

Variant Discussion for l and dotless l



Latin GP
July 2020

Variant Mechanism

- ⦿ Variant codepoints are two codepoints which are interpreted to be “same”.
- ⦿ What happens with variant codepoints?
 - Each of them can equally be used to generate a label, if both are in the repertoire.
 - When someone applies for a TLD using any of these codepoints, all possible variant labels will be generated. These Variant labels are recommended to be either assigned to the same registry operator or blocked.

Issue for dotless l and l

- The behavior of dotless l downcasing also contribute to the variant consideration

Example of Browser Behavior

Using lower case letters in the URL bar

User Input	zil	zil
Output UTF after Downcasing	007a 0069 006c	007a 0131 006c
LDH Label	zil	xn--zl-hpa

Using upper case letters in the URL bar

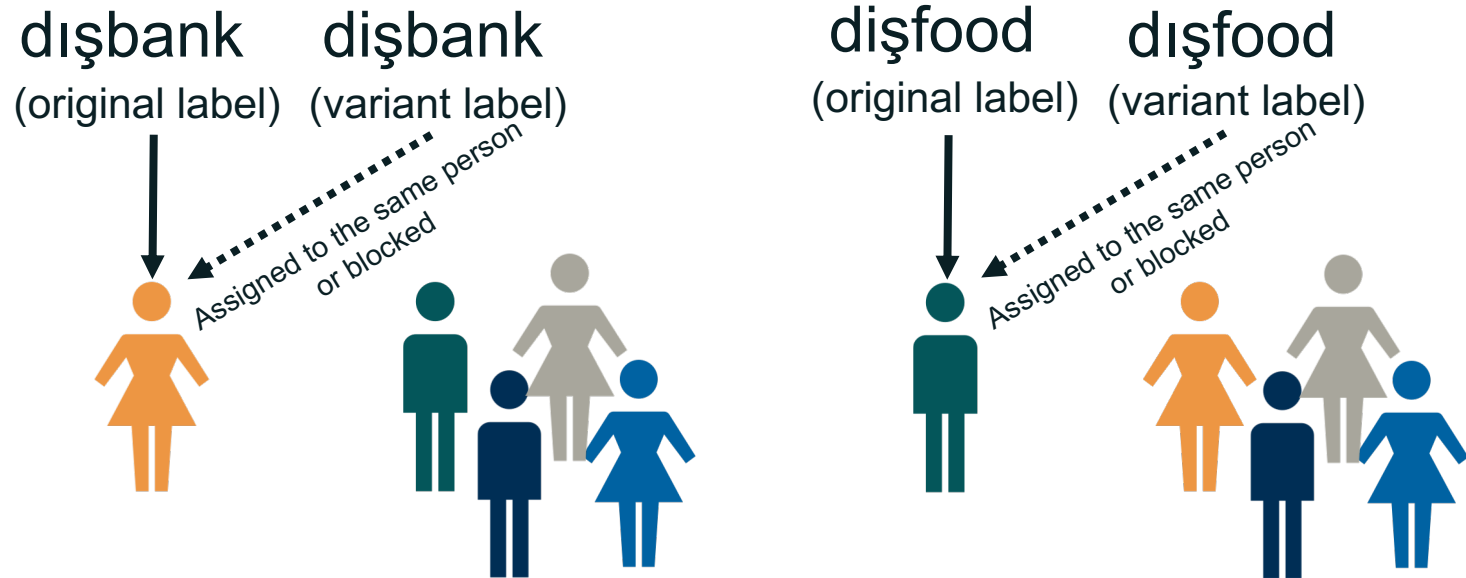
User Input	ZiL	ZIL
Output UTF after Downcasing	007a 0069 0307 006c	007a 0069 006c
LDH Label	xn--zil-9dc	zil

Variant Mechanism Examples

Assumption: LATIN SMALL LETTER DOTLESS I U+0131(ı) and
LATIN SMALL LETTER I U+0069 (i) are variant codepoints.

Possible Example:

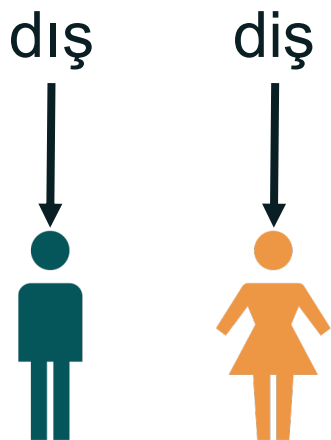
A bank 'dış' wants to apply for a TLD 'dışbank', while
a restaurant 'dis' wants to apply for a TLD 'disfood'.



Each code point can still be used in the TLD application. Once a TLD is applied, its variant TLD(s) will be generated and be blocked or be assigned to the same applicant.

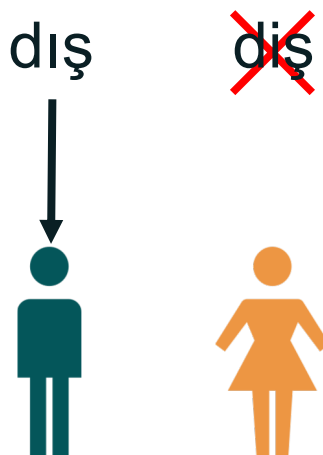
Variant Mappings and Dispositions

Case 1: No variant



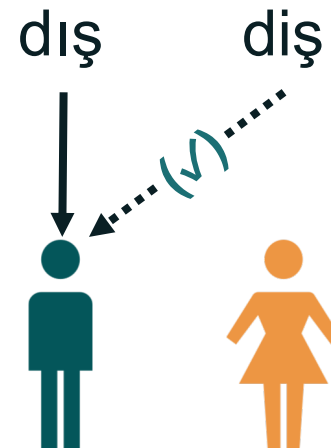
Case1: The TLD diş and diş could be owned by different applicant, which result different websites for the end-users

Case 2: variant with blocked disposition



Case2: Only one of the two can be registered. The other will be blocked and not available to original or any other applicant

Case 3: variant with allocatable disposition



Case3: The TLD diş is registered and the TLD diş is possible to be allocated only to the same applicant, based on the policy, directly or any later time.

An example

An IT company ‘issız communications’ wants to apply for a TLD ‘issız’, while a restaurant ‘issiz restaurant’ wants to apply for a TLD ‘issiz’.

- Both “issız” and “issiz” are legitimate words in Turkish. They have different meanings.
 - The communications company (issız) would be happy to have “issiz” as TLD, too, due to current habit of using “i” instead of “ı” in Turkey.
 - The restaurant will not be interested in having “issız” as TLD, in addition to “issiz”, since nobody would use “ı” instead of “i” in a domain name.

Case 1 (no variant) solution is not applicable, since capital “I” has lowercase “ı” in Turkish whereas it has lowercase “i” in English. The uppercase of “i” in Turkish is capital “İ”. It is not possible to know which language the user is using in a browser. Due to this ambiguity, we can ignore Case 1 as a solution.

Variant Dispositions: 'blocked' and 'allocatable'

- ⦿ Each codepoint variant mapping will be assigned a disposition value, e.g. 'blocked' or 'allocatable'.
- ⦿ The disposition 'blocked' means that the generated variant TLD(s) cannot be activated at all, even for the original TLD applicant.
- ⦿ The disposition 'allocatable' means that the generated variant TLD(s) can be activated to the same applicant.

Questions?

- ⦿ Should dotless I be allowed in the root-zone?
- ⦿ If dotless I is included in the LGR, is a variant relationship with letter I warranted?, why?
- ⦿ If variant relationship is warranted, then using the dıŝ or diŝ examples, how to handle them?
 - A. Only one of them (either dıŝ or diŝ) to be added to the root zone.
For this case, U+0131(ı) and U+0069 (i) should be variant code points with 'blocked' disposition.
 - B. Both dıŝ or diŝ can be added to the root zone but could only be assigned to the same TLD applicant.
For this case, U+0131(ı) and U+0069 (i) should be variant code points with 'allocatable' disposition.
 - C. Both dıŝ or diŝ can be added to the root zone and can assigned to the different TLD applicant.
For this case, U+0131(ı) and U+0069 (i) are not variant code points.