

Motivation

These punctuation marks are often typed and used interchangeably, and it is surprising to users when they sort far apart and don't match in text search.

For example, due to the primary differences, the current order sorts names like this (alternate=non-ignorable):

- O'Connor
- O'Neill
- O'Shaughnessy
- