
APPENDIX A: CONTACT SUPPORT IN .TEL

A1. INTRODUCTION

To ensure that .tel domain name registrants are provided with a good service, the Sponsoring Organisation specifies the minimum level of support in provisioning (and in presentation and use of contacts) that will be populated in .tel domains. Programs that provision or operate on this data will be expected to provide these features.

Technically, communications contacts are stored within domains in the DNS using Naming Authority Pointer (NAPTR) resource records. Systems used to support or query the .tel TLD will need to help the user in provisioning and interpreting the communications contacts. This document lists the main features to be supported by these systems.

A1.1. SPECIFICATIONS

The specification for NAPTRs is included in the IETF document set RFC 3401, RFC 3402, RFC 3403. These documents cover the “Distributed Delegation Discovery Service” (DDDS), with RFC 3403 in particular covering the overall syntax and use of NAPTR resource records. The variant of DDDS on which the .tel service is based is described in RFC 3761 - this specifies the “E2U” DDDS Application. More clarification and interpretation of these standards is given in RFC5483.

This document covers the aspects not considered in these specifications and should be read in conjunction with them. In particular, this document lists the services people can expect to use to populate their .tel domain names, and the extra elements and services that will be supported in .tel-compliant programs.

In the following text, presence of “MUST” indicates an element that it is mandatory to implement, while “SHOULD” indicates an element that must be supported unless there is an overwhelming reason not to do this in a particular context, and “MAY” indicates an element for which implementation and/or use is optional.

A1.2. DOCUMENT STRUCTURE

This document lists the Enumservices to be supported in Section 2. This is followed in Section 3 by descriptions of the additional “non-Standards Track IETF” Enumservices that have been adopted for use in the .tel TLD as a convenience to users. Section 4 covers “auto-provisioning” support for client programs in discovering the system these should use to provision .tel domain data.

A2. ENUMSERVICE SUPPORT IN .TEL

Any TelHosting Provider’s Provisioning System, Web Proxies, and client programs designed to operate with .tel MUST recognise the following Enumservices. Provisioning Systems SHOULD allow provisioning of all of these, with the exception of those mentioned in Sections 2.1.1 and 2.2, where support MUST be provided. In the case of clients, these are of course dependent on the local availability of features capable of supporting the specified communications methods. However, these SHOULD (at least) recognise the following listed Enumservices and not treat their presence as an error.

Proxy web clients SHOULD support these services, and provide clickable links (as further described in Section 3, and the “Web-based Contact Handling” table). Whether or not a computer on which a browser (requesting the web proxy results) operates has external programs that support these services will of course vary, and so the result of clicking on these links will be computer-dependent.

A2.1. IETF STANDARDS TRACK ENUMSERVICES

These are those listed on the IANA Number Registry’s web site, at least including the following:

- sip
- h323
- voice:tel
- sms:tel
- ems:tel
- mms:tel
- sms:mailto
- ems:mailto
- mms:mailto
- email:mailto
- web:http, web:https
- ft:ftp
- fax:tel

Enumservices are listed on the IANA web site.
See <<http://www.iana.org/assignments/enum-services>>.

A2.1.1. NON-TERMINAL NAPTR SUPPORT

As a core element of the DDDS standards and thus the “E2U” application on which .tel is based, non-terminal NAPTRs (NTNs) are mandatory to implement. The .tel TLD supports provisioning and use of sub-domains to partition the information published by domain owners. For this reason, it also makes extensive use of non-terminal NAPTRs to “point” to these sub-domains. All clients (stand alone or web based) and all provisioning systems MUST implement support for non-terminal NAPTRs both internally and within the user interface they present to users.

A2.2. NON-IETF ENUMSERVICES

These are the additional Enumservices that have been adopted for use in the .tel TLD. These are detailed in the next Section, but are summarised here.

A2.2.1. VOICE OVER IP AND INSTANT MESSAGING ENUMSERVICES

The following Enumservices indicate that this contact can be used to start a communications session including voice/video and instant messaging with a user of typical services (such as AIM, Skype, and so on). The sub-type identifies the URI scheme of the system.

- x-voice:<system>
- x-im:<system>

A2.2.2. AUXILIARY/DESCRIPTIVE ENUMSERVICES

These auxiliary Descriptive Enumservices are “purely descriptive” Enumservices, used to label a containing NAPTR with textual information that can be presented to a querying user. They cannot exist alone in a NAPTR, but must be used with one of the Enumservices listed in the Sections above.

