

## **The DNSII Multilingual Domain Name Protocol**

### STATUS OF THIS MEMO

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The reader is cautioned not to depend on the values that appear in examples to be current or complete, since their purpose is primarily educational. Distribution of this memo is unlimited.

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

A copy of this particular draft is also archived at <http://www.dnsii.org>.

### Abstract

The core thinking for DNSII is that multilingual DNS requests SHOULD be signaled within a DNS label whether by way of a binary tag or an alphanumeric prefix, and that compatibility to legacy client applications SHOULD be taken into concern alongside legacy server implementations.

Besides the original specifications, four alternatives including the use of EDNS are included for discussion purposes in this document.

Historically, the DNS is capable of handling only names within the basic English alphanumeric character set (plus the hyphen), yet the standards were so elegantly and openly designed that the extension of the DNS into a multilingual and symbols based system proves to be possible with simple adjustments.

These adjustments will be made on both the client side and the server

side. However, DNSII works on the principal that it is preferable to make the transition to multilingual domain names seamless and

transparent to the end-user. Which means initially the server SHOULD take the primary responsibility for the technical implementation of the changes required for a multilingual Internet.

The DNSII protocol is designed to allow the preservation of interoperability, consistency and simplicity of the original DNS, while being expandable and flexible for the handling of any character or symbol used for the naming of an Internet domain. DNSII intends to provide a platform for the implementation of multilingual domain names.

## Table Of Contents

<a href="#">1. Introduction.....</a>	<a href="#">2</a>
<a href="#">1.1 Terminology.....</a>	<a href="#">2</a>
<a href="#">1.2 DNSII.....</a>	<a href="#">3</a>
<a href="#">2. DNSII Protocol.....</a>	<a href="#">3</a>
<a href="#">2.1 InPacket DNSII Identifier.....</a>	<a href="#">3</a>
<a href="#">2.2 InPacket Label Encoding Type (ILET).....</a>	<a href="#">4</a>
<a href="#">2.3 The Rationale for using ILET.....</a>	<a href="#">5</a>
<a href="#">2.4 Considerations for Specific Requests.....</a>	<a href="#">6</a>
<a href="#">2.4.1 PTR Records.....</a>	<a href="#">6</a>
<a href="#">3. Alternate Implementations.....</a>	<a href="#">7</a>
<a href="#">3.1 Restricted ILET Values.....</a>	<a href="#">7</a>
<a href="#">3.2 Reduced ILET Bit Allocation.....</a>	<a href="#">8</a>
<a href="#">3.3 DNSII over EDNS.....</a>	<a href="#">9</a>
<a href="#">3.3.1 DNSII Identifier using EDNS.....</a>	<a href="#">9</a>
<a href="#">3.3.2 ILET using EDNS OPT RRs.....</a>	<a href="#">10</a>
<a href="#">4. Implementation &amp; Deployment Strategies.....</a>	<a href="#">11</a>
<a href="#">5. IDN Requirements Considerations.....</a>	<a href="#">12</a>
<a href="#">6. DNSSEC, EDNS and IPv6 Considerations.....</a>	<a href="#">12</a>
<a href="#">7. Intellectual Property Considerations.....</a>	<a href="#">13</a>
<a href="#">8. References.....</a>	<a href="#">13</a>

## [1. Introduction](#)

This Internet-draft describes details of the DNSII Multilingual Domain Name protocol. The Internet-Draft assumes that the reader is familiar with the concepts discussed in the widely distributed RFCs "Domain Names Concepts and Facilities" [[RFC 1034](#)] and "Domain Names Implementation and Specification" [[RFC 1035](#)].

### [1.1 Terminology](#)

The key words "MUST", "SHALL", "REQUIRED", "SHOULD", "RECOMMENDED",

and "MAY" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

A number of multilingual characters are used in this document for examples. Please select your view encoding type to UTF-8 for it to be displayed properly.

DNSII-MDNP

Multilingual Domain Name Protocol

August 2000

## [1.2](#) DNSII

The DNSII specifications takes a radically different approach: it successfully identifies the difference between original DNS and DNSII packets within the labels and at the same time allows the use of multiple charsets to be easily incorporated in a standardized manner. It causes no harm to the current DNS because it embraces the original format for DNS laid out in [RFC1035](#), complemented with the ideas incorporated in EDNS [[RFC2671](#)].

## [2.](#) DNSII Protocol

The DNSII Protocol consists mainly of two parts: the InPacket DNSII Identifier and the InPacket Label Encoding Type. In addition, there are several special considerations for specific record types.

### [2.1](#) InPacket DNSII Identifier

In the DNSII specifications, an InPacket DNSII Identifier MUST be inserted before a label to signify that it contains extended characters that are not supported by the current DNS.

This DNSII flag, which is the first two bits of a label, effectively distinguishes a DNSII compliant request from the existing format, without having to conduct a guess from a name check whether the incoming packet is multilingual aware. This is a substantial improvement over character encoding schemes and multilingual implementations in which it is almost impossible to determine the language of an incoming request. The DNSII flag makes the process clear and simple.

Currently:

"00" regular label [[RFC1035](#)]

"11" a redirection for DNS compression [[RFC1035](#)]

"01" indicates the use of EDNS for multiple UDP packets [[RFC2671](#)]

DNSII calls for the use of the bit sequence "10" to identify that the querying node is DNSII aware. This will mean that all the possible variations at top two label bits will be used. Therefore, in consideration, following two bits MUST be reserved for future

flagging use. The 2 bits SHOULD be arbitrarily set to "00". This effectively opens up 3 more possible implementations for future enhancements.

The motivation for this approach is the belief there should be no ambiguity in name resolution. Any name that the client wishes to resolve, should resolve, regardless of the client side-encoding scheme.

## 2.2 InPacket Label Encoding Type (ILET)

Immediately following the 2 assigned DNSII flag and the 2 reserved bits are 12 bits assigned to determine the InPacket Label Encoding Type (ILET).

The ILET is a 12-bit number that is used to determine the encoding scheme used by the characters of the label. The MIBenum numbers [\[RFC1700\]](#) SHOULD be used in this field. The allocation of 12 bits aligns perfectly with the MIBenum specification, of which the value goes up to over 2200. With 12 bits, the total possible values would be 4096 (with 11 bits, the largest value that can be represented is only 2047, slightly short of the specification). The reason for the adoption of MIBenum is to make use of the existing list of encoding numbering schemes rather than re-inventing the wheel.

The value in the ILET field SHOULD only be allowed for the valid encoding schemes defined in the MIBenum list. After identifying the encoding type, the regular count-label scheme of the DNS resumes. The resulting label should look like this:

```

          1 1 1 1 1 1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+---+---+---+---+---+
| 1 0 | z |           ILET           |
+---+---+---+---+---+---+---+
|      COUNT      | characters... |
+---+---+---+---+---+---+

```

To minimize the size of a DNS packet, if the entire label is constituted in characters only from the ANSI table, the DNS label will appear identical to current implementations. The first two bits will remain "00".

For example, using the DNSII format the label for "dns" MAY be represented as:

```

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 1 0 | 0 0 | 0 0 0 0 0 0 0 0 0 0 0 0 1 1 | MIBenum 3 = ANSI

```

```

+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           3           |           6           4           | "d"=64
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           6           E           |           7           3           | "n"=6E  "s"=73
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Or, the same domain label "dns" MAY also be represented as:

```

    0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           3           |           d           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           n           |           s           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

With a multilingual domain name ns. Éì p© ´ .tld as an example:

```

          1 1 1 1 1 1          1 1 1 1 1 1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----+-----+-----+-----+-----+-----+-----+-----+
|1 0| z |          ANSI=3          |          2          |          n          |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          s          |1 0|0 0|          UCS-2=1000          |          4          |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          (U+57DF)          |          Éì          (U+540D)          |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          p©          (U+7CFB)          |          ´          (U+7D71)          |
+-----+-----+-----+-----+-----+-----+-----+-----+
|0 0|          3          |          t          |          l          |          d          |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          0          |
+-----+

```

From the above example, we can see that the DNSII format is used for the first label "ns", as well as for the second label, which is in Chinese (the MIBenum for UCS-2 or ISO 10646 [Unicode] is 1000). The third label "tld" however uses the current format.

In any case, the count-label-count-label mechanism is largely preserved. Especially in the case of extended characters where in other proposals, the "count" no longer represents the character count. In the above example, the domain is still represented as 2ns4 Éì p© ´ 3t ld0, exactly in line with the original specifications.

Note that the first label in any query could be represented in DNSII format to alert the destination server that it is DNSII aware. This is not required but is specifically configured for the considerations with CNAME, A6, DNAME and PTR records.

This approach is used to ensure that there is no confusion about the encoding format of the label. ILET allows the capability of employing all existing encoding schemes (UTF-7, UTF-8, ISO 10646 [UCS-2], ISO 10646 [UCS-4]). ILET also allows the flexibility of employing future encoding schemes.

### [2.3 The Rationale for using ILET](#)

Besides being able to preserve the count-label-count-label structure, which in itself is actually a very important part because of the problematic non-uniform byte encoding schemes, the use of ILET aligns perfectly with previous IETF specifications as well as beneficial for tricky case folding and canonicalization issues.

We know that all protocols MUST identify, for all character data,

which charset is in use [[RFC2277](#)], therefore it is necessary to specify whatever encoding scheme, whether it be UTF-8, UTF-7, 16-bit UCS-2 or ISO 8859 that is being used. In essence, we understand that

it is paramount that a charset be clearly identified, especially in situation like the DNS where no direct communication is established.

At times and in specific cases, language information may be required to achieve a particular level of quality for the purpose of displaying a text stream. For example, UTF-8 encoded Han may require transmission of a language tag to select the specific glyphs to be displayed at a particular level of quality.

Note that information other than language may be used to achieve the required level of quality in a display process. In particular, a font tag is sufficient to produce identical results. However, the association of a language with a specific block of text has usefulness far beyond its use in display. In particular, as the amount of information available in multiple languages on the World Wide Web grows, it becomes critical to specify which language is in use in particular documents, to assist automatic indexing and retrieval of relevant documents.\_ [[RFC2130](#)]

In effect, this means for different languages, it is beneficial to be able to identify the language in order to perform specific functions to the characters, including case folding. With ILET, the local encoding scheme could be used and with them there are well defined folding methods. Therefore, the use of ILET enables an optimized folding mechanism brought about by the preservation of local encoding schemes, which is otherwise very difficult or virtually impossible to do if only UTF-8 is used.

For the DNS however, a language tag is less feasible because if a name is consisted of multiple languages, it would be very difficult for tagging to be performed. The possibility of having multiple languages is very sound, and is used frequently as trademarks around the world. For example the famous Toys"İ"Us name, uses a character from the Cyrillic language set.

## **2.4 Considerations for Specific Requests**

For certain requests, an ANSI only name could result in a multilingual domain as an answer. These include PTR, CNAME, A6 and DNAME requests. Special considerations are made within the DNSII protocol to make sure that non-DNSII aware servers will not be fed with a DNSII format packet.

### **2.4.1 PTR Records**

For all PTR requests, the first label of the query MUST use DNSII format to alert the destination server. Upon which, a DNSII packet will be replied should the name contain extended characters.



If the DNSII format is not used, and the PTR record stumbles upon a multilingual domain name, one of the following responses SHOULD be given:

- a. The implementer of DNSII MAY chose to reject the request;
- or
- b. An ACE format domain with a "for.ref.only" suffix MAY be returned;
- or
- c. A DNSII compliant server MAY return an 8-bit format of the requested domain.

