

Network Working Group  
Request for Comments: 1664  
Category: Experimental

C. Allocchio  
A. Bonito  
GARR-Italy  
B. Cole  
Cisco Systems Inc.  
S. Giordano  
Centro Svizzero Calcolo Scientifico  
R. Hagens  
Advanced Network & Services  
August 1994

## Using the Internet DNS to Distribute RFC1327 Mail Address Mapping Tables

### Status of this Memo

This memo defines an Experimental Protocol for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

### Abstract

This memo defines how to store in the Internet Domain Name System the mapping information needed by e-mail gateways and other tools to map [RFC822](#) domain names into X.400 O/R names and vice versa. Mapping information can be managed in a distributed rather than a centralised way. Gateways located on Internet hosts can retrieve the mapping information querying the DNS instead of having fixed tables which need to be centrally updated and distributed. This memo is a joint effort of X400 operation working group (x400ops) and RARE Mail and Messaging working group (WG-MSG).

### 1. Introduction

The connectivity between the Internet SMTP mail and other mail services, including the Internet X.400 mail and the commercial X.400 service providers, is assured by the Mail eXchanger (MX) record information distributed via the Internet Domain Name System (DNS). A number of documents then specify in details how to convert or encode addresses from/to [RFC822](#) style to the other mail system syntax. However, only conversion methods provide, via some algorithm or a set of mapping rules, a smooth translation, resulting in addresses indistinguishable from the native ones in both [RFC822](#) and foreign world.

[RFC1327](#) describes a set of mappings which will enable interworking between systems operating the CCITT X.400 (1984/88) Recommendations

and systems using the [RFC822](#) mail protocol, or protocols derived from [RFC822](#). That document addresses conversion of services, addresses, message envelopes, and message bodies between the two mail systems. This document is concerned with one aspect of [RFC1327](#): the mechanism for mapping between X.400 O/R addresses and [RFC822](#) domain names. As described in [Appendix F of RFC1327](#), implementation of the mappings requires a database which maps between X.400 O/R addresses and domain names, and this database is statically defined.

This approach requires many efforts to maintain the correct mapping: all the gateways need to get coherent tables to apply the same mappings, the conversion tables must be distributed among all the operational gateways, and also every update needs to be distributed. This static mechanism requires quite a long time to be spent modifying and distributing the information, putting heavy constraints on the time schedule of every update. In fact it does not appear efficient compared to the Internet Domain Name Service (DNS). More over it does not look feasible to distribute the database to a large number of other useful applications, like local address converters, e-mail User Agents or any other tool requiring the mapping rules to produce correct results.

A first proposal to use the Internet DNS to store, retrieve and maintain those mappings was introduced by two of the authors (B. Cole and R. Hagens) adopting two new DNS resource record (RR) types: TO-X400 and TO-822. This new proposal adopts a more complete strategy, and requires one new RR only. The distribution of the [RFC1327](#) mapping rules via DNS is in fact an important service for the whole Internet community: it completes the information given by MX resource record and it allows to produce clean addresses when messages are exchanged among the Internet [RFC822](#) world and the X.400 one (both Internet and Public X.400 service providers).

A first experiment in using the DNS without expanding the current set of RR and using available ones was in the mean time deployed by some of the authors. The existing PTR resource records were used to store the mapping rules, and a new DNS tree was created under the ".it" top level domain. The result of the experiment was positive, and a few test applications ran under this provisional set up. This test was also very useful in order to define a possible migration strategy during the deployment of the new DNS containing the new RR. The Internet DNS nameservers wishing to provide this mapping information need in fact to be modified to support the new RR type, and in the real Internet, due to the large number of different implementations, this takes some time.

The basic idea is to adopt a new DNS RR to store the mapping information. The [RFC822](#) to X.400 mapping rules (including the so

called 'gate' rules) will be stored in the ordinary DNS tree, while the definition of a new branch of the name space defined under each national top level domain is envisaged in order to contain the X.400 to [RFC822](#) mappings. A "two-way" mapping resolution schema is thus fully implemented.

The creation of the new domain name space representing the X.400 O/R names structure also provides the chance to use the DNS to distribute dynamically other X.400 related information, thus solving other efficiency problems currently affecting the X.400 MHS service.

In this paper we will adopt the [RFC1327](#) mapping rules syntax, showing how it can be stored into the Internet DNS.

### [1.1](#) Definitions syntax

The definitions in this document is given in BNF-like syntax, using the following conventions:

```
|   means choice
\   is used for continuation of a definition over several lines
[]  means optional
{}  means repeated one or more times
```

The definitions, however, are detailed only until a certain level, and below it self-explaining character text strings will be used.

## [2.](#) Motivation

Implementations of [RFC1327](#) gateways require that a database store address mapping information for X.400 and [RFC822](#). This information must be disseminated to all [RFC1327](#) gateways. In the Internet community, the DNS has proven to be a practical mean for providing a distributed name service. Advantages of using a DNS based system over a table based approach for mapping between O/R addresses and domain names are:

- It avoids fetching and storing of entire mapping tables by every host that wishes to implement [RFC1327](#) gateways and/or tools
- Modifications to the DNS based mapping information can be made available in a more timely manner than with a table driven approach.
- It allows full authority delegation, in agreement with the Internet regionalization process.

- Table management is not necessarily required for DNS-based [RFC1327](#) gateways.
- One can determine the mappings in use by a remote gateway by querying the DNS (remote debugging).

Also many other tools, like address converters and User Agents can take advantage of the real-time availability of [RFC1327](#) tables, allowing a much easier maintenance of the information.

### 3. The domain space for X.400 O/R name addresses

Usual domain names (the ones normally used as the global part of an [RFC822](#) e-mail address) and their associated information, i.e., host IP addresses, mail exchanger names, etc., are stored in the DNS as a distributed database under a number of top-level domains. Some top-level domains are used for traditional categories or international organisations (EDU, COM, NET, ORG, INT, MIL...). On the other hand any country has its own two letter ISO country code as top-level domain (FR, DE, GB, IT, RU, ...), including "US" for USA. The special top-level/second-level couple IN-ADDR.ARPA is used to store the IP address to domain name relationship. Our proposal defines in the above structure the appropriate way to locate the X.400 O/R name space, thus enabling us to store in DNS the [RFC1327](#) mapping data.

The [RFC1327](#) mapping information is composed by three tables: 'table1' gives the translation from X.400 to [RFC822](#) while 'table2' and 'gate' tables map [RFC822](#) into X.400. Each mapping table is composed by mapping rules, and a single mapping rule is composed by a keyword (the argument of the mapping function derived from the address to be translated) and a translator (the mapping function parameter):

keyword#translator#

the '#' sign is a delimiter enclosing the translator. An example:

foo.bar.us#PRMD\$foo\bar.ADM\$intx.C\$us#

Local mappings are not intended for use outside their restricted environment, thus they should not be included in DNS. If local mappings are used, they should be stored using static local tables, exactly as local static host tables can be used with DNS.

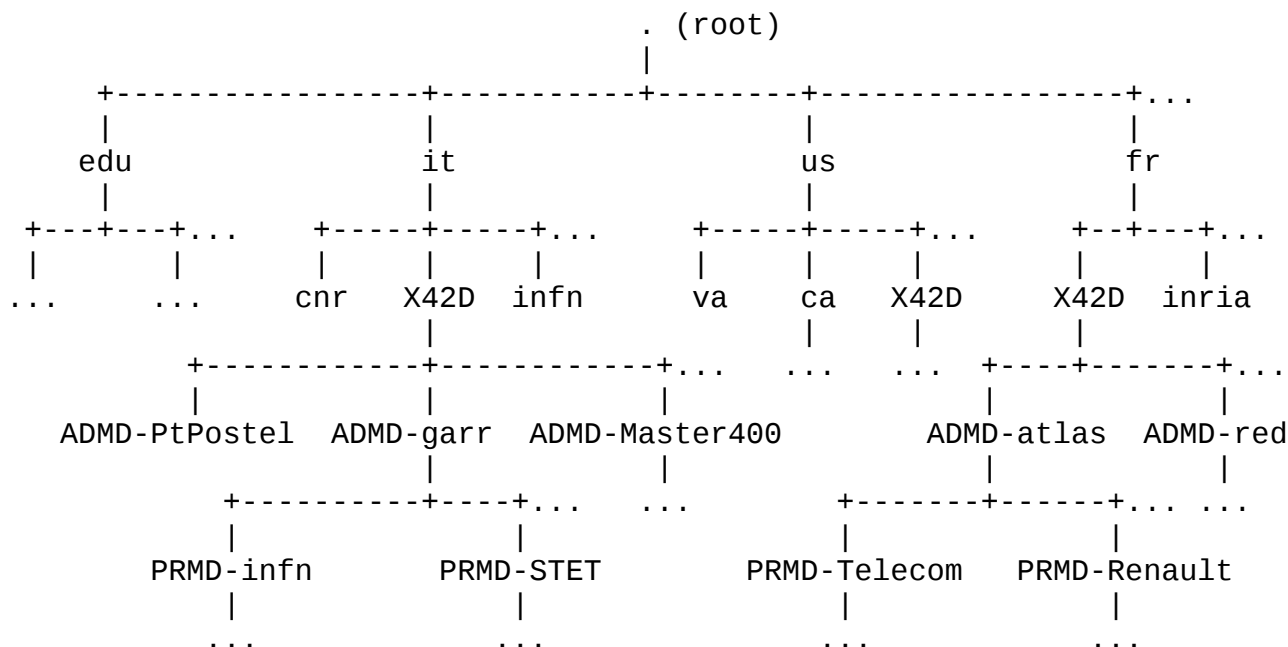
The keyword of a 'table2' and 'gate' table entry is a valid [RFC822](#) domain; thus the usual domain name space can be used without problems to store these entries.

On the other hand, the keyword of a 'table1' entry belongs to the X.400 O/R name space. The X.400 O/R name space does not usually fit into the usual domain name space, although there are a number of similarities; a new name structure is thus needed to represent it. This new name structure contains the X.400 mail domains.

To ensure the correct functioning of the DNS system, the new X.400 name structure must be hooked to the existing domain name space in a way which respects the existing name hierarchy.

A possible solution was to create another special branch, starting from the root of the DNS tree, somehow similar to the in-addr.arpa tree. This idea would have required to establish a central authority to coordinate at international level the management of each national X.400 name tree, including the X.400 public service providers. This coordination problem is a heavy burden if approached globally. Moreover the X.400 name structure is very 'country oriented': thus while it requires a coordination at national level, it does not have concepts like the international root. In fact the X.400 international service is based on a large number of bilateral agreements, and only within some communities an international coordination service exists.

The X.400 two letter ISO country codes, however, are the same used for the [RFC822](#) country top-level domains and this gives us an appropriate hook to insert the new branches. Our proposal is, in fact, to create under each national top level ISO country code a new branch in the name space. This branch represents exactly the X.400 O/R name structure as defined in each single country, following the ADMD, PRMD, O, OU hierarchy. A unique reserved label 'X42D' is placed under each country top-level domain, and hence the national X.400 name space derives its own structure:



The creation of the X.400 new name tree at national level solves the problem of the international coordination. Actually the coordination problem is just moved at national level, but it thus becomes easier to solve. The coordination at national level between the X.400 communities and the Internet world is already a requirement for the creation of the national static [RFC1327](#) mapping tables; the use of the Internet DNS gives further motivations for this coordination.

The coordination at national level also fits in the ongoing proposal intended to define exactly the [RFC1327](#) Mapping Authorities. The DNS in fact allows a step by step authority distribution, up to a final complete delegation, which can be easily controlled at national level accordingly with national needs and situations. A further advantage of the national based solution is to allow each country to set up its own X.400 name structure in DNS and to deploy its own authority delegation according to its local time scale and requirements, with no loss of global service in the mean time. And last, placing the new X.400 name tree and coordination process at national level fits into the Internet regionalization and internationalisation process, as it requires local bodies to take care of local coordination problems.

The DNS name space thus contains completely the information required by an e-mail gateway or tool to perform the X.400-RFC822 mapping: a simple query to the nearest nameserver provides it. Moreover there is no more any need to store, maintain and distribute manually any mapping table. The new X.400 name space can also contain further information about the X.400 community, as DNS allows for it a

complete set of resource records, and thus it allows further developments. This set of RRs in the new X.400 name space must be considered 'reserved' and thus not used until further specifications.

The construction of the new domain space trees will follow the same procedures used when organising at first the already existing DNS space: at first the information will be stored in a quite centralised way, and distribution of authority will be gradually achieved. A separate document will describe the implementation phase and the methods to assure a smooth introduction of the new service.

#### 4. The new DNS resource record for [RFC1327](#) mapping rules: PX

The specification of the Internet DNS ([RFC1035](#)) provides a number of specific resource records (RRs) to contain specific pieces of information. In particular they contain the Mail eXchanger (MX) RR and the host Address (A) records which are used by the Internet SMTP mailers. As we will store the [RFC822](#) to X.400 mapping information in the already existing DNS name tree, we need to define a new DNS RR in order to avoid any possible clash or misuse of already existing data structures. The same new RR will also be used to store the mappings from X.400 to [RFC822](#). More over the mapping information, i.e., the [RFC1327](#) mapping rules, has a specific format and syntax which require an appropriate data structure and processing. A further advantage of defining a new RR is the ability to include flexibility for some eventual future development.

The definition of the new 'PX' DNS resource record is:

```
class:      IN    (Internet)

name:       PX    (pointer to X.400/RFC822 mapping information)

value:      26
```

The PX RDATA format is:

```
+---+---+---+---+---+---+---+---+---+---+---+---+
|                                     |
|               PREFERENCE           |
|                                     |
+---+---+---+---+---+---+---+---+---+---+---+---+
/                               /
/                               /
+---+---+---+---+---+---+---+---+---+---+---+---+
/                               /
/                               /
+---+---+---+---+---+---+---+---+---+---+---+---+
```

where:



- PREFERENCE    A 16 bit integer which specifies the preference given to this RR among others at the same owner. Lower values are preferred;
- MAP822        A <domain-name> element containing <[rfc822](#)-domain>, the [RFC822](#) part of the [RFC1327](#) mapping information;
- MAPX400       A <domain-name> element containing the value of <x400-in-domain-syntax> derived from the X.400 part of the [RFC1327](#) mapping information (see sect. 4.2);

PX records cause no additional section processing. The PX RR format is the usual one:

<name> [<class>] [<TTL>] <type> <RDATA>

When we store in DNS a 'table1' entry, then <name> will be an X.400 mail domain name in DNS syntax (see sect. 4.2). When we store a 'table2' or a 'gate' table entry, <name> will be an [RFC822](#) mail domain name, including both fully qualified DNS domains and mail only domains (MX-only domains). All normal DNS conventions, like default values, wildcards, abbreviations and message compression, apply also for all the components of the PX RR. In particular <name>, MAP822 and MAPX400, as <domain-name> elements, must have the final "." (root) when they are fully qualified.

#### [4.1](#) Additional features of the PX resource record

The definition of the RDATA for the PX resource record, and the fact that DNS allows a distinction between an exact value and a wildcard match for the <name> parameter, represent an extension of the [RFC1327](#) specification for mapping rules. In fact, any [RFC1327](#) mapping table entry is an implicit wildcard entry, i.e., the rule

net2.it#PRMD\$net2.ADM\$P400.C\$it#

covers any [RFC822](#) domain ending with 'net2.it', unless more detailed rules for some subdomain in 'net2.it' are present. Thus there is no possibility to specify explicitly an [RFC1327](#) entry as an exact match only rule. In DNS an entry like

\*.net2.it.    IN   PX   10    net2.it.   PRMD-net2.ADM-p400.C-it.

specify the usual wildcard match as for [RFC1327](#) tables. However an entry like

ab.net2.it.   IN   PX   10    ab.net2.it.   0-ab.PRMD-net2.ADMdb.C-it.



is valid only for an exact match of 'ab.net2.it' [RFC822](#) domain.

Note also that in DNS syntax there is no '#' delimiter around MAP822 and MAPX400 fields: the syntax defined in sect. 4.2 in fact does not allow the <blank> (ASCII decimal 32) character within these fields, making unneeded the use of an explicit delimiter as required in the [RFC1327](#) original syntax.

Another extension to the [RFC1327](#) specifications is the PREFERENCE value defined as part of the PX RDATA section. This numeric value has exactly the same meaning than the similar one used for the MX RR. It is thus possible to specify more than one single mapping for a domain (both from [RFC822](#) to X.400 and vice versa), giving as the preference order. In [RFC1327](#) static tables, however, you cannot specify more than one mapping per each [RFC822](#) domain, and the same restriction apply for any X.400 domain mapping to an [RFC822](#) one.

More over, in the X.400 recommendations a note suggests than an ADMD=<blank> should be reserved for some special cases. Various national functional profile specifications for an X.400 MHS states that if an X.400 PRMD is reachable via any of its national ADMDs, independently of its actual single or multiple connectivity with them, it should use ADMD=<blank> to advertise this fact. Again, if a PRMD has no connections to any ADMD it should use ADMD=0 to notify its status, etc. However, in most of the current real situations, the ADMD service providers do not accept messages coming from their subscribers if they have a blank ADMD, forcing them to have their own ADMD value. In such a situation there are problems in indicating properly the actually working mappings for domains with multiple connectivity. The PX RDATA 'PREFERENCE' extension was introduced to take in consideration these problems.