Descriptive Enumservices currently include Location Indicator Hints (LIH). As the name implies, the Location Indicator Hint class is intended as hints to the location of a user, or that this URL involves a Premium Rate Service (and so, if contactable, use of this URL may cost the caller more than a normal call). Examples of this class are:

- x-mobile
- x-home
- x-work
- x-main
- x-transit
- x-prs

In previous versions of this document, Descriptive Labels using the enumservice tag “x-lbl:” were specified and support was mandatory. These are now deprecated (as they could not support non-ASCII text, and so were too limited). Their role has been taken over by Textual Label records (.tlb TXT records) specified in the “TXT Support in .tel” document. “x-lbl” enumservices should no longer be provisioned, and can be ignored if encountered within a NAPTR.

A2.2.3. PROTECTED CONTENT ENUMSERVICE

In previous versions of this document, Protected Content NAPTRs using the Enumservice tag “x-crypto:” were specified and support was mandatory. In the current version of this document, support is not required. If encountered by a proxy system, these records can be discarded. Similarly, any Textual Label records (.tlb TXT records) specified in the “TXT Support in .tel” document that are associated solely with Protected Content NAPTRs can be discarded.

A3. ADDITIONAL ADOPTED CONTACT SPECIFICATIONS

This Section describes the Enumservices (other than those meeting current IETF standards track specifications) that have been adopted for use in the .tel TLD. It gives further details and specification for those Enumservices listed in Section 2.2, and in particular Enumservice(s) in Sections 2.2.1 and 2.2.2.

Note that unless indicated these specifications have not been condoned by or introduced to the IETF at this time. The IETF Enumservice registration process is in the process of change, and registrations for these elements will be re-considered once the new IETF Enumservice registration process is complete.

A3.1. VOIP AND IM ENUMSERVICES

These Enumservices imply that this contact is used for common VoIP/IM services (such as google or skype). By selecting this contact, the user would expect an appropriate program to

run and to initiate contact with the person who has an account with this service. The sub-type indicates the particular service within which the associated URI exists; this holds the service-specific URL scheme (e.g. gtalk, ymsgr).

In these Enumservices, x-voice is used for all real time communication sessions other than Instant Messaging. Thus it indicates that a video and/or a voice session could be started by using the associated URI in the NAPTR holding this Enumservice. Conversely, x-im is used to start an Instant Messaging chat session with the user listed in the URI. Also note that most existing services allow the user to switch between video/voice and IM chats in mid-session, so that the Enumservice indicates merely the kind of session to start.

The URI generated from the enclosing NAPTR will be the one appropriate for the VoIP/IM service, so will be of the form skype:jamesbrown, gtalk:awebuser, or msnim:alive1.

- x-voice:aim
- x-voice:skype
- x-voice:gtalk
- x-im:aim
- x-im:icq
- x-im:ymsgr
- x-im:msnim
- x-im:xmpp

In addition, “legacy” Enumservices that have been used in other public contexts are still supported by the client programs and web proxies, and are treated as if they were the replacements shown:

x-skype:callto => x-voice:skype

A3.2. AUXILIARY DESCRIPTIVE ENUMSERVICES

This class of Enumservices does NOT indicate the abstract Application or protocol to be used to process the URI contained in its NAPTR. Instead, it adds further descriptive information on the service this NAPTR represents.

An Auxiliary Enumservice is independent of the application and of the URI generated by the NAPTR that contains it.

The NAPTR holding an auxiliary Enumservice MUST contain at least one “active” Enumservice that indicates the treatment of the NAPTR’s generated URI; a NAPTR cannot include only Auxiliary Enumservices.

As mentioned above, Location Indicator Hints are used and must be supported. These are described next, with examples of NAPTR services fields containing these Enumservices following those descriptions.

A3.2.1. LOCATION INDICATOR HINTS

Members of this group are Experimental Enumservices. As such, in each case the type starts with the facet “X-”. They are also Auxiliary Enumservices. There is no intended “active” service implied by the presence of one or more of this set of Enumservices.

