 RFC 3490)。

步骤 4: 对每个域名字段进行 LDH 字符转换或通用字符编码转换的处理。例如, 把域名放入不能识别 CDN 的域名槽时, 使用 LDH 字符转换操作; 将域名显示给用户时, 宜使用通用字符编码转换操作。

4.2.1 给出了应用要求的详细规定。

步骤 5: 如果第 4 步使用了 LDH 字符转换操作, 应把所有域名字段分隔符转成 U+002E (英文句号“.”)。

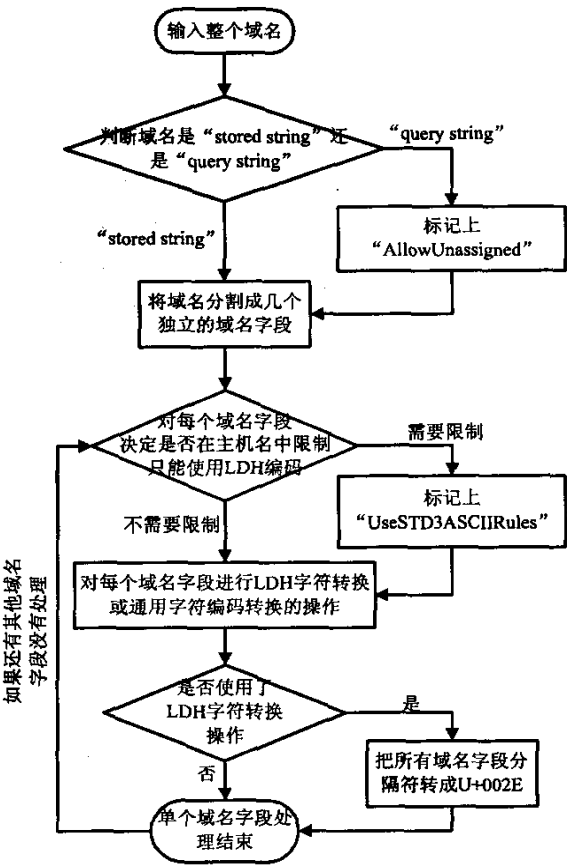


图 1 字符转换流程

4.3.1 LDH 字符转换操作

LDH 字符转换操作将域名字段由通用字符编码方式转换成 ASCII 编码方式。如果 LDH 字符转换操作转换成功, 初始域名字段和结果域名字段应是等同的。在 CDNA 中应用 LDH 字符转换操作时, 如果设置了 UseSTD3ASCIIRules 标记, 当输出结果含有非 LDH 形式的 ASCII 字符时, 表示转换操作失败。

转换过程中每一步的失败都可能导致 LDH 字符转换操作的失败, 如果转换失败则此域名不应作为中文域名使用。

LDH 字符转换操作的输入是一串通用字符编码码位, AllowUnassigned 标记和 UseSTD3ASCIIRules 标记。输出是一串 LDH 码位或者失败信息。

LDH 字符转换操作对已经是 LDH 形式的码位不作任何转换。多次运行 LDH 字符转换操作获得的结果和运行一次获得的结果相同。

LDH 字符转换的操作流程示意如图 2 所示。

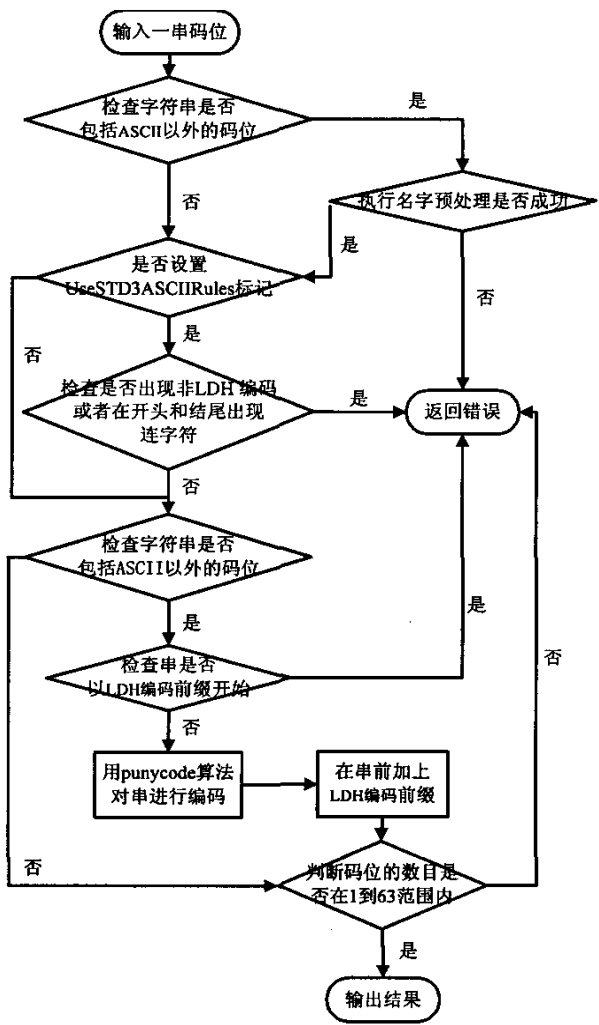


图 2 LDH 字符转换操作流程

详细的步骤如下所述：

步骤 1：如果字符串包括 ASCII 以外的码位，跳转到第 2 步，否则跳转到第 3 步。

步骤 2：执行名字预处理，使用 AllowUnassigned 标记，若失败应返回错误信息。

步骤 3：如果设置了 UseSTD3ASCIIRules 标记，进行下面的检查：

a) 确认不存在非字母数字或连接符（非 LDH）的 ASCII 码位，即不存在范围在 0..2C, 2E..2F, 3A..40, 5B..60, 和 7B..7F 之间的码位。

b) 确认在串的开头和结尾不存在 U+002D（英文连字符“-”）。

步骤 4：如果串里包括除 ASCII 之外的码位，应跳转到第 5 步，否则跳转到第 8 步。

步骤 5：确认串不是以 LDH 编码前缀开始，否则操作终止并返回错误信息。

步骤 6：用 punycode 编码算法对串进行编码。

步骤 7：在串前加上 LDH 编码前缀。

步骤 8：确认码位的数目在 1~63 范围内，返回 ASCII 形式的码位结果，否则返回错误信息。

4.3.2 通用字符编码转换操作

通用字符编码转换操作将域名字段转换成通用字符编码方式。如果输入的字符串是带有 LDH 前缀的 LDH 形式的域名字段，那么输出结果是一个等同的非 LDH 形式的中文域名字段；否则初始输入的字符串不变。

通用字符编码转换操作中如果任何一步失败，应将最初的输入串作为结果返回。

通用字符编码转换操作的输入是一串码位，AllowUnassigned 标记和 UseSTD3ASCIIRules 标记。输出是一串通用字符编码码位。

通用字符编码转换操作示意如图 3 所示。：

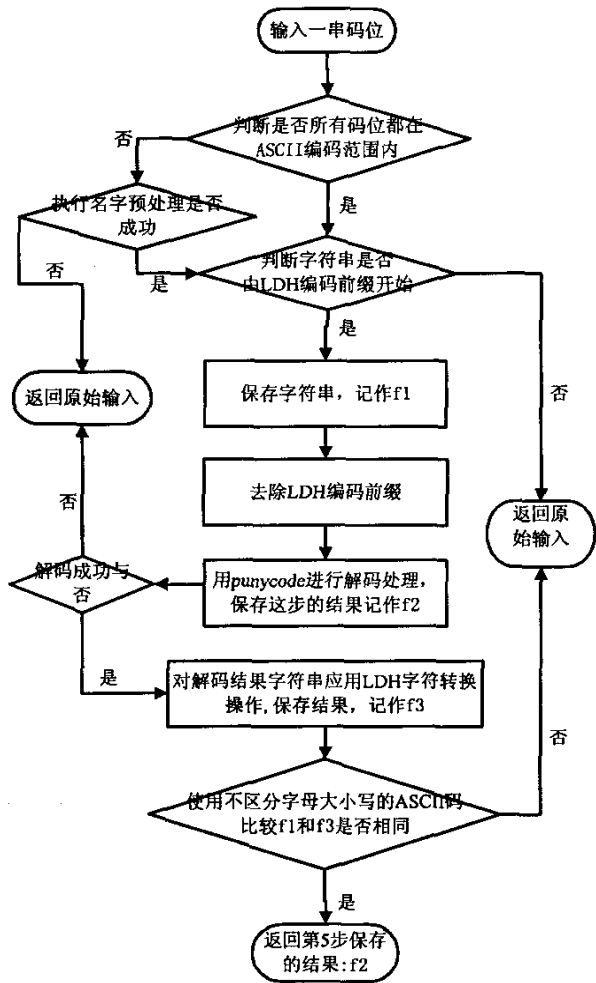


图 3 通用字符编码转换操作流程

- 步骤 1：如果串中的所有码位都在 ASCII 编码范围内（0...7F），应跳转到第 3 步。
- 步骤 2：执行名字预处理，使用 AllowUnassigned 标记，若失败应停止操作，应将原始输入作为结果返回。
- 步骤 3：确认字符串由 LDH 编码前缀开始，保存字符串；否则应将原始输入作为结果返回。
- 步骤 4：去除 LDH 编码前缀。

步骤 5: 用 punycode 解码算法对这个字符串进行解码处理, 保存这步的结果, 若失败应停止操作, 应将原始输入作为结果返回。

步骤 6: 对解码结果字符串应用 LDH 字符转换操作。

步骤 7: 使用不区分字母大小写的 ASCII 码比较, 确认第 6 步的结果和第 3 步保存的字符串相同; 否则应将原始输入作为结果返回。

步骤 8: 返回第 5 步保存的结果。

4.4 使用 DNS 的典型应用

应用 CDNA, 应可实现: 从用户端输入中文域名, 将中文域名显示给用户, 处理来自 DNS 的输入和输出, 处理与域名相关的其他协议。

各个部分以及之间的接口如图4所示。

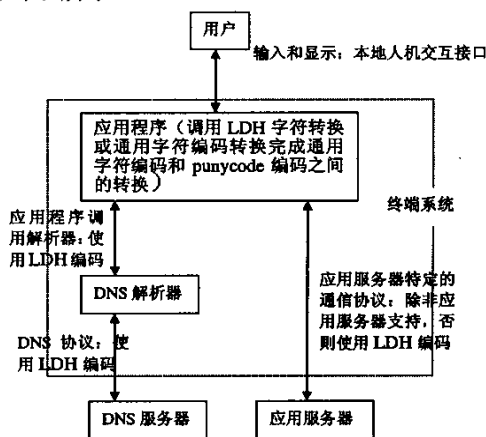


图 4 CDNA 技术应用框架

框图中的“应用程序”是将域名分成多个域名字段, 加上适当的标记, 之后执行 LDH 字符转换操作或通用字符编码转换操作。

4.4.1 应用中的登录和显示

CDNA 框架不影响用户和应用之间的接口。应用应可接受应用开发者中所希望使用的任何字符集中任何字符表示的域名, 并应可用相应的字符集的字符显示出来。

支持 CDNA 的应用应可支持两种格式 (LDH 形式和非 LDH 形式) 接收和显示中文域名。用户应可选择 LDH 域名的显示方式是 LDH 域名还是中文域名, 缺省配置是后者。因为 LDH 形式的编码是难理解的, 所以它应只显示给少部分真正需要的用户。

域名在很多地方被存储和传送。例如: 域名可以是电子邮件头信息的一部分或者是浏览器地址栏的输入。

使用域名的所有协议都可以处理 LDH 字符表示的所有域名, 因此这些协议同样可以处理中文域名的 LDH 形式。

4.4.2 应用和解析器

应用软件在进行域名查询操作时通常使用操作系统自带的函数。这些函数称为“解析器”, 应用软件和解析器通过应用程序接口 (API) 进行通信。

目前使用的解析器要求域名应是 LDH 编码形式, 因此应用软件应使用 LDH 字符转换操作对中文域

名字段进行转换，将不能直接表示为 LDH 字符形式的中文域名字段转换成 LDH 域名字段。

操作系统中可能会安装一套用于执行 LDH 字符转换操作的函数库。实际使用时，到达这些函数库的输入可能是具体应用中的一个或者多个字符串（在大部分操作系统中会是 UTF-8 和 UTF-16 编码，一些本地化的操作系统中可能是一些本地字符编码）。

支持 CDNA 的应用不但应能够处理 CDL 而且应能够处理传统域名（符合[RFC 1034]、[RFC 1035]、[RFC 1122《互联网主机传输层要求》和 RFC 1123《互联网主机应用与支持要求》）。

将来新版本的解析器内建议包含 LDH 字符转换和通用字符编码转换函数，可以直接接受中文域名的输入。这样，应用开发者应可把通用字符编码或者是其他中文编码格式的中文域名直接发送给新版本解析器。

4.4.3 DNS 服务器

在 ZONE 文件中存储的域名规应遵守 stringprep。

对于不能直接表示成 LDH 形式的中文域名字段，DNS 服务器应使用 LDH 字符转换操作产生的 LDH 形式。DNS 服务器处理的所有中文域名应使用只包含 LDH 字符的 LDH 形式。

4.4.4 避免给用户显示未经处理的 LDH 形式的编码

如果不希望让用户看见 LDH 形式的中文域名，则所有涉及到中文域名信息显示的不支持 CDNA 的应用应升级。

如果对 LDH 形式的域名使用通用字符编码转换操作解码后所得的中文域名不能正确显示（比如，域名中包含输出设备不能显示的字符），则应把域名以 LDH 形式显示，而不是使用通用字符编码（类似 U+FFFD）替代不能显示的字符。

4.4.5 中文域名的 DNSSEC 授权

如果需要对中文域名使用 DNSSEC，在对其进行私钥加密前，应使用 LDH 字符转换操作，把该中文域名转成 LDH 形式。

5 注册管理机制

5.1 中文字符变体

在中文域名中要用到中文字符。中文字符存在着“变体”（如同一个字可能有 5 种表示方式），这样可能导致某些中文字符可能被认为是同一个字符，但在计算机使用的字符集中，同一个概念上的字符却通过几个不同的码位来识别。外形相同的字符，或者是有相同或相似语义却被分配了不同码位的字符有可能使用户产生混淆。中文域名标准也会影响到一些互联网协议及其应用，增加了技术管理与服务方面的复杂性，在中文域名部署时有必要提高警惕来防止混淆与欺诈行为。本章规定了一个框架来处理这些变体以达到输入字符和它的变体等效的功能，同时减少用户在获得域名、使用域名时发生严重混淆的可能性。

5.2 注册管理

ZONE 管理者负责管理其控制下的中文域名字段，不论 ZONE 管理者负责较大的 ZONE 还是较小的 ZONE，他们对于所有 ZONE 的实际技术管理工作，如在 ZONE 内域名的添加、删除、授权及在域名持有者之间的转移，都是相似的。

本小节主要规定中文域名注册原则、注册步骤、CDL 包的使用和中文异体对照表的使用及更新。

5.2.1 基于的原则

虽然在这节介绍中, FCFS (First-Come, First-Served) 并没有被列为所基于的原则之一。在多数情况下, 处理域名权利纠纷的缺省策略是 FCFS。域名管理者应根据下面所提供的原则, 对特定的 ZONE 制定特定的优先级处理或权利管理策略来处理这些域名纠纷。

具体的原则如下:

1) CDL 通常由在语言中有具体意义的单词或词组组成。尽管 ZONE 管理可能并不需要“具体含义”作为注册标准, 但是宜使用有具体含义的 CDL 以避免用户在使用时产生混淆。

每一个即将被注册的 CDL 都应与中文异体对照表绑定后再进行管理。

2) 中文字符集只是通用字符集的一个子集。因此, 在 CDL 注册时, CDL 中的每个中文字符都应在该子集之中进行检查。

每个即将被注册的 CDL 应在通用字符集中的中文字符子集中予以验证。如果该 CDL 没能通过验证, 那么该 CDL 应是无效的, 不能进行注册。

3) 当一个 CDL 和中文异体对照表绑定的时候, 除了建议字外, 还可能有一些异体字。这些异体字与初始 CDL 中中文字符的码位可能存在潜在的混淆。因此, 由异体字构成的 CDL 不宜用于注册。

注册时, 所有由异体字产生的 CDL 都应予以保留。

所有保留的 CDL 通常不出现在 DNS ZONE 文件中, 不能够被解析。域名持有者应可要求将这些被保留的 CDL 写入 ZONE 文件中, 从而被激活, 使其可被解析。

ZONE 管理应建立有关如何激活被保留的 CDL 的本地策略。

4) 当一个 CDL 代表一个单词或词组时, 通常会有变体字出现, 其中的一个或多个变体有可能会比其他的变体使用更加频繁, 这些变体通常宜被作为建议字。

为了提高中文域名被终端用户正确解析的可能性, 由建议字生成的建议 CDL 宜都是可以被解析的。

5) 在 CDL 的注册过程中可能存在大量的变体 CDL, 某些 CDL 可能是在语言上毫无意义的或无效的的中文字符组合。因而 ZONE 管理应制定特定的步骤来采纳那些在语言上可接受的 CDL 的注册。

特定 ZONE 管理者宜施加额外的规则或处理过程来限制实际已保留或注册的异体 CDL 或建议 CDL 的数目。

6) 每一个 CDL 都对一些它的建议 CDL 和异体 CDL, 这些 CDL 之间应是相互等价的。为避免混淆, 所有这些 CDL 都应是 CDL 包的一部分, 应被分配给同一个域名持有者。

CDL 包在注册的时候创建并且具有原子特性: 一个 CDL 的转移或修改以 CDL 包作为整体进行。CDL 包内的任一个 CDL 不能被单独删除或转移; 任何重新注册, 转移或其他关系到 CDL 的操作都会影响到包内的相关 CDL。

如果特定 ZONE 的域名冲突解决策略和 CDL 包的原子特性原则发生冲突, 在这种情况下, 该策略应明确定义处理的优先级。

5.2.2 CDL 的注册

本节详细介绍两个概念: 中文异体对照表和 CDL 包。

5.2.2.1 中文异体对照表的使用

每一个 ZONE 应和特定的中文异体对照表相关。该对照表包括了表头部分和表体部分, 表头部分规定参考及版本信息。以下是表体部分:

1) 第一栏中所有字符的集合构成了中文的有效字集合, 该集合是通用字符集的子集, 用来检查用户注册域名的合理性 (如 5.2.1 中的原则 2)。同后面的注册过程类似, 这一栏作为索引来检查出现在 CDL 中要被处理的字符。一般认为有效字集合定义了中文的文字表示。

2) 第二栏包括的是和第一栏的通用字符编码字符相对应的建议字。这些建议字被用来生成建议 CDL, 这些建议 CDL 应是可被解析的 (如 5.2.1 中的原则 4)。在通常情况下, 所有的建议 CDL 在相关的 ZONE 文件中被激活, 从而当域名查询它们的时候应可得到正确解析。