Since the PTR record is usually used for display purposes only, the rejection (the IP address will then be used) or ACE format is acceptable. If the response is however used for further resolution, an ACE format MUST not be used.

#### **2.4.2 CNAME, A6 & DNAME**

For queries concerning the record types CNAME, A6 or DNAME, a DNSII aware server should first check to see if the incoming request is DNSII compliant (flagged by the "10" bits in the first label):

If so, and the domain to be returned includes extended characters, the response SHOULD be in DNSII format.

If not, any multilingual domains returned should be in an 8 bit form.

For the above record types it is strongly RECOMMENDED not to associate an alphanumeric label to a multilingual label as the RDATA. However, it is permissible to associate a multilingual label with an alphanumeric label as the RDATA.

### **3. Alternate Implementations**

The DNSII-MDNP is intended to be a framework for the implementation of multilingual domain names. While the core concepts and the design principles remain consistent, it is possible to contemplate alternative implementations. The four alternatives introduced here include the arbitrary restriction of ILET values, the reduction of ILET bit allocation for reflecting ISO 10646 transformations only, and finally two implementations using DNSII over EDNS.

#### **3.1 Restricted ILET Values**

One possible implementation guideline is for the ILET to be restricted to values only representing ISO 10646 transformations including UCS-2, UCS-4, UTF-7, UTF-8, UTF-16 and other as they become

available and included as a standard MIBenum.

Although this takes away some of the benefits of keeping the local encoding scheme which includes the issues of case folding, canonicalization and other related concerns, it creates a system that on one hand contains only encoding schemes from ISO 10646, but on the other hand still provides the flexibility of deploying new encoding schemes that stem from ISO 10646, such as the 32-bit format that is due to be used soon.

We understand it is specified that in protocols, which up to now have used US-ASCII only, UTF-8 forms a simple upgrade path; however, its use should be negotiated either by negotiating a protocol version or by negotiating charset usage, and a fallback to UTF-7 MUST be available. [[RFC2130](#)] With DNSII, the required fallback to UTF-7 could easily be done by setting the ILET value to reflect UTF-7.

### 3.2 Reduced ILET Bit Allocation

Furthering the restriction of the ILET to ISO 10646 transformations only, the ILET bit allocations could also be reduced from 12 bit to 5 bit. This successfully creates a total of 32 possible values. The reserved bits are also reduced to one.

```

          1 1 1 1 1 1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+-----+-----+
|1 0|z| ILET  |   COUNT   |
+-----+-----+
| characters... |
+-----+
```

For example, the label " Éì þ© ´ " will now be reflected in DNS packets in the following form:

```

          1 1 1 1 1 1          1 1 1 1 1 1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----+-----+-----+-----+-----+
|1 0|z| ILET=1 |   4   |           (U+57DF)           |
+-----+-----+-----+-----+
|           Éì   (U+540D)           |           þ©   (U+7CFB)           |
+-----+-----+-----+-----+
|           ´   (U+7D71)           |
+-----+-----+
```

To start off with, the ILET values MAY be determined as follows:

0 = reserved for ANSI	1 = UTF-7
2 = UCS-2	3 = UTF-8
4 = UCS-4	5 = UTF-16
6 = 6 octets per character	7 = 7 octets per character

8 = UCS-8 (8 octets per character) 9 = UTF-31  
etc.

The ILET number will be the number of octets that should be read each time, with exceptions at 0,3,5,9 or any number that is just after a full UCS. ILET=3 corresponds to UTF8 because ILET=2 = UCS2 and UTF-8 is a compatibility transformation for UCS-2 (in 16bit) in 8bit. A possible future expansion to UCS-8 is also included to make the schema truly future prove.

This arrangement opens up opportunity and versatility of the use of private schemes that make use of odd byte length characters or symbols such as 6, 7 or even 18octet representations without the need for the DNS to update or adjust to, while maintaining the integrity and unique nature of domain names.