The set of these services consists of:

- x-mobile
- x-work
- x-main
- x-home
- x-transit
- x-prs

The intended action for these Enumservices is that they can be interpreted by a suitable client to indicate that this NAPTR is associated with a means of communication available when the publishing user is mobile, when that user is at work, as the main contact, when that user is at home, when that user is in transit, or that this contact will involve a premium rate call.

The client program may choose to select an appropriate indication when presenting this contact to a user. The textual values of these indicators need not be presented literally. Instead the client may interpret the hint and either act on this or present the contact with its own choice of indication. Thus x-mobile might be presented to the user with a specific Icon variant, or with the label “Handyphone number:”, or other appropriate indications.

A3.2.2. AUXILIARY ENUMSERVICE EXAMPLES

- E2U+voice:tel+x-work
- E2U+email:mailto+x-home
- E2U+voice:tel+x-prs
- E2U+sms:tel+x-mobile

Full examples:

```
IN  NAPTR      10   49   "u"  "E2U+voice:tel+x-prs"
    "!^.*$!tel:+44904999999!"      .

IN  NAPTR      10   50   "u"  "E2U+web:http+x-home"
    "!^.*$!http://members.yahoo.com/interests?.oc=t&.kw=myuserid&a
    mp;.sb=1!"      .

IN  NAPTR      10   51   "u"  "E2U+web:http+x-mobile"
    "!^.*$!http://opi.yahoo.com/online?u=myuserid&m=g&t=2!"
    .

IN  NAPTR      10   52   "u"  "E2U+sip+x-work"
    "!^.*$!sip:myuserid@example.net!"      .

IN  NAPTR      10   53   "u"  "E2U+email:mailto+x-main"
    "!^.*$!mailto:myworkuserid@example.com!"      .
```

Notes:

It is recommended that the length of auxiliary Enumservices should be limited where possible. For example, in a protected NAPTR, the service field is encrypted along with the rest of the NAPTR. As the total length of the protected content is limited, the maximum length of the REGEXP field will depend on the Services field and any LIH elements that contains.

A4. AUTO-PROVISIONING SUPPORT USING CONTACTS

This Section outlines the process by which .tel client software acting for a domain owner may automatically configure TelHosting settings based on the domain contents. The motivating factor behind this feature is that users may not be comfortable provisioning TelHost system connection information into a freshly installed .tel client program, particularly if this involves typing in long and complicated SOAP URLs to run TelHost-related actions.

Instead, it would be convenient for a domain owner to type in just the domain name he or she owns, and to have the settings (except for the TelHost account password, of course) configured automatically by the client program.

To this end, the Telnic-funded client programs (and TelHosting software) will look at contacts within three reserved sub-domains within a .tel domain name. These sub-domains are “_soap._nspapi”, “_http._nsp._apps” and “_https._nsp._apps”. The contacts expected in these sub-domains are shown below, and will allow the client programs to find the soap end point and the (human readable) web pages respectively. In this example, the TelHosting system hosts the example.tel’s zone, and will provision these contacts as shown:

```
$ORIGIN example.tel.
.      IN SOA ...
.      IN NS ...

_soap._nspapi IN NAPTR 10 50 "u" "e2u+web:https"
              "!^.*$!https://exampletelhost.com/client!" .
              IN NAPTR 10 51 "u" "e2u+web:http"
              "!^.*$!https://exampletelhost.com/client!" .
_http._nsp._apps IN NAPTR 10 50 "u" "e2u+web:http"
              "!^.*$!http://www.exampletelhost.com/login!" .
_https._nsp._apps IN NAPTR 10 50 "u" "e2u+web:https"
              "!^.*$!https://www.exampletelhost.com/login!" .
```

In all cases, the TelHosting Provider is advised that the sub-domain that the clients will expect is “_soap._nspapi” for the web service end point, and “_https._nsp._apps” or “_http._nsp._apps” for the “human readable” web pages.

The client programs will look for contacts in these sub-domains when they are given the .tel domain name. If these are not found (or the contacts within them are not usable) then the client program will inform the end user that auto-provisioning is not available, and ask for the settings to be entered manually.

Note that the ORDER/PRIORITY reflects the TelHosting Provider’s preference. The client may well have its own preference.

TXT SUPPORT IN .TEL

B1. INTRODUCTION

This document describes the usage of TXT records within .tel domains.