However, as these extensions are not available with [RFC1327](#) static tables, it is strongly discouraged to use them when interworking with any table based gateway or application. The extensions were in fact introduced just to add more flexibility, like the PREFERENCE value, or they were already implicit in the DNS mechanism, like the wildcard specification. They should be used very carefully or just considered 'reserved for future use'. In particular, for current use, the PREFERENCE value in the PX record specification should be fixed to a value of 50, and only wildcard specifications should be used when specifying <name> values.

#### [4.2](#) The DNS syntax for an X.400 'domain'

The syntax definition of the [RFC1327](#) mapping rules is defined in [appendix F](#) of that document. However that syntax is not very human oriented and contains a number of characters which have a special

meaning in other fields of the Internet DNS. Thus in order to avoid any possible problem, especially due to some old DNS implementations still being used in the Internet, we define a syntax for the X.400 part of any [RFC1327](#) mapping rules (and hence for any X.400 O/R name) which makes it compatible with a <domain-name> element, i.e.,

```
<domain-name> ::= <subdomain> | " "
<subdomain>   ::= <label> | <label> "." <subdomain>
<label>       ::= <alphanum> |
                  <alphanum> {<alphanumhyphen>} <alphanum>
<alphanum>    ::= "0".."9" | "A".."Z" | "a".."z"
<alphanumhyphen> ::= "0".."9" | "A".."Z" | "a".."z" | "-"
```

(see [RFC1035](#), section 2.3.1, page 8). The legal character set for <label> does not correspond to the IA5 Printablestring one used in [RFC1327](#) to define mapping rules. However a very simple "escape mechanism" can be applied in order to bypass the problem. We can in fact simply describe the X.400 part of an [RFC1327](#) mapping rule format as:

```
<map-rule>    ::= <map-elem> | <map-elem> { "." <map-elem> }
<map-elem>    ::= <attr-label> "$" <attr-value>
<attr-label>  ::= "C" | "ADMD" | "PRMD" | "O" | "OU"
<attr-value> ::= " " | "@" | IA5-Printablestring
```

As you can notice <domain-name> and <map-rule> look similar, and also <label> and <map-elem> look the same. If we define the correct method to transform a <map-elem> into a <label> and vice versa the problem to write an [RFC1327](#) mapping rule in <domain-name> syntax is solved.

The [RFC822](#) domain part of any [RFC1327](#) mapping rule is of course already in <domain-name> syntax, and thus remains unchanged.

In particular, in a 'table1' mapping rule the 'keyword' value must be converted into <x400-in-domain-syntax> (X.400 mail DNS mail domain), while the 'translator' value is already a valid [RFC822](#) domain. Vice versa in a 'table2' or 'gate' mapping rule, the 'translator' must be converted into <x400-in-domain-syntax>, while the 'keyword' is already a valid [RFC822](#) domain.

#### [4.2.1](#) IA5-Printablestring to <alphanumhyphen> mappings

The problem of unmatched IA5-Printablestring and <label> character set definition is solved by a simple character mapping rule: whenever an IA5 character does not belong to <alphanumhyphen>, then it is mapped using its 3 digit decimal ASCII code, enclosed in hyphens. A small set of special rules is also defined for the most frequent cases. Moreover some frequent characters combinations used in [RFC1327](#)

rules are also mapped as special cases.

Let's then define the following simple rules:

<a href="#">RFC1327</a> rule	DNS store translation	conditions
<code>&lt;attr-label&gt;\$@</code>	<code>&lt;attr-label&gt;</code>	missing attribute
<code>&lt;attr-label&gt;\$&lt;blank&gt;</code>	<code>&lt;attr-label&gt;"b"</code>	blank attribute
<code>&lt;attr-label&gt;\$xxx</code>	<code>&lt;attr-label&gt;-xxx</code>	elsewhere

Non `<alphanumhyphen>` characters in `<attr-value>`:

<a href="#">RFC1327</a> rule	DNS store translation	conditions
<code>-</code>	<code>-h-</code>	hyphen
<code>\.</code>	<code>-d-</code>	quoted dot
<code>&lt;blank&gt;</code>	<code>-b-</code>	blank
<code>&lt;non A/N character&gt;</code>	<code>&lt;3digit-decimal&gt;-</code>	elsewhere

If the DNS store translation of `<attr-value>` happens to end with an hyphen, then this last hyphen is omitted.

Let's now have some examples:

<a href="#">RFC1327</a> rule	DNS store translation	conditions
<code>PRMD\$@</code>	<code>PRMD</code>	missing attribute
<code>ADMD\$&lt;blank&gt;</code>	<code>ADMDb</code>	blank attribute
<code>ADMD\$400-net</code>	<code>ADMD-400-h-net</code>	hyphen mapping
<code>PRMD\$UK\.</code>	<code>PRMD-UK-d-BD</code>	quoted dot mapping
<code>O\$ACME Inc\.</code>	<code>O-ACME-b-Inc-d</code>	blank & final hyphen
<code>PRMD\$main-400-a</code>	<code>PRMD-main-h-400-h-a</code>	hyphen mapping
<code>O\$-123-b</code>	<code>O--h-123-h-b</code>	hyphen mapping
<code>OU\$123-x</code>	<code>OU-123-h-x</code>	hyphen mapping
<code>PRMD\$Adis+co</code>	<code>PRMD-Adis-043-co</code>	3digit mapping

Thus, an X.400 part from an [RFC1327](#) mapping rule like

`OU$uuu.O$@.PRMD$ppp\.rrr.ADMD$aaa ddd-mmm.C$cc`

translates to

`OU-uuu.O.PRMD-ppp-d-rrr.ADMD-aaa-b-ddd-h-mmm.C-cc`

Another example:

`OU$sales dept\..O$@.PRMD$ACME.ADMD$ .C$GB`

translates to

OU-sales-b-dept-d.O.PRMD-ACME.ADMDB.C-GB

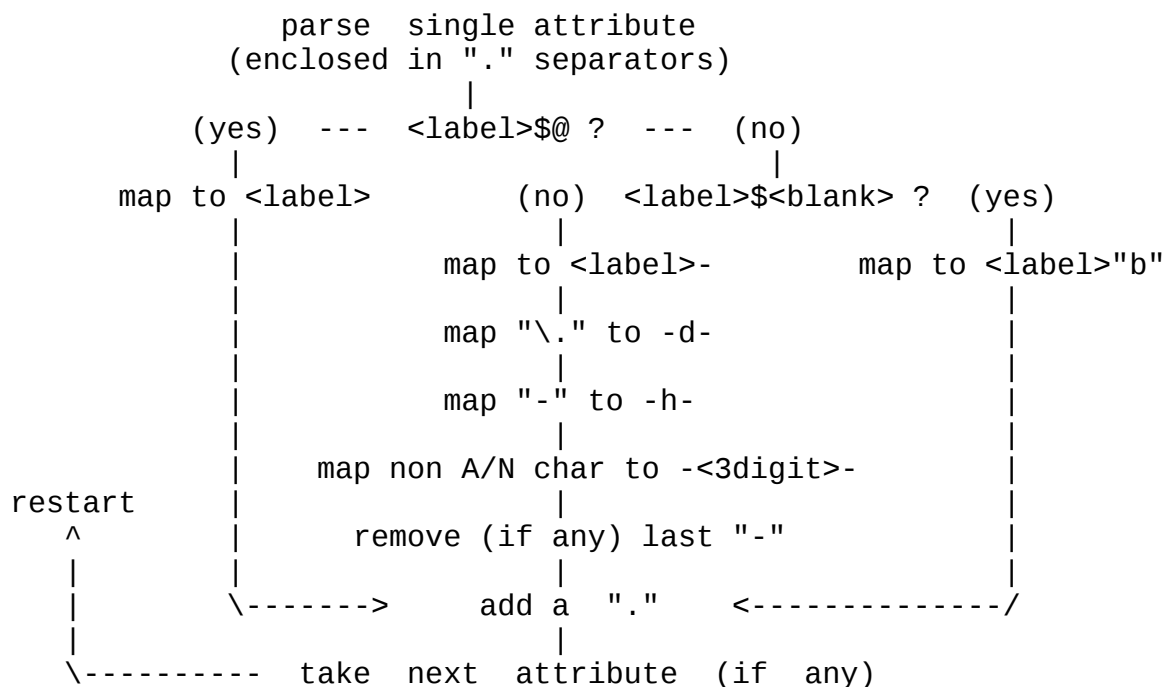
#### 4.2.2 Flow chart

In order to achieve the proper DNS store translations of the X.400 part of an [RFC1327](#) mapping rules or any other X.400 O/R name, some software tools will be used. It is in fact evident that the above rules for converting mapping table from [RFC1327](#) to DNS format (and vice versa) are not user friendly enough to think of a human made conversion.

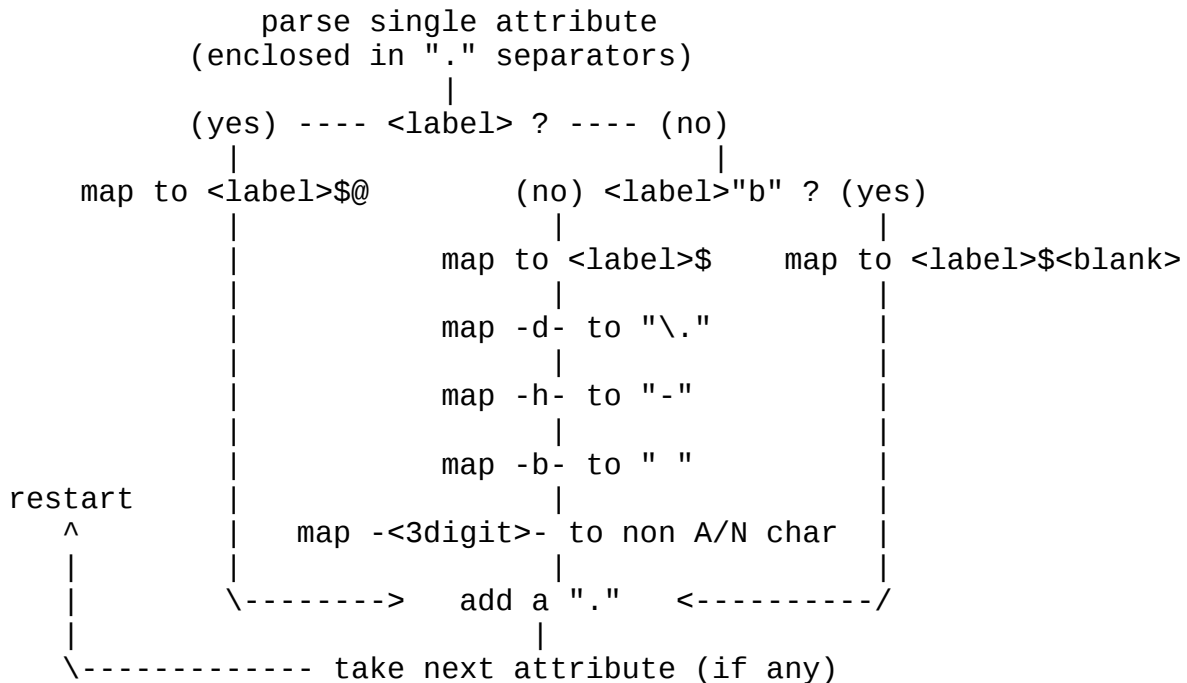
To help in designing such tools, we describe hereunder a small flow chart. The fundamental rule to be applied during translation is, however, the following:

"A string must be parsed from left to right, moving appropriately the pointer in order not to consider again the already translated left section of the string in subsequent analysis."

