# What’s a TLD?

A top-level domain (TLD) is the suffix at the end of a domain name, such as “.com”, “.uk” and “.nz”. It represents the highest level division of the Domain Name System (DNS) hierarchy.

# What’s the challendge with acceptance of all TLDs?

The Internet is continuing to grow at a rapid pace. During the 1980s/90s, the format of domain names followed a simple pattern ending in either a small number of common three characters like “.com” and “.net”, or a two-letter code that represented a country like “.de” and “.uk”.

Times have changed. Since 2001, TLDs comprising of more than three characters long (think of “.info” or “.museum”) were introduced, and since 2010, non-Latin strings – known as internationalized domain names (IDNs) – have been added to the root zone.

The ICANN Board’s approval of the new gTLD program in 2011 has allowed hundreds of additional TLDs to be added. This means that the variety of domain names has expanded even further.

Software vendors, web site developers, and others may limit what they allow as a valid domain name in their applications. This constrained the Internet’s growth, consumer choice and promotion of market competition on-line. The effort toward universal acceptance of all TLDs seeks to ensure that the systems that perform domain name validation and display do it in a correct way that allows for all valid domains to function correctly. Essentially, all domains should function properly regardless of the script they are written in or the time at which theywere implemented: 20 years ago or yesterday.

To properly and fully support today’s DNS, implementers need to deploy software and solutions that take account of all of these developments. Software needs to fully accept all the varieties of domain names, including domain endings containing four or more characters as well as IDNs.

# What has changed?

**No predetermined length.** Until 2001, TLDs were either 2 or 3 characters long. This is no longer true. Does your software limit TLDs to a certain number of characters?

**No fixed set of TLDs. In 2001, there were about 250 TLDs.** Now there are over 900 and growing. Hundreds more TLDs are likely to be added to the root zone in the following years. Does your software have a hard-coded list of valid TLDs that it checks against? Is it regularly updated? Or does your application have a fixed drop-down box?

**Non-ASCII domains.** Fields that accept domain names as input (such as email addresses, URLs, etc.) need to accept not just ASCII but other scripts to work properly (e.g. Han, Devanagari, Cyrillic, etc). Can your software correctly accept “испытание” if entered into a domain-related field?

**Multiple representations.** Non-Latin script domains introduce a new idea – presentation and wire formats are different. For example, “xn--zckzah” and “f-.A!--” are the exact same domain. Would your software know to treat them the same?

**Dos and Don’ts**

**✘ Don’t check domain validity if you don’t need to.** A lot of applications don’t need to constrain the domain field, so unless you have a compelling reason to constrain it, leave it open.

**✘ Don’t check the length of a domain to determine validity.** You can no longer assume domain endings will be 2 or 3 characters long. They potentially can be between 2 and 63 characters long.

## ✔ Do use an IDN library to properly identify and handle Internationalized Domain Names.

There are many libraries (a lot of them are free) that are used by major software vendors to implement this functionality. Make sure the library supports the most current standards (i.e IDNA2008), as the older standard introduces compatibility issues.

**✘ Don’t use a hard-coded list of top level domains in your application.** If you need to check if a domain exists, the best way to do it is using the DNS protocol. A live DNS query happens quickly and will provide your application with the most up-to-date data available.

**✔ If you require a hard-coded list, do make sure it is regularly updated** (e.g., daily) using an appropriate methodology. ICANN provides some sample toolkits on how this might be done.

**✔ Do ask questions if you are not sure.** ICANN is happy to help provide advice to software developers and implementers on what is needed. Contact us at: *tld-acceptance@icann.org*.

**✔ Do report websites or software that has problems accepting existing domains.** If you notice a website that has problems, let us know and we’ll try to reach out to the operators to encourage them to follow these guidelines.

# Sample toolkit - TLD verification tool

The primary method of correctly checking if a domain is valid is to use the DNS. If the application has access to the Internet (most applications do), the best way is simply to perform a DNS query. This ensures the most accurate and up-to-date data is returned from the most authoritative source – the DNS itself.

In some rare cases, it is not possible to use the DNS protocol. When software programs need to check whether a top-level domain is valid, but is not able to perform an online check, ICANN provides guidance on alternate methods. In particular, the program may need to use a regularly updated list of valid top-level domains to perform its checking. ICANN has put together sample programming code that software developers can use. The code is available under an open source license at: *https://github.com/icann*.

# More information

To learn more about the effort, visit us at: *[http://www.icann.org/en/resources/universalacceptance](http://www.icann.org/en/resources/tld-acceptance)*

To share your ideas and suggestions on the topic email us at: *tld-acceptance@icann.org*.

Follow us on Twitter @ xxxxxxxx