The TXT resource record type is defined in the main DNS standard (RFC 1035). TXT records can be stored in the DNS and hold a series of strings, each of which is less than or equal to 255 octets in length. Multiple TXT records can exist in a single domain, with all available TXT records being returned in response to a query. The order in which these records are returned is not guaranteed.

TXT records are used in .tel domains to store non-contact related information about the registrant. Generic descriptions or messages can be displayed as well as more structured data such as addresses or the category of business (for corporate registrants). These records can also be used for “system messages” – data intended as a “hint” to the recipient program, and not normally for presentation to the user.

B1.1. GENERIC DATA IN TXT RECORDS

Where a TXT record is returned and the first (or only) string in that record does not hold either of the reserved identifier patterns shown below (in Sections 1.2.1 and 1.2.2), it is treated as holding a generic set of data to be displayed to the end user in the normal way. Within .tel, clients (either stand-alone or web proxy mediated) are expected to be able to handle TXT records that have multiple strings, and to present these accordingly to the end user. Similarly, provisioning systems are expected to allow the registrant to populate TXT records into his or her domain, and to allow each of these TXT records to hold multiple strings.

B1.2. STRUCTURED DATA WITHIN TXT RECORDS

There are several kinds of structured data that are given special processing rules in .tel domains. Of these, the two main forms are Structured Keywords, and Structured System Messages. In all cases, the initial string holds a specific identifier pattern. This is informally named the “magic identifier” and should be used by clients to determine whether the TXT record is one of these special cases, or instead should be treated as a generic TXT record. The second string will hold a version value, expressed as a decimal numeric.

The subsequent strings in the TXT record are interpreted according to the structured message type; for both keywords and system messages these are arranged in pairs, with the first of the pair holding the data type, and the second string in that pair holding the data value. There will be at least one pair of such type and data strings. Beyond this structure, the formal limit is the maximum size of a DNS resource record. In practice, of course, creating a massive TXT record (or set of TXT records) is unwise; the DNS is optimised for relatively short messages, and not all networks react correctly to excessively long DNS messages.

B1.2.1. STRUCTURED KEYWORDS RECORD

A Structured Keywords record begins with a special value in its first string; this is “.tkw” (short for .tel keywords). Strings after the next (version number) string consist of keyword (type, value) pairs. The types of keywords are listed in Section 3.1.

B1.2.2. STRUCTURED SYSTEM MESSAGE RECORD

This type of structured data uses the TXT resource record, and has similarities to the Keyword form described above. The difference lies in the intended use of the contained data and the identifier by which this kind of content is indicated. The expectation is that this TXT record is intended to influence the client behaviour, and is NOT intended for presentation to the end user.

A client program receiving a TXT record in which the first string is exactly “.tsm” (short for .tel system message) should interpret the strings following the version in this TXT resource record as a structured System Message according to this specification.

System message types are covered in Section 3.4.

B1.2.3. VERSION MATCHING

When evaluating its capability against that required to understand the received structured record format, the client program will consider the next string in the TXT record after the magic identifier – this contains the version value. The client program will be expected to be capable of handling a certain major version of this specification. To compare its capability with that indicated as required in the TXT record, it should examine the version string. This version string should be padded with “0” characters to the right if it has less characters than the client’s internal capability value. Conversely, the client’s internal capability value should be right-padded with “0” characters if it is shorter than the version value in the received structured data record.

Once this is done, the client should treat both its internal capability string and the extended TXT record’s version string as if they were integer values, and compare them numerically. If the client program’s capability is higher or equal to the version value, then this client can be expected to understand the rest of this structured data. If the first digit in the internal capability string is the same as that in the TXT record’s version string, then there is major version compatibility; this means that the client will be able to parse the TXT record content, even if it does not recognise the values of the fields. If the value in the version string is higher than in the client’s capability, then the client cannot make this assumption, and must discard the TXT record.

For this current version of the specification, provisioning systems that populate .tel domains with structured data records must use the value “1” as the version string in these records.

B2. .TEL TXT RECORD FORMAT

There is no guarantee of the order in which TXT records will be delivered in a DNS response. Registrants (and the provisioning systems they use) cannot assume the order in which these TXT records will be processed by a client program. However, strings that exist within a TXT record will be processed in a left to right order, and advantage can be taken of this by registrants: the order of strings provisioned within a TXT record can and will be maintained by client programs.

For a generic TXT record, the strings it contains will be presented to the user in the left-to-right sequence in which they appear in the record.

Where the TXT record contains structured keywords, a similar approach is taken. In principle, keywords are independent of one another. By grouping keywords together in a single TXT record, a logical relationship between these keywords is established. For example, a TXT record with an initial keyword type “postalAddress” may contain the subsequent keyword types “townCity”, “postalCode”, and “stateProvince”. The value of the postal address keyword type denotes a “label” that can be used to differentiate between different addresses, e.g. “Home”, “Work”, “School”. The values for the subsequent keywords will normally be displayed to the user in the order in which they appear within the TXT record. This allows a postal address to be presented in the “natural form” as chosen by the registrant, rather than requiring the client program to collate and interpret these values before display.

Only a single structured system message is permitted within a TXT record. However, registrants must take care not to have conflicting system message records within a domain. As the order in which TXT records are delivered is indeterminate, contradictory system messages are unwise (as the eventual interpretation made by a querying client program cannot be determined). For example, to avoid such conflicts, there **MUST** be only one .tsm PDDX record in any domain or subdomain.

B2.1. PARSING

Parsing the structured data records should be performed on a per-TXT record basis. The general form for structured data records is shown here:

```
<magic> <version> <primaryType> <primaryValue> [ <secondaryType> <secondaryValue> ]  
...  
  
magic = ".tkw" / ".tsm" / ".tlb" / ".tad" / ".gad" / ".tia"  
        - indication that this TXT record contains keywords/system message/label/...  
  
version = 1*2(0..9) -      examples: "1", "11", "20"
```

For structured keywords, the TXT record may contain more than one keyword type/data pair. In the specific case of the pa and bpa keywords, subsequent keywords are *expected* in the same TXT record (forming a complete postal address), and will be processed and presented in the order in which these appear in that TXT record.

In the case of system message records, this version of the specification allows only one system message per TXT record. However, that system message may include a sequence of key/value pairs, and the pairs that are included are version-dependant.

B2.1.1. LAZY PARSING AND PRESENTATION OF STRUCTURED KEYWORDS

When a client receives a record that holds keywords, the record can be parsed by considering the initial keyword type and value, and then taking each subsequent keyword in turn. The types of these subsequent keywords can be discarded, and their keyword values can be simply concatenated (with a suitable separator between each of the strings). The result can then be presented to the user with no further interpretation; in the majority of cases, the initial

keyword type will give the user enough of a semantic “hint” to understand the context and meaning of the record.

B3. SPECIAL STRINGS IN TXT RECORDS IN .TEL

B3.1. KEYWORD TYPES

The keyword types and values in the following tables are deliberately loosely defined to allow extension and innovation by the community. The suggested keyword types in this document provide a “core” to which others can be added in the future. Where possible the keyword types should remove the need for structured values. For instance, separating out keyword types for the address components removes the need to parse comma-delimited addresses.

The keywords listed here are semantically broken down into two categories: Individual and Corporate. There is no restriction on mixing the types when provisioning, but it makes sense to logically differentiate between the two when implementing separate interfaces for White and Yellow pages or when presenting these records to a user who has asked for information on a queried domain.

As an example of the difference in usage between the corporate keywords and individual keywords: Individuals will have the opportunity to populate the “organisation” field with the name of their employer – should they wish to do this.

So “Adam Smith”, employee of Telnic, would put “Telnic Limited” in the “Organisation” field. Conversely, Telnic would populate the field “Business Name” with “Telnic Limited” and leave “Organisation” blank.

B3.1.1. CHARACTER SETS AND LANGUAGE PREFERENCES

The keyword types listed below are all in the US-ASCII range. Shorthand equivalents to the keyword types are shown inside parentheses and in *italics* within the following tables. Where possible, it is recommended that these shorthand forms be used in the keyword type strings when provisioning keywords within a TXT record. To limit the potential for misrepresentation, it is also recommended that these type values be provisioned into the TXT record, (regardless of the language preference of the registrant or potential readers), rather than creating new language-specific keyword types. Client programs receiving these TXT records should be able to convert the types into a “local” equivalent for presentation, and provisioning systems should be able to map local keyword type names into these “canonical forms”.

It is assumed that the contents of the keyword value strings are in the Universal Character Set, and are encoded in UTF-8. Registrants should be aware that potential readers of these keywords might not share the same language or cultural preferences.