Flow chart 1 - Translation from [RFC1327](#) to DNS format:



Flow chart 2 - Translation from DNS to [RFC1327](#) format:



Note that the above flow charts deal with the translation of the attributes syntax, only.

#### [4.2.3](#) The Country Code convention in the <name> value.

The [RFC822](#) domain space and the X.400 O/R address space, as said in [section 3](#), have one specific common feature: the X.400 ISO country codes are the same as the [RFC822](#) ISO top level domains for countries. In the previous sections we have also defined a method to write in <domain-name> syntax any X.400 domain, while in [section 3](#) we described the new name space starting at each country top level domain under the X42D.cc (where 'cc' is then two letter ISO country code).

The <name> value for a 'table1' entry in DNS should thus be derived from the X.400 domain value, translated to <domain-name> syntax, adding the 'X42D.cc.' post-fix to it, i.e.,

```
ADMD$acme.C$fr
```

produces in <domain-name> syntax the key:

```
ADMD-acme.C-fr
```

which is post-fixed by 'X42D.fr.' resulting in:

ADMD-acme.C-fr.X42D.fr.

However, due to the identical encoding for X.400 country codes and [RFC822](#) country top level domains, the string 'C-fr.X42D.fr.' is clearly redundant.

We thus define the 'Country Code convention' for the <name> key, i.e.,

"The C-cc section of an X.400 domain in <domain-name> syntax must be omitted when creating a <name> key, as it is identical to the top level country code used to identify the DNS zone where the information is stored".

Thus we obtain the following <name> key examples:

X.400 domain	DNS <name> key
ADMD\$acme.C\$fr	ADMD-acme.X42D.fr.
PRMD\$ux\.av.ADMDB\$.C\$gb	PRMD-ux-d-av.ADMDB.X42D.gb.
PRMD\$ppb.ADMDBDat 400.C\$de	PRMD-ppb.ADMDBDat-b-400.X42D.de.

### [4.3](#) Creating the appropriate DNS files

Using [RFC1327](#)'s assumption of an asymmetric mapping between X.400 and [RFC822](#) addresses, two separate relations are required to store the mapping database: [RFC1327](#) 'table1' and [RFC1327](#) 'table2'; thus also in DNS we will maintain the two different sections, even if they will both use the PX resource record. More over [RFC1327](#) also specify a third table: [RFC1327](#) 'gate' Table. This additional table, however, has the same syntax rules than [RFC1327](#) 'table2' and thus the same translation procedure as 'table2' will be applied; some details about the [RFC1327](#) 'gate' table are discussed in [section 4.4](#).

Let's now check how to create, from an [RFC1327](#) mapping rule entry, the appropriate DNS entry in a DNS data file. We can again define an [RFC1327](#) mapping rule entry as defined in [appendix F](#) of that document as:

<x400-domain>#<[rfc822](#)-domain># (case A: 'table1' entry)

and

<[rfc822](#)-domain>#<x400-domain># (case B: 'table2' and 'gate' entry)

The two cases must be considered separately. Let's consider case A.

- take <x400-domain> and translate it into <domain-name> syntax, obtaining <x400-in-domain-syntax>;
- create the <name> key from <x400-in-domain-syntax> i.e., apply the Country Code convention described in sect. 4.2.3;
- construct the DNS PX record as:

```
*.<name> IN PX 50 <rfc822-domain> <x400-in-domain-syntax>
```

Please note that within PX RDATA the <rfc822-domain> precedes the <x400-in-domain-syntax> also for a 'table1' entry.

an example: from the rule

```
PRMD$ab.ADMMD$ac.C$fr#ab.fr#
```

we obtain

```
*.PRMD-ab.ADMMD-ac.X42D.fr. IN PX 50 ab.fr. PRMD-ab.ADMMD-ac.C-fr.
```

Note that <name>, <rfc822-domain> and <x400-in-domain-syntax> are fully qualified <domain-name> elements, thus ending with a ".".

Let's now consider case B.

- take <rfc822-domain> as <name> key;
- translate <x400-domain> into <x400-in-domain-syntax>;
- construct the DNS PX record as:

```
*.<name> IN PX 50 <rfc822-domain> <x400-in-domain-syntax>
```

an example: from the rule

```
ab.fr#PRMD$ab.ADMMD$ac.C$fr#
```

we obtain

```
*.ab.fr. IN PX 50 ab.fr. PRMD-ab.ADMMD-ac.C-fr.
```

Again note the fully qualified <domain-name> elements.



A file containing the [RFC1327](#) mapping rules and [RFC1327](#) 'gate' table written in DNS format will look like the following fictitious example:

```
!
! RFC1327 table 1: X.400 --> RFC822
!
*.ADMD-acme.X42D.it.          IN PX 50  it. ADMD-acme.C-it.
*.PRMD-accred.ADMD-tx400.X42D.it. IN PX 50  \
                                accred.it. PRMD-accred.ADMD-tx400.C-it.
*.O-u-h-newcity.PRMD-x4net.ADMDb.X42D.it. IN PX 50  \
                                cs.ncty.it. O-u-h-newcity.PRMD-x4net.ADMDb.C-it.
!
! RFC1327 table 2: RFC822 --> X.400
!
*.nrc.it.      IN PX 50  nrc.it. PRMD-nrc.ADMD-acme.C-it.
*.ninp.it.     IN PX 50  ninp.it. O.PRMD-ninp.ADMD-acme.C-it.
*.bd.it.       IN PX 50  bd.it.  PRMD-uk-d-bd.ADMDb.C-it.
!
! RFC1327 Gate Table
!
my.it.  IN PX 50  my.it.  OU-int-h-gw.O.PRMD-ninp.ADMD-acme.C-it.G.
co.it.  IN PX 50  co.it.  O-mhs-h-relay.PRMD-x4net.ADMDb.C-it.G.
```

(here the "\" indicates continuation on the same line, as wrapping is done only due to typographical reasons).

