Proposal to add a derived data file for the "IDNA_Property" to /public/idna

Nov 11, 2021 — Asmus Freytag, Ken Whistler

Proposed outcome:

Establish a new derived property (IDNA_property)and add it /public/idna updated each version, starting with Unicode 15.0

File Name: idna2008-15.0.0.txt
Property Type: Derived, Enumerated
Property Alias: IDNA_Property (long), IDNA (short)
Property Value Aliases:

PVALID;	Protocol_Valid
DISALLOWED;	Disallowed
UNASSIGNED;	Unassigned
CONTEXTO;	Context_Other
CONTEXTJ;	Context_Joiner

(These property values are generally used in allcaps, and should therefore be listed this way)
Stability: New enumeration values may be added if IDNA2008 is modified, exceptions to the derivation will track any exceptions published by IETF, values may change between versions

Background and Rationale

The IDNA 2008 protocol uses a derivation from Unicode property data to decide whether a code point is DISALLOWED or PVALID (unless explicit exceptions apply).

This means, implementers can immediately use a new version of Unicode, by making the required derivations on their own. Unicode has long maintained a data file that contains the relevant data, but that file is more complex and a bit unwieldy to use if all you want to know whether something is PVALID under IDNA2008.

For the convenience of application and library developers and others, the IETF has supplied, and IANA maintains, derived property tables for several version of Unicode; however the latest of these, at the time of writing covers Unicode 11.0.0, while the UTC is already developing Unicode 15.0.0.

We propose to provide a purely derived data file that essentially matches these derived data, but provides timely updates anytime the Unicode Standard adds characters or tweaks General Category values that influence the derivation.

The proposed derived property provides only the classification of code points by IDNA2008_property value and does so in a way that can be easily compared to the files in the IANA repository.

The proposed format is:

```
0000..002C  ; DISALLOWED  # NULL..COMMA[1]
002D        ; PVALID      # HYPHEN-MINUS
002E..002F  ; DISALLOWED  # FULL STOP..SOLIDUS
0030..0039  ; PVALID      # DIGIT ZERO..DIGIT NINE
003A..0060  ; DISALLOWED  # COLON..GRAVE ACCENT
0061..007A  ; PVALID      # LATIN SMALL LETTER A..LATIN SMALL LETTER Z
007B..00B6  ; DISALLOWED  # LEFT CURLY BRACKET..PILCROW SIGN
00B7        ; CONTEXTO    # MIDDLE DOT
00B8..00DE  ; DISALLOWED  # CEDILLA..LATIN CAPITAL LETTER THORN
 .
 .
```

To maintain interoperability with tools designed for UCD data files, we suggest retaining the standard UCD conventions for range indicator (“..”), field delimiter (‘;’) and comment identifier (‘#’), but to allow ranges to cross script, block and general category boundaries. (In addition to using “NULL” instead of <cotrol-0000> in the comment in the first line). This format makes any files line-by-line comparable to the earlier files in the IANA registry after a few simple, global substitutions.

Here’s the same data as formatted in https://www.iana.org/assignments/idna-tables-11.0.0/idna-tables-11.0.0.txt.

```
    Codepoint    Property  Status                Description

      0000-002C DISALLOWED        NULL..COMMA
           002D PVALID            HYPHEN-MINUS
      002E-002F DISALLOWED        FULL STOP..SOLIDUS
      0030-0039 PVALID            DIGIT ZERO..DIGIT NINE
      003A-0060 DISALLOWED        COLON..GRAVE ACCENT
      0061-007A PVALID            LATIN SMALL LETTER A..LATIN SMALL LETTER Z
      007B-00B6 DISALLOWED        LEFT CURLY BRACKET..PILCROW SIGN
           00B7 CONTEXTO          MIDDLE DOT
      00B8-00DE DISALLOWED        CEDILLA..LATIN CAPITAL LETTER THORN
```

The suggested header for the file is as follows:

```
# Idna2008-00.0.0.txt
# Date: 2021-00-00, 15:23:07 GMT [KW]
# Copyright 2021 Unicode, Inc.
# For terms of use, see https://www.unicode.org/terms_of_use.html
#
# This file lists the values for the derived property IDNA_property
# as defined by applying the algorithm in RFC5892 to the Unicode
```

[1] The value “NULL” is an exception made to more precisely track the data file format in the IANA registry, normally it would have been <control-0000>. Note that this exception would appear in a comment.

```
# character properties for this version. Where RFC5892 or its successor
# RFCs define overrides to the derivation, these are applied as of the
# time of publication. The following values are possible:
#
#  PVALID
#  DISALLOWED
#  UNASSIGNED
#  CONTEXTO
#  CONTEXTJ
#
# Format:
# <code point/range>;  <IDNA_property> # name / name range
#
# Code point ranges are interrupted only at the change of IDNA property
# value, and may cross script, block and general category boundaries.
# All Unicode code points are given a value, there are no missing items
#
# The comments use “NULL” as the name for U+0000 to match the convention
# used in the context of RFC5892. For the same reason, property values
# are in all caps.
#
# See UTS #46, Unicode IDNA Compatibility Processing, for more information.
#
```

Proposed process:

The reason that the IETF has not released more recent versions of this data is that IETF performs or intends to perform a review after each Unicode revision in case some incompatible property assignments result in IDNA property values that either cause an incompatible change to an earlier IDNA property for the same code point or assigns problematic values to new code points.

We suggest that this derived file be published any time a new version of Unicode (and therefore UTS#46) is published and be made available for public review ahead of release. We believe this would facilitate speedier IETF review or speedier adoption of new Unicode characters for implementations supporting IDNA that had relied heretofore on the convenience of the IANA registry data.

The release process should contain a mechanical check[2] flagging any change in property values between PVALID, DISALLOWED and the CONTEXT... values, as these would represent incompatible changes in IDNA_property value compared to earlier versions. If UTC decides that the underlying changes to general category values causing these changes to the derived property are unavoidable, the liaison to IETF should be notified.

Note:

Derived files for earlier versions of Unicode exist and can be placed into a suitable archival location.

[2] Because this is a derived property, an equivalent check may be placed on the input values of the derivation. The goal should be to alert IETF at the earliest possible point in case of incompatible changes