This is especially important when provisioning corporate keywords such as Business Area and Business Sub-Area. If the registrant wants a multi-lingual audience to know that it is involved in shoe repairs, then the language specific value “Cobbler” may not be appropriate for the bsa.

B3.1.2. INDIVIDUAL KEYWORDS

These keywords relate to individuals, both commercial and non-commercial. The “short-hand” version is shown in *italics*.

- salutation (*s*) e.g. “Mr”, “Mrs”, etc
- commonName (*cn*) e.g. “Adam J. Smith”
- firstName (*fn*) e.g. “Adam”
- lastName (*ln*) e.g. “Smith”
- nickName (*nn*) e.g. “Ade”
- date of Birth (*dob*)² e.g. “20/03/1966”
- gender (*g*) e.g. “Male” or “Female”
- maritalStatus (*ms*) e.g. “Married”, “Single”, etc
- postalAddress (*pa*) e.g. “Favourite restaurant”
“a1” “8 Wilfred Street”
“tc” “London”
“pc” “SW1E 6PL”
“c”, “UK”
- addressLine1 (*a1*) e.g. “8 Wilfred Street”
- addressLine2 (*a2*) e.g. “Victoria”
- addressLine3 (*a3*) e.g. “Westminster”
- townCity (*tc*) e.g. “London”
- stateProvince (*sp*) e.g. “Hampshire”
- postalCode (*pc*) e.g. “SW1E 6PL”
- country (*c*) e.g. “Scotland”
- latitudeLongitude (*ll*)³ e.g. " 52° 11' 1.55" N / 0° 5' 16.37" W "
- organization (*o*) e.g. “Telnice Limited”
- department (*d*) e.g. “IT”
- jobTitle (*jt*) e.g. “Chief Rocket Scientist”

² Dates should be assumed to be dd/mm/yy or dd/mm/yyyy, rather than mm/dd/yy or mm/dd/yyyy.

³ Note that this value includes the degree character – this character (u+00B0) is represented by two bytes in UTF-8; ‘C2 B0’

- hobbiesInterests (*hi*) e.g. “Scuba Diving”
- freeText (*ft*) e.g. “Selfless, innovating, trustworthy, red hair”

B3.1.3. CORPORATE KEYWORDS

These keywords extend those relating to individuals and pertain to businesses and corporate entities. The shorthand version is again shown in *italics*.

- businessName (*bn*) e.g. “Marie’s Marriage Emporium”
- businessPostalAddress (*bpa*) e.g. “London Office”
 “a1” “8 Wilfred Street”
 “tc” “London”
 “pc” “SW1E 6PL”
 “c” “UK”
- businessArea (*bar*) e.g. “Weddings”
- businessSubArea (*bsa*) e.g. “Dresses, Flowers, Venues”
- serviceArea (*sa*) e.g. “Hertfordshire, Essex, Cambridgeshire”

B3.2. ADVERT TYPES

At present, a number of advert formats are supported within .tel. These are all held within a dedicated “_ad” sub-domain and can be split into three broad categories

B3.2.1. GOOGLE ADSENSE

These are identified by the “.gad” type and are further structured as follows:

- Identifier – Always set to “.gad”
- Version – Originally will be set to “1” in all cases
- Display Preference – Required numeric value in the range 1-3 where:
 - Indicates advert should be displayed at the top of the page
 - Indicates advert should be displayed on the right of the page
 - Indicates advert should be displayed at the foot of the page
- Preference - Numeric value between 0 – 65535.
 Within each display-preference set, this is the sort order.
 Value 0 is 1st, value 65535 is last.
- PublisherID – Alpha numeric unique identifier
- AdUnitID – Numeric ad identifier

B3.2.2. TELNIC ADS

These are identified by the “.tad” type and are further structured as follows:

- Identifier – Always set to “.tad”
- Version – Originally will be set to “1” in all cases.
- Display Preference – Required numeric value in the range 1-4 where:
 1. Indicates advert should be displayed on the top of the page
 2. Indicates advert should be displayed on the right of the page
 3. Indicates advert should be displayed at the foot of the page
 4. Indicates that content is RichContent and should be displayed in the area of the page reserved for RichContent.
- Preference – Numeric value between 0 - 65535.
Within each display-preference set, this is the sort order.
Value 0 is 1st, value 65535 is last.
- Title – Up to 255 bytes of text which forms the title of the ad/richcontent.
- Label – Up to 255 bytes of link text to be shown as the last line of the advert.
For RichContent this will be shown immediately after the end of the content.
- URI – Collection of key / value pairs (where each key is labeled “uri”). Consecutive values will be concatenated to provide the final URI to be used within the ad / richcontent. This ensures URIs in excess of 255 bytes can be created.
- Description – Collection of key / value pairs (where each key is labeled “desc”). Consecutive values will be concatenated to provide the final description to be used within the ad / richcontent. This ensures descriptions in excess of 255 bytes can be created.