Note the special suffix ".G." on the right side of the 'gate' Table section whose aim is described in [section 4.4](#). The corresponding [RFC1327](#) tables are:

```
#
# RFC1327 table 1: X.400 --> RFC822
#
ADMD$acme.C$it#it#
PRMD$accred.ADMD$tx400.C$it#accred.it#
O$u-newcity.PRMD$x4net.ADMD$ .C$it#cs.ncty.it#
#
# RFC1327 table 2: RFC822 --> X.400
#
nrc.it#PRMD$nrc.ADMD$acme.C$it#
ninp.it#O.PRMD$ninp.ADMD$acme.C$it#
bd.it#PRMD$uk\bd.ADMD$ .C$it#
#
# RFC1327 Gate Table
#
my.it#OU$int-gw.O$@.PRMD$ninp.ADMD$acme.C$it#
co.it#O$mhs-relay.PRMD$x4net.ADMD$ .C$it#
```

#### 4.4 Storing the [RFC1327](#) Gate table

[Section 4.3.4 of RFC1327](#) also specify how an address should be converted between [RFC822](#) and X.400 in case a complete mapping is impossible. To allow the use of DDAs for non mappable domains, the [RFC1327](#) 'gate' table is thus introduced. DNS must store and distribute also these data.

One of the major features of the DNS is the ability to distribute the authority: a certain site runs the "primary" nameserver for one determined sub-tree and thus it is also the only place allowed to update information regarding that sub-tree. This fact allows, in our case, a further additional feature to the table based approach. In fact we can avoid one possible ambiguity about the use of the 'gate' table (and thus of DDAs encoding).

The authority maintaining a DNS entry in the usual [RFC822](#) domain space is the only one allowed to decide if its domain should be mapped using Standard Attributes (SA) syntax or Domain Defined Attributes (DDA) one. If the authority decides that its [RFC822](#) domain should be mapped using SA, then the PX RDATA will be a 'table2' entry, otherwise it will be a 'gate' table entry. Thus for an [RFC822](#) domain we cannot have any more two possible entries, one from 'table2' and another one from 'gate' table, and the action for a gateway results clearly stated.

The [RFC1327](#) 'gate' table syntax is actually identical to [RFC1327](#) 'table2'. Thus the same syntax translation rules from [RFC1327](#) to DNS format can be applied. However a gateway or any other application must know if the answer it got from DNS contains some 'table2' or some 'gate' table information. This is easily obtained flagging with an additional ".G." post-fix the PX RDATA value when it contains a 'gate' table entry. The example in [section 4.3](#) shows clearly the result. As any X.400 O/R domain must end with a country code ("C-xx" in our DNS syntax) the additional ".G." creates no conflicts or ambiguities at all. This postfix must obviously be removed before using the [RFC1327](#) 'gate' table data.

#### 5. Finding [RFC1327](#) mapping information from DNS

The [RFC1327](#) mapping information is stored in DNS both in the normal [RFC822](#) domain name space, and in the newly defined X.400 name space. The information, stored in PX resource records, does not represent a full [RFC822](#) or X.400 O/R address: it is a template which specifies the fields of the domain that are used by the mapping algorithm.

When mapping information is stored in the DNS, queries to the DNS are issued whenever an iterative search through the mapping table would

be performed ([RFC1327: section 4.3.4](#), State I; [section 4.3.5](#), mapping B). Due to the DNS search mechanism, DNS by itself returns the longest possible match in the stored mapping rule with a single query, thus no iteration and/or multiple queries are needed. As specified in [RFC1327](#), a search of the mapping table will result in either success (mapping found) or failure (query failed, mapping not found).

When a DNS query is issued, a third possible result is timeout. If the result is timeout, the gateway operation is delayed and then retried at a later time. A result of success or failure is processed according to the algorithms specified in [RFC1327](#). If a DNS error code is returned, an error message should be logged and the gateway operation is delayed as for timeout. These pathological situations, however, should be avoided with a careful duplication and chaching mechanism which DNS itself provides.

Searching the nameserver which can authoritatively solve the query is automatically performed by the DNS distributed name service.

### [5.1](#) A DNS query example

An [RFC1327](#) mail-gateway located in the Internet, when translating addresses from [RFC822](#) to X.400, can get information about the [RFC1327](#) mapping rule asking the DNS. As an example, when translating the address SUN.CCE.NRC.IT, the gateway will just query DNS for the associated PX resource record. The DNS should contain a PX record like this:

```
*.cce.nrc.it.  IN PX 50   cce.nrc.it.  0-cce.PRMD-nrc.ADM-d-acme.C-it.
```

The first query will return immediately the appropriate mapping rule in DNS store format.

There is no ".G." at the end of the obtained PX RDATA value, thus applying the syntax translation specified in paragraph 4.2 the [RFC1327](#) Table 2 mappingrule will be obtained.

Let's now take another example where a 'gate' table rule is returned. If we are looking for an [RFC822](#) domain ending with top level domain "MW", and the DNS contains a PX record like this,

```
*.mw.  IN  PX  50  mw.  0-cce.PRMD-nrc.ADM-d-acme.C-it.G.
```

DNS will return 'mw.' and '0-cce.PRMD-nrc.ADM-d-acme.C-it.G.', i.e., a 'gate' table entry in DNS store format. Dropping the final ".G." and applying the syntax translation specified in paragraph 4.2 the original rule will be available. More over, the ".G." flag also tells

the gateway to use DDA encoding for the inquired [RFC822](#) domain.

On the other hand, translating from X.400 to [RFC822](#) the address

C=de; ADMD=pkz; PRMD=nfc; O=top;

the mail gateway should convert the syntax according to paragraph 4.2, apply the 'Country code convention' described in 4.2.3 to derive the appropriate DNS translation of the X.400 O/R name and then query DNS for the corresponding PX resource record. The obtained record for which the PX record must be queried is thus:

O-top.PRMD-nfc.ADMD-pkz.X42D.de.