### 3.3 DNSII over EDNS

While DNSII and EDNS could coexist, it is possible to implement the DNSII concept onto an EDNS based platform. It is however preferable to use both EDNS and the DNSII scheme together as described in [Section 6](#), because EDNS(0) compliant servers may have trouble when the label count exceeds the value of 63 (and smaller than 127) because then, the EDNS mechanism MAY be reiterated.

Nevertheless, it is possible to utilize EDNS to act as a DNSII Identifier. The short-comings and pit-falls are however further laid in another draft DNSII-EDNS.

#### 3.3.1 DNSII Identifier using EDNS

EDNS could simply be used in place of the DNSII Identifier. Assuming that the reduced ILET values introduced in [Section 3.2](#) is used, the ILET will fit nicely into one octet immediately following the ELT portion. The resulting representation for the domain "host. Èì p© ´.tld" would be as follows:

```

                                1 1 1 1 1 1
                                1 1 1 1 1 1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----+
20|0 0|      4      |      h      |      o      |      s      |
+-----+
|      t      |0 1|0 0 0 0 1 0| ILET=UCS-2=2 |      4      |
+-----+
|      (U+57DF)      |      Èì      (U+540D)      |
+-----+
|      p©      (U+7CFB)      |      ´      (U+7D71)      |
+-----+
|0 0|      3      |      t      |      l      |      d      |
+-----+
|      0      |

```

+-----+

The OPT RR could be used not only for version control but also as a notification for the destination server on the level of conformance of the inquirer. This is especially beneficial when considering the issues raised in [Section 2.4](#) where ANSI only requests my result in a multilingual response.

Proposed identifications within the OPT RR (used in this document for discussion purposes):

ELT = 0b000010

EDNS-VERSION = 2 (DNSII)

OPTION-CODE = 1 (IDN)

OPTION-DATA = conformance level (defined in DNSII-MDNR [Section 4](#))

Plus other information if required

The conformance level SHOULD be included in the first octet of the OPTION-DATA field and reflect the value corresponding to:

BASIC conformance = 1

INTERMEDIATE conformance = 2

FULL conformance = 3

A resulting DNSII OPT RR from a fully compliant inquirer SHOULD be in the form:

										1 1 1 1 1 1																1 1 1 1 1 1													
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5								
NAME=0										TYPE = OPT = 41										Sender's UDP																			
Payload Size										EXTENDED-RCODE=0										EDNS-VERSION=2										z									
z										RDLENGTH = 5										OPTION-CODE =																			
IDN = 1										OPTION-LENGTH = 1										Conformance=3																			

### [3.3.2](#) ILET using EDNS OPT RRs

Instead of using the OPT RR only as a notification of DNSII awareness, it utilized to indicate the type of encoding that is being used in the labels. In other words, the ILET value could be included in the OPT RR instead of within the label.

The ILET value will be included right after the conformance level octet in the OPTION-DATA field within the OPT RR RDATA.





The resulting QNAME labels and OPT RR for the domain "www. Éì þ© ´ .tld" from a fully compliant inquirer sending the name in UCS-2 would become:

```

          1 1 1 1 1 1          1 1 1 1 1 1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----+-----+-----+-----+-----+-----+-----+-----+
| 0 0 |      3      |      w      |      w      |      w      |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 0 1 | 0 0 0 0 1 0 |      4      |      (U+57DF)      |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Éì      (U+540D)      |      þ©      (U+7CFB)      |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      ´      (U+7D71)      | 0 0 |      3      |      t      |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      1      |      d      |      0      |
+-----+-----+-----+-----+-----+-----+-----+