B3.2.3. TELNIC IMAGE ADS

These are identified by the “.tia” type and are further structured as follows:

- Identifier – Always set to “.tia”
- Version – Originally will be set to “11” in all cases
- Display Preference – Required numeric value in the range 1-6 (excluding 4) where:
 - 1 – Indicates desktop advert that should be displayed at the top of the page
 - 2 - Indicates desktop advert that should be displayed on the right of the page
 - 3 – Indicates desktop advert that should be displayed at the foot of the page
 - 5 - Indicates smartphone advert that should be displayed at the top of the page
 - 6 - Indicates smartphone advert that should be displayed at the top of the page
- Preference - Numeric value between 0 – 65535.
Within each display-preference set, this is the sort order. Value 0 is 1st, value 65535 is last.
- Img – Collection of key / value pairs (where each key is labeled “img”). Consecutive values will be concatenated to provide the final URI to be used to load in the referenced image advert. This approach ensures URIs in excess of 255 bytes can be created.

- URI – Collection of key / value pairs (where each key is labeled “uri”). Consecutive values will be concatenated to provide the final URI to be used within the image ad. This ensures URIs in excess of 255 bytes can be created.
- Caption - Collection of key / value pairs (where each key is labeled “cap”). Consecutive values will be concatenated to provide the final caption to be used within the image ad. This ensures captions in excess of 255 bytes can be created.

B3.3. TEXT LABELS

Contacts (held in NAPTR records) can have associated textual labels. These textual labels are typically presented to users along with the contact with which they are associated. Each such text label is held in a structured text record with a “magic identifier” type of “.tlb”. The version number string follows this. Next, two strings are stored that hold textual representations of the order and preference values of the NAPTR with which this label record is tied. The remaining string (or strings) within the record holds the label itself, interpreted as a sequence of UTF-8 characters. Each text label record is stored in the same domain as the NAPTR to which it applies, and MUST have the same order and preference values as an existing NAPTR. If a text label record is encountered that does not have a NAPTR with matching order and preference values, it must be discarded.

B3.4. SYSTEM MESSAGE TYPES

At present, only two Structured System Message types are defined. One is a tag indicting a Domain Display String (with message type “dds”). This is used to provide a “human-friendly” replacement for the fully qualified domain name. The other is the system message type “pddx”. This indicates a range of information and is a hint to clients on how to present the domain’s content in which it appears.

B3.4.1. DOMAIN DISPLAY STRING (“DDS”)

The DDS record is optional. If present, it is used by clients in place of the fully qualified domain name. This record holds a string that is interpreted as a sequence of UTF-8 characters. The string could include characters not possible in the domain name itself, so that, for example, the domain foo-bar.example.tel might include a DDS string of “Foos und Bäs”, which could be presented to the user in place of the domain name itself.

B3.4.2. PDDX

This record is present in every user-visible fully qualified domain within .tel, and holds a set of data items that clients (such as a web proxy) can use to determine how to present different data items. It is structured as a set of key/value pairs, where each key/value pair exists exactly once in the record.

Note that the content of the record changes to reflect new functionality, and so the version string is very important. Clients will assume that, if the version string has the same “major” version number to the one they expect, the client can understand the content of the record. If however the “major” version number in this record is higher than the one for which a client is

designed, it cannot assume that the items in the record can be decoded, and so must ignore the record entirely.

If the major version number in a record is the same as the default value for which a client is designed whilst the minor version value is higher, there may be other key/value pairs in the record, but the ones expected by the client will be present. If the minor version number is lower than the client's default, some key/value pairs it expects may be missing, but the client should be able to assume default values for the missing items; the record content will be recognisable. The list of key/value items present in this record includes:

Type	Value	Description
pddx	"1" or "0"	"1" is used to indicate that private data is absent (does not exist) with "0" indicating that there may be private data present within the domain.
color1	"#<six digit hex code>"	Six digit hex code used to overwrite the standard colour for a specific section of a specified CSS template. This is purely a TelProxy directive.
color2	"#<six digit hex code>"	Six digit hex code used to overwrite the standard colour for a specific section of a specified CSS template. This is purely a TelProxy directive.
color3	"#<six digit hex code>"	Six digit hex code used to overwrite the standard colour for a specific section of a specified CSS template. This is purely a TelProxy directive.
color4	"#<six digit hex code>"	Six digit hex code used to overwrite the standard colour for a specific section of a specified CSS template. This is purely a TelProxy directive.
css	CSS template indicator	Value used to specify which TelProxy template should be used to render the domain name This is purely a TelProxy directive.
pss	"2" or "1" or "0"	"0" indicates that the Telproxy search functionality should cover all .tel domains, "1" indicates that only the current apex domain (and subdomains) will be searched, while "2" indicates no searching.
hml	"1" or "0"	"1" is used to indicate that a "Manage" domain link should be provided within the TelProxy with "0" indicating that no link

		should be shown. This is purely a TelProxy directive.
htl	"1" or "0"	"1" is used to indicate that a link to telnic.org should be provided within the TelProxy with "0" indicating that no link should be shown. This is purely a TelProxy directive.
gan	Empty or Analytics ID	Is added to indicate whether GoogleAnalytics logging is required. If empty, GoogleAnalytics is deemed to be inactive, otherwise GoogleAnalytics is deemed active and the relevant GoogleAnalytics script will be activated using the contained value as the GoogleAnalytics key. This is purely a TelProxy directive.
log	N/A	Added for possible future usage in logging statistics to custom endpoints.
bkg	URL	Optional fully qualified URL that describes the location of the background image. This is purely a TelProxy directive.
bip	"2" or "1" or "0"	Used in conjunction with "bkg" to indicate how the TelProxy background image should be rendered. "2" indicates that the image should be tiled, "1" indicates the image should be stretched, "0" indicates should be displayed once. This is purely a TelProxy directive.

Clients can use these indications as they choose. As an example, the Telnic-developed web proxy client will disable the "friend" UI elements when presenting queried .tel domains that include a TXT record with this system message with a pddx value of 1. The rationale for this choice is that, if there is no private data published in this .tel domain, then there is little point in sending a friend request message to the domain owner.

B3.5. FULL EXAMPLES OF TXT RECORDS IN .TEL

```

IN  TXT  "Fun Contacts" "" "Life outside work"
IN  TXT  ".tkw" "1" "o" "Telnic Limited"
IN  TXT  ".tkw" "1" "s" "Mr" "fn" "James" "fn" "Fenimore" "ln"
      "Cooper" "jt" "Technical Author" "g" "male" "dob" "15/09/1789"
IN  TXT  ".tkw" "1" "pa" "Where I work"
IN  TXT  ".tkw" "1" "bpa" "Telnic South" "a1" "8 Wilfred Street" "a2"
      "Victoria" "tc" "London" "pc" "SW1E 6PL" "c" "UK"
IN  TXT  ".tlb" "1" "10" "50" "This one redirects to my voicemail"

```

```
IN   TXT   ".tsm" "14" "dds" "I call this domain MiniMe"

IN   TXT   ".tsm" "14" "pddx" "1" "color1" "#aabbcc" "color2" "#123456"
      "color3" "#ffffff" "color4" "#111111" "css" "2" "pss" "1" "hml"
      "1" "htl" "1" "gan" "UA-12345678-9" "log" "" "bkg"
      "https://www.example.com/img/example.jpg" "bip" "1"


---


IN   TXT   ".tad" "1" "4" "51" "Vertical Farming" "more >>"
      "uri" "http://www.richcontent.com/h"
      "uri" "uwssample/rich.html?domain=huw.tel"
      "desc" " Vertical farming is a proposed agricultural technique"
      "desc" " involving large-scale agriculture in urban high-rises "
      "desc" "or farmscrapers Using recycled resources and greenhouse"
      "desc" " methods such as hydro..."

IN   TXT   ".gad" "1" "2" "51" "pub-6575877768956787" "9113106953"


---


```