3) 第三栏包括的是与有效字相对应的异体字。这些异体字用来产生异体 CDL, 然后被保留下来。这些异体 CDL 的注册、激活通常由域名持有人决定, 但是受 ZONE 管理的本地策略的制约。

每一栏中的每个条目应包括一个或多个码位, 这些码位表示为通用字符编码表中的数字字符形式, 数字字符后接可选的带圆括号的参考值。第一栏, 即有效字栏, 在特定行中可能仅有一个码位, 而其他栏可能有多个。任何一行都应可用 '#' 开头的注释结束。

对照表的正式句法以及其结构更加准确的定义在附录 A 中予以介绍, 对照表的示例在附录 B 中予以介绍。

中文异体对照表应由国家相关的工作组, 组织或团体提供。

5.2.2.2 CDL 包

在特定的 ZONE 中成功注册一个 CDL 后, ZONE 应会创建一个 CDL 包, 包的内容有:

- 1) 被注册的 CDL;
- 2) 与 CDL 绑定的中文语言;
- 3) 绑定的中文异体对照表的版本;
- 4) 被保留的 CDL;
- 5) 被加入到 DNS ZONE 文件中被激活的 CDL。

5.2.2.3 注册 CDL 的步骤

简单的算法流程如图 5 所示。

具体算法规定如下, “<=”表示等于或者赋予的意思;

步骤 1: $IN \leq$ 需要注册的初始 CDL。

$\{L\} \leq$ 与此 IN 相关联的中文在特定环境中的书写形式, 如中文简体形式, 中文繁体形式等。

该步骤应输入将要注册的 CDL 和与之绑定的中文在特定环境中的书写形式。

步骤 2: 生成此 IN 的 nameprep 处理后的版本, 然后对其进行映射和规格化处理。如果本步骤失败, 注册过程停止并返回错误信息。

步骤 2.1: $NP(IN) \leq$ 经过 nameprep 处理后的 IN 。

步骤 2.2: 同特定 ZONE 内其他已经创建的 CDL 包进行核对, 来检查 $NP(IN)$ 中的 CDL 是否已经被注册或保留, 如果某个 CDL 已被注册或保留, 就存在冲突, 冲突应采用适用该 ZONE 的某种冲突解决策略。例如: 采用 FCFS 策略。

步骤 3: 对在 $\{L\}$ 中的每种形式 (AL), 进行每种形式循环处理。循环开始。

步骤 3.1: IN 中的 CDL 应在相应的中文形式的异体对照表中检查 CDL 中每个中文字符的有效性。若失败, 应跳出循环, 停止处理, 返回相应错误信息。

步骤 3.2: $PV(IN, AL) \leq$ 利用经过 nameprep 处理且通过 nameprep 验证的 CDL 从建议字生成建议

CDL 的集合。

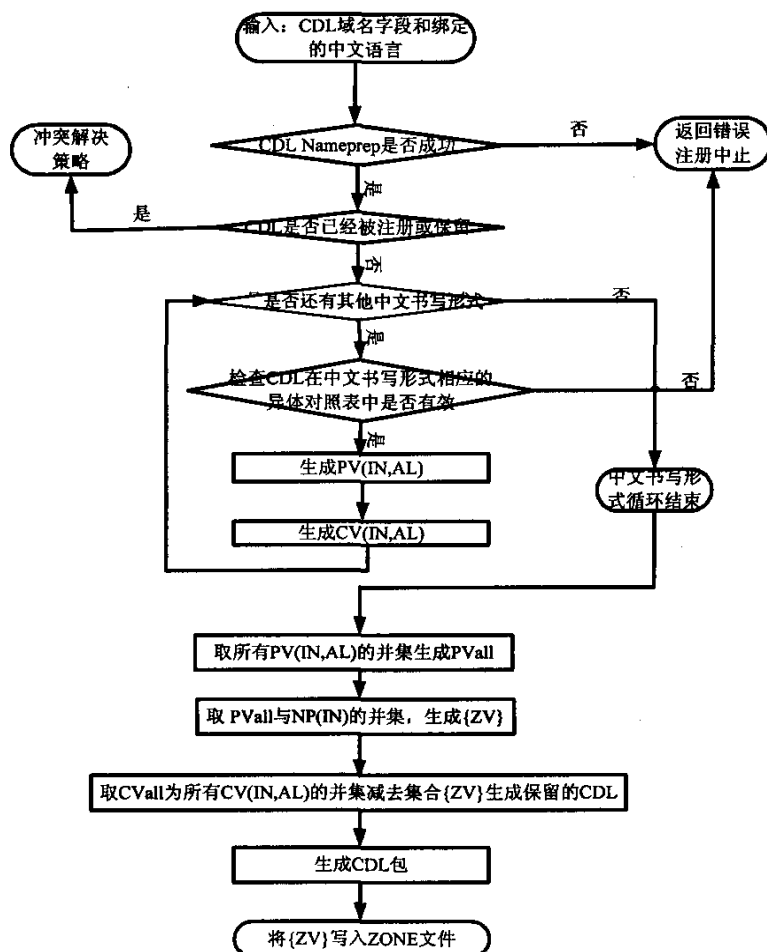


图5 CDL注册算法流程

该步骤利用经过 nameprep 处理且通过 nameprep 验证的 CDL 中的所有中文字符相对应的列在(AL) 相应的异体对照表的建议字栏的所有可能的建议字进行组合, 产生这个建议 CDL 的集合。这些产生的建议 CDL 应经过 nameprep 处理。如果对任何的建议 CDL 进行 nameprep 处理失败, 那么该 CDL 应被从这个集合中删除。然后对该集合内剩余的 CDL 进行检查来确定它们是否已被注册或保留。如果存在有 CDL 已被注册或保留, 应采用冲突解决策略。通常, 这种冲突解决策略不会阻止初始 CDL 的注册。例如, 如果采用 FCFS 策略, 那么已被注册或保留的 CDL 应会被从该 CDL 集合中删除掉, 但是初始 CDL 和其余的变体将有可能被注册。

该步骤 3.2 和下文中步骤 3.3 中, 如果初始 CDL 中的一个中文字符有相对应的建议字或异体字, 一个由建议字或异体字取代初始的中文字符所组成的新的 CDL 就产生了。重复上述过程, 应会产生许多 CDL。这些 CDL 和初始 CDL 一起形成 CDL 集合。

步骤 3.3: $CV(IN, AL) \Leftarrow$ 利用经过 nameprep 处理且通过 nameprep 验证的 CDL 从异体字生成异体 CDL 的集合。

该步骤利用经过 nameprep 处理且通过 nameprep 验证的 CDL 中的所有中文字符相对应的列在(AL)

相应的异体对照表的异体字栏的所有可能的异体字进行组合,产生这个异体 CDL 的集合,该集合中的所有 CDL 应是经过 nameprep 处理并为 nameprep 所接受的。如果采用 nameprep 处理的 CDL 的过程失败,那么该 CDL 应被从该集合中删除。然后再检查剩下的 CDL 以确保它们未被注册或未被保留。如果存在 CDL 已经被注册或被保留,应采用冲突解决策略。通常,除非采取其他的方法进行阻止,这种冲突解决策略不会阻止最初请求的初始 CDL 注册。例如,如果采用 FCFS 策略,若最初请求的初始 CDL 未被注册或保留,那么这个 CDL 将被注册。

步骤 3.4: 如果每种书写形式都处理结束,应停止循环;否则针对下一种书写形式跳转到步骤 3.1 继续执行循环。

步骤 4: $PV_{all} \leftarrow$ 取所有 $PV(IN,AL)$ 的并集。

ZONE 管理者应可对步骤 3.1 中产生的 $PV(IN,AL)$ 以及步骤 6 的结果施加一些额外的规则和处理过程来限制产生的建议 CDL 和异体 CDL 的数目。不同的 ZONE 应可添加不同的规则及处理过程。注意:取并集的操作会使相同的重复项只保留一个。

步骤 5: $\{ZV\} \leftarrow PV_{all}$ 与 $NP(IN)$ 的并集。产生最初的 ZONE 变体集合。这个集合包括 PV_{all} 和开始时被 Nameprep 处理过的初始 CDL。除非被进一步的处理过程排除,这些 ZONE 变体应被写入 DNS 的 ZONE 文件中被激活。

步骤 6: 取 CV_{all} 为所有 $CV(IN,AL)$ 的并集减去集合 $\{ZV\}$,从而产生保留而暂不被激活的 CDL 集合。

步骤 7: 使用上述的 IN , $\{L\}$, $\{ZV\}$ 和 CV_{all} 为 IN 生成 CDL 包。利用初始 CDL、与 CDL 绑定的中文书写形式、ZONE 变体和保留的 CDL 集合创建 CDL 包。如果对指定 ZONE 采用附加处理或过滤来消除语言上或格式上的不适合,那么这些处理的应用应在 CDL 包创建前进行。

步骤 8: 将 $\{ZV\}$ 写入 ZONE 文件。

采用 UseSTD13ASCIIRules[RFC 3490]激活的 CDL,在被写入 ZONE 文件前应经过 LDH 字符转换转换。这个转换使得 CDL 转换为“punycode”格式。如果任何需要被激活的 CDL 的 LDH 字符转换转换过程失败,该 CDL 不应被写入 ZONE 文件中。

5.2.3 CDL 与 CDL 包的删除与转移

在传统的域名管理中,每一个域名字段应独立于该 ZONE 的其他字段。域名字段的注册、删除和转移都是与其他域名字段无关的。根据本标准中所规定,每一个 CDL 都应和 CDL 包中所有被激活的或被保留的 CDL 绑定。