```

(OPT RR containing ILET information)

```

:
/
+-----+-----+-----+-----+-----+-----+-----+-----+
|  NAME=0      |  TYPE = OPT = 41      |  Sender's UDP  |
+-----+-----+-----+-----+-----+-----+-----+-----+
|  Payload Size  | EXTENDED RCODE=0 | EDNS-VERSION=2 |      z      |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      z      |  RDLENGTH = 9      |  OPTION-CODE =  |
+-----+-----+-----+-----+-----+-----+-----+-----+
|  IDN = 1      |  OPTION-LENGTH = 4      |  Conformance=3  |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      1      |      0      |      0      |      0      |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Note that instead of allocating only 12 bits for the ILET, the MIBenum value is expressed over 4 octets with each octet carrying a numeric value. Since UCS-2 is used and the MIBenum value for UCS-2 = 1000, the 4 octets corresponds to the values 1, 0, 0, 0 respectively.

If the reduced ILET values are used, only 1 octet is required to reflect the encoding scheme being used.

#### 4. Implementation & Deployment Strategies

The first step in any multilingual domain name implementation should be to encourage an 8-bit clean approach to DNS. However, even when the system is 8-bit clean the problem with conflicting characters still exists. This is where the DNSII protocol becomes most

valuable.

Although the DNSII protocol could be implemented at any level of the DNS, the following phased rollout is contemplated.

(4) Root Level - Eventually, as the DNSII is proven to be stable and beneficial for the Internet at large, it could be used in the Root Level so that new multilingual TLDs could be created.

	1 1 1 1 1 1															1 1 1 1 1 1																
	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5
20	0 1		ELT					0 0		3				d					n													
			s					1 0		0 0				UCS-2=1000					4													
								(U+57DF)							Éì					(U+540D)												
			p©					(U+7CFB)												(U+7D71)												
	0 0		3							t				l					d													
			0																													

+-----+

```

:                                     :
/                                     /
+-----+
|0 0|   4   |   h   |   o   |   s   |
+-----+
|   t   |1 1|   21   |
+-----+

```

## 7. Intellectual Property Considerations

It is the intention of Neteka to submit the DNSII protocol and other elements of the multilingual domain name server software to IETF for review, comment or standardization.

Neteka Inc. has applied for one or more patents on the technology related to multilingual domain name server software and multilingual email server software suite. If a standard is adopted by IETF and any patents are issued to Neteka with claims that are necessary for practicing the standard, any party will be able to obtain the right to implement, use and distribute the technology or works when implementing, using or distributing technology based upon the specific specifications under fair, reasonable and non-discriminatory terms.

Other DNSII related documents and discussions could be found at <http://www.dnsii.org>.

## 8. References

- [DNSII-MDNR] E. Chung, D. Leung, \_DNSII Multilingual Domain Name Resolution\_, August 2000
- [RFC1700] J. Reynolds, J. Postel, "ASSIGNED NUMBERS", [RFC 1700](#), October 1994.
- [ISO10646] ISO/IEC 10646-1:2000. International Standard - Information technology -- Universal Multiple-Octet Coded Character Set (UCS)
- [RFC1034] Mockapetris, P., "Domain Names - Concepts and Facilities," STD 13, [RFC 1034](#), USC/ISI, November 1987
- [RFC1035] Mockapetris, P., "Domain Names - Implementation and Specification," STD 13, [RFC 1035](#), USC/ISI, November 1987
- [RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels," [RFC 2119](#), March 1997
- [RFC2130] C. Weider, et al. \_The Report of the IAB Character Set

Workshop held 29 February - 1 March, 1996\_ [RFC 2130](#),  
April 1997

- [RFC2277] H. Alvestrand, "\_IETF Policy on Character Sets and Languages\_" [RFC 2277](#), January 1998
- [RFC2671] Paul Vixie, "Extension Mechanisms for DNS (EDNS0)", August 1999, [RFC 2671](#).

Authors:

Edmon Chung  
Neteka Inc.  
2462 Yonge St. Toronto,  
Ontario, Canada M4P 2H5  
edmon@neteka.com

David Leung  
Neteka Inc.  
2462 Yonge St. Toronto,  
Ontario, Canada M4P 2H5  
david@neteka.com