The DNS could contain:

\*.ADMD-pkz.X42D.de. IN PX 50 pkz.de. ADMD-pkz.C-de.

Assuming that there are not more specific records in DNS, the wildcard mechanism will return the [RFC1327](#) 'table1' rule in encoded format.

## 6. Administration of mapping information

The DNS, using the PX RR, will be able to distribute the mapping information to all [RFC1327](#) gateways located on the Internet. However, not all [RFC1327](#) gateways will be able to use the Internet DNS. It is expected that some gateways in a particular management domain will conform to one of the following models:

(a) Table-based, (b) DNS-based, (c) X.500-based

Table-based management domains will continue to submit and retrieve their mapping tables from the International Mapping Table coordinator manually or via some automated procedures. Their mapping information should be made available in DNS by the appropriate DNS authority using the same mechanism already in place for MX records: if a branch has not yet in place its own DNS server, some higher authority in the DNS tree will provide the service for it. A transition procedure similar to the one used to migrate from the 'hosts.txt' tables to DNS can be applied also to the deployment phase of this proposal. An informational document describing the implementation phase and the detailed coordination procedures is expected. The deployment phase must also follow the directives produced by the current work on [RFC1327](#) mapping authorities, in order to insure consistency in the mapping information itself.

Another distributed directory service which can distribute the [RFC1327](#) mapping information is X.500. The coordination, alignment and uniqueness of mapping information between DNS and X.500 is an essential fact if it happens to have both systems in place. The ideal solution is a dynamic alignment mechanism which transparently makes the DNS mapping information available in X.500 and vice versa. Some work in this specific field is already being done [see Costa] which can result in a global transparent directory service, where the information is stored in DNS or in X.500, but is visible completely by any of the two systems.

## 7. Conclusion

The introduction of the new PX resource record and the definition of the X.400 O/R name space in the DNS structure provide a good repository for mapping information. The mapping information is stored in the DNS tree structure so that it can be easily obtained using the DNS distributed name service. At the same time the definition of the appropriate DNS space for X.400 O/R names provide a repository where to store and distribute some other X.400 MHS information. The use of the DNS has many known advantages in storing, managing and updating the information. A successful number of tests have been performed under the provisional top level domain "X400.IT", and their results confirmed the advantages of the method.

Software to query the DNS and then to convert between the textual representation of DNS resource records and the address format defined in [RFC1327](#) needs to be developed. This software must also allow a smooth implementation and deployment period, eventually taking care of the transition phase. A further informational document describing operational and implementation of the service is expected.

## 8. Acknowledgements

We wish to thanks all those who contributed to the discussion and revision of this document: many of their ideas and suggestions constitute essential parts of this work. In particular thanks to Jon Postel, Paul Mockapetris, Rob Austin and the whole IETF x400ops, RARE wg-msg and IETF namedroppers groups. A special mention to Christian Huitema for his fundamental contribution to this work.

## 9. References

- [CCITT] CCITT SG 5/VII, "Recommendation X.400, Message Handling Systems: System Model - Service Elements", October 1988.
- [RFC 1327] Kille, S., "Mapping between X.400(1988)/ISO 10021 and [RFC 822](#)", [RFC 1327](#), March 1992.
- [RFC 1034] Mockapetris, P., "Domain Names - Concepts and Facilities", STD 13, [RFC 1034](#), USC/Information Sciences Institute, November 1987.
- [RFC 1035] Mockapetris, P., "Domain names - Implementation and Specification", STD 13, [RFC 1035](#), USC/Information Sciences Institute, November 1987.
- [RFC 1033] Lottor, M., "Domain Administrators Operation Guide", [RFC 1033](#), SRI International, November 1987.
- [Costa] Costa, A., Macedo, J., and V. Freitas, "Accessing and Managing DNS Information in the X.500 Directory", Proceeding of the 4th Joint European Networking Conference, Trondheim, NO, May 1993.
- [Houttin] Houttin, J., Hansen, K., and S. Aumont, "Address Mapping Functions and Authorities", Internet-DRAFT, May 1993.

## 10. Security Considerations

Security issues are not discussed in this memo.

11. Authors' Addresses

Claudio Allocchio  
Sincrotrone Trieste  
Padriciano 99  
I 34012 Trieste  
Italy

[RFC822](#): Claudio.Allocchio@elettra.trieste.it  
X.400: C=it;A=garr;P=Trieste;O=Elettra;  
S=Allocchio;G=Claudio;  
Phone: +39 40 3758523  
Fax: +39 40 226338

Antonio Blasco Bonito  
CNUCE - CNR  
Reparto infr. reti  
Viale S. Maria 36  
I 56126 Pisa  
Italy

[RFC822](#): bonito@cnuce.cnr.it  
X.400: C=it;A=garr;P=cnr;O=cnuce;S=bonito;  
Phone: +39 50 593246  
Fax: +39 50 589354

Bruce Cole  
Cisco Systems Inc.  
P.O. Box 3075  
1525 O'Brien Drive  
Menlo Park, CA 94026  
U.S.A.

[RFC822](#): bcole@cisco.com  
X.400: C=us;A= ;P=Internet;  
DD.[rfc-822](#)=bcole(a)cisco.com;  
Phone: +1 415 6888245  
Fax: +1 415 6884575



Silvia Giordano  
Centro Svizzero di  
Calcolo Scientifico  
Via Cantonale  
CH 6928 Manno  
Switzerland

[RFC822](#): giordano@cscs.ch  
X.400: C=ch;A=arcom;P=switch;O=cscs;  
S=giordano;  
Phone: +41 91 508213  
Fax: +41 91 506711

Robert Hagens  
Advanced Network and Services  
1875 Campus Commons Drive  
Reston, VA 22091  
U.S.A.

[RFC822](#): hagens@ans.net  
X.400: C=us;A= ;P=Internet;  
DD.[rfc-822](#)=hagens(a)ans.net;  
Phone: +1 703 7587700