CDL 包应被视为一个独立的原子单元,同一个包内的所有 CDL 应属于一个单独的域名持有者。如果处理域名冲突的本地策略要求对 CDL 的转移与删除独立于所属的 CDL 包,那么应优先使用域名冲突解决策略。

在特定 ZONE 中,一个 CDL 包被删除后,应可被再次注册。一个 CDL 包的删除并不会改变该 ZONE 中任何其他的 CDL 包。

5.2.4 CDL 变体的激活与去活

因为 CDL 包中既有已激活的 CDL,也有未激活的 CDL,所以应了解变体 CDL 的激活和去活过程。

5.2.4.1 激活算法

简单的算法流程如图 6 所示。

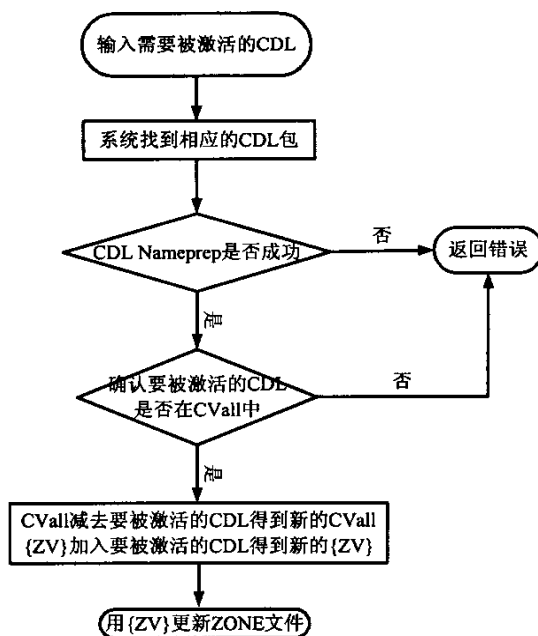


图6 CDL 激活算法流程

具体算法规定如下：

步骤 1: $IN \leftarrow$ 需要被激活的 CDL。

$PA \leftarrow$ 相应的 CDL 包。

该步骤输入要被激活的 CDL，然后系统找到该 CDL 所属的 CDL 包。

步骤 2: $NP(IN) \leftarrow$ 经过 nameprep 处理的 IN。

该步骤对要被激活的 CDL 采用 nameprep 处理保证该 CDL 的有效性。

步骤 3: 如果 $NP(IN)$ 不在 CVall 中，应停止。

该步骤确认要被激活的 CDL 确实在保留的 CDL 集合中，如果没有，应停止操作并返回错误信息。

步骤 4: $CVall \leftarrow$ CVall 扣除 $NP(IN)$ 之后的集合。

$\{ZV\} \leftarrow$ $\{ZV\}$ 和 $NP(IN)$ 的并集。

在一个 CDL 包内，应把要被激活的 CDL 从保留的 CDL 集合中减去，并应把其加入到 $\{ZV\}$ 中。

步骤 5: 用 $\{ZV\}$ 更新 ZONE 文件。

把新的 $\{ZV\}$ 更新到 ZONE 中。

5.2.4.2 去活算法

简单的算法流程如图 7 所示。

具体算法规定如下：

步骤 1: $IN \leftarrow$ 需要去活的 CDL

$PA \leftarrow$ 相应的 CDL 包

该步骤输入要被去活的 CDL，然后系统找到该 CDL 所属的 CDL 包。

步骤 2: $NP(IN) \leftarrow$ 经过 nameprep 处理过的 IN

该步骤对要被去活的 CDL 采用 nameprep 处理保证该 CDL 的有效性。

步骤 3：如果 NP(IN)不在{ZV}中，应停止。

确认需要去活的经过 nameprep 处理后的 CDL 确实在集合{ZV}中。如果不在{ZV}中，操作应停止并返回错误信息。

步骤 4：CVall<=CVall 和 NP(IN)的并集

{ZV}<={ZV}扣除 NP(IN)之后的集合。

在一个 CDL 包内，从激活的 CDL 变体集合中删除被去活的 CDL，并应把这个 CDL 添加到保留的变体 CDL 集合中。

步骤 5：用{ZV}更新 ZONE 文件。

把新的{ZV}更新到 ZONE 中。

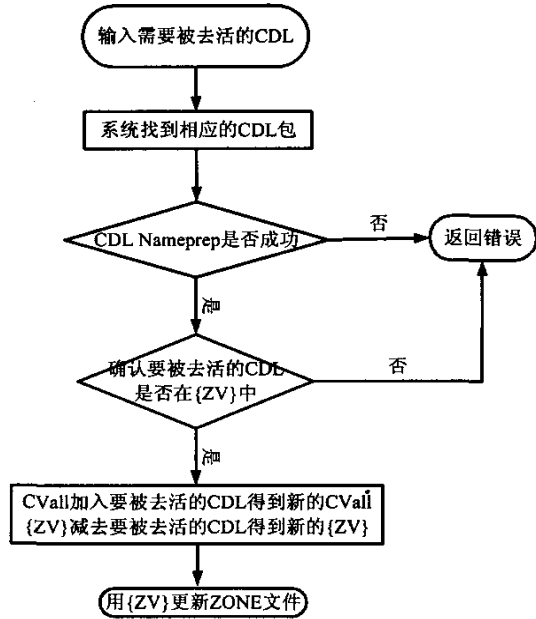


图 7 CDL 去活算法流程

5.2.5 中文异体对照表的变更处理

中文异体对照表处在不断的变化之中，这些变化不一定是向后兼容的。更新中文异体对照表可能会产生不同的建议字集合和异体字集合。为了维护 CDL 包的原子性，当中文异体对照表被更改后，采用原来的中文异体对照表创建的 CDL 包应不会被更新或受到影响。

附录 A

(规范性附录)

中文异体对照表句法规定

A.1 中文异体对照表句法规定

A.1中文异体对照表的正式句法如下所示，利用了IETF“ABNF”元语言（详见RFC 2234《语法规范的扩展巴科斯范式：ABNF》）。下面是对该句法的规定。

ABNF 句法

LanguageVariantTable = 1*ReferenceLine VersionLine 1*EntryLine

ReferenceLine = "Reference" SP RefNo SP RefDescription [Comment] CRLF

RefNo = 1*DIGIT

RefDescription = *[VCHAR]

VersionLine = "Version" SP VersionNo SP VersionDate [Comment] CRLF

VersionNo = 1*DIGIT

VersionDate = YYYYMMDD

EntryLine = VariantEntry/Comment CRLF

VariantEntry = ValidCodePoint ";" PreferredVariant ";" CharacterVariant [Comment]

ValidCodePoint = CodePoint

RefList = RefNo 0*("," RefNo)

PreferredVariant = CodePointSet 0*("," CodePointSet)

CharacterVariant = CodePointSet 0*("," CodePointSet)

CodePointSet = CodePoint 0*(SP CodePoint)

CodePoint = 4*8DIGIT ["(" Reflist ")"]

Comment = "#" *VCHAR

YYYYMMDD 是一个用字母表示代表日期的整数，其中YYYY表示4位的年份，MM代表2位的月份，DD代表2位的日期。

A.2 句法的注释与解释

以#开头的行或者#后面的部分被视为注释。注释在对表进行处理时没有任何意义，在#与行尾之间也没有具体的句法要求。表中的空行应可被完全忽略。

中文异体对照表由国家相关的工作组，组织或特定团体提供。通常情况下，该表应是基于某些已建立的标准或者重要参考文献而制定的。定义中文异体对照表的工作组在表的开头应该用标签“Reference”后接一个整数按照顺序记录下所参考的标准以及对该标准的描述信息。

示例如下：

Reference 0 ISO/IEC 10646-1: 2000

每一个中文异体对照表应有版本号及其发行日期。“version”后接一个整数，再接一个YYYYMMDD格式的日期。

示例如下：

Version 4.0 20050331

该表分为三栏，由分号进行分隔：“有效字”，“建议字”和“异体字”。

其中“有效字”栏是能够进行有效注册的通用字符集中的中文字符集的子集。

可能会有多于一个的建议字，因此在一个“建议字”栏中可能存在多个条目。

异体字栏包括了有效字的所有变体。由于有效字是自身的一种变体，为了避免冗余，它不必在异体字栏中重复出现。

如果建议字栏和异体字栏中的变体字由多个码位组成，码位以空格隔开。如果在建议字栏或异体字栏中有多个变体存在，每一个变体由逗号隔开。

列在建议字栏中的任何码位应被相关的中文语言规则所许可。

表中的每一个码位都应有一个对应的明确参考号（和该参考绑定）证明该条目来源正确。该参考号放置在码位后面括号内。如果存在有多个参考，那么参考号放置在一列括号中，这些括号用逗号隔开。

附 录 B
(资料性附录)
中文异体对照表示例

B.1 中文异体对照表示例

下面给出了一个中文异体对照表的例子（注：这个表不是真实的中文异体对照表，它不包括实际操作中需要的足够的条目）。如有需要，可以参考IANA（Internet Assigned Numbers Authority）已注册的真实的异体对照表。

Reference 0 ISO/IEC 10646-1: 2000

Reference 1 简化字总表

Reference 2 异体字整理表

Reference 3 汉语大字典

Reference 4 Unihan工程用汉字关联字表

Reference 5 GB2312

Version 4.0 20050331

#2005年3月31日4.0版本

4E2D(0);4E2D(5)

#中；中

4EA4(0);4EA4(5)

#交；交

4E9E(0);4E9A(1,3);4E9A(1,3),4E9C(3)

#亞；亚；亚，亜

56FD(0);56FD(1,3);56EF(4),5700(3),570B(1,3)

#国；国；国，圀，國