

Introduction to Normalization and Modern Collation

Roozbeh Pournader
Sharif FarsiWeb, Inc.

roozbeh@farsiweb.info

The gap that needed filling...

- For compatibility reasons, Unicode has more than one way to encode things:

$$\text{Ä} \equiv \text{A} + \text{¨}$$

$$\text{ó + ,} \equiv \text{o + , + ´} \equiv \text{o + ´ + ,}$$

- Unicode requires treating them as the same
- But how can one find about equal strings? Through equivalence tables

Canonical Equivalence

- Unicode data file
 - Canonical decomposition: $\ddot{A} = A + \text{¨}$
 - Combining class: ¨ = top center,
, = bottom center attached

Canonical Equivalence

- Algorithm

1. Decompose everything: $\ddot{A} \rightarrow A + \ddot{}$
2. Sort marks according to their combining class:

$$0 + \text{,} + \text{' } \rightarrow 0 + \text{' } + \text{,}$$
$$0 + \ddot{} + \text{' } \neq 0 + \text{' } + \ddot{}$$

Compatibility Equivalence

- For more loose equivalence:

$$\mathbb{R} \cong \mathbb{R}$$

$$\frac{3}{4} \cong 3 + / + 4$$

- The algorithm is the same, only the data comes from a different column of Unicode data files

But I don't want to do that!

- We understand!
- We can ease your pain!
- Normalization forms: NFC, NFD, NFKC, NFKD
- Required for W3C standards like XML, IDN

How to do it?

- It's not trivial
- It's important that you do it 100% conformingly
- Use existing tools and libraries (charlint, GNOME's glib, ...)
- If you really want to do it yourself, pass the test suite
- It's all available here:
<http://www.unicode.org/reports/tr15>

How to use it?

- For XML data, make sure it is in NFC before you pass it on
- For your own software, add input and output normalization filters: this helps a lot in Unicode compliance
 - This means everywhere (character set converters, display engines, sorting engines, text editors, ...)

Questions on Normalization



What is “collation”?

- This is sorting:
me, you, him, her → her, him, me, you
- This is collation:
me ? you
me < you
- You can do sorting using whatever algorithm (*The Art of Computer Programming, Volume 3, Sorting and Searching*)
- Collation is mainly linguistic

Collation should be localized

- One order is not good for all languages:
 - Swedish: z < ö, German: ö < z
 - Arabic: ه < و, Urdu: و < ه
- One order is not good for all uses:
 - German dictionary: öf < of, German phonebook: of < öf
- People still need to customize:
 - Oxford: a < A, Cambridge: A < a

Collation standards

- There are standards you *must* follow:
 - ISO/IEC 14651: International string ordering and comparison (GNU/Linux uses that through glibc)
 - UTS #10: Unicode Collation Algorithm (Java and Mac OS use that through ICU)
 - Microsoft uses a third unknown way (but should generally follow the same model)

Collation standards

- They follow the same model, even are *mathematically equivalent*
- ISO/IEC 14651 specifies a way to customize (tailor), UTS #10 doesn't
- ICU has a more powerful tailoring mechanism

The Collation Model

- Comparison Levels
 - L1, base characters: role < roles < rule
 - L2, Accents: role < rôle < roles
 - L3, Case: role < Role < rôle
 - L4, Punctuation: role < “role” < Role
- Canonical Equivalence
 - Equivalent strings should collate equally

The Collation Model

- Contextual sensitivity
 - Slovakian: $H < Z$, but $CH > CZ$
 - English: $OE < \text{Œ} < OF$
 - Thai: pre-reordering
 - French: accents sorted backward
 - Urdu: پ < بھ < ب

The Collation Model

- Customization
 - Case ordering: optional or mandatory
 - User-defined rules: “?” = “question mark”
 - Merged tailoring: French for Latin, Urdu for Arabic
 - Script Order: Devanagari before Latin
 - Numbers: A-2 < A-10

Common misperception

- No relation to character sets or their code point order
- No relation to code point (binary) order
 - DON'T NAG TO UNICODE ABOUT THIS, since we can't do anything about it
 - Even English doesn't work that way:

Z < a

Common misperception

- The language of the strings is not considered
- No relation to “stable” sort (although it can be “semi-stable”)
- The order is not fixed, things may get changed, things may get added

Mathematics of Collation

- $X =_1 Y$: X is primarily equal to Y
- $X <_2 Y$: X is secondarily less than Y
- Collation Element:
 - 0300 (´) : [0000.0021.0002]
 - 0061 (a) : [06D9.0020.0002]
 - 00E1 (á) : [06D9.0021.0002]
 - 0062 (b) : [06EE.0020.0002]
 - 0042 (B) : [06EE.0020.0008]
 - 0063 (c) : [0706.0020.0008]
 - 0064 (d) : [0712.0020.0002]

Complex Collation Elements

- Expansion:
 - 00E6 (æ) : [06D9.0020.0002] ,
[073A.0020.0002]
- Contraction:
 - 0063 0068 (ch) : [0707.0020.0002]
- Backward accents
- Rearrangement (Thai, Lao)
- Variable Weighting (don't use it)

The Stability Issue

- Stable sorting:

$c, a_1, b, a_2 \rightarrow a_1, a_2, b, c$

- Semi-stable sorting:

- If two strings are not equivalent, you should prefer one

Tailoring

- Unicode and ISO/IEC 14651 provide a default ordering: *intentionally* bad for everybody
- You can only fix that if you're the only user of your script or every user agrees on all specifics
- Ignore it if you're not alone (Latin, Cyrillic, Arabic, Devanagari, ...)

Tailoring

- ISO/IEC 14651 provides a clear syntax
- UTS #10 implementations may have different syntaxes
 - But ICU syntax is OK
 - It's also easier and possibly more powerful, although not well-documented

Questions on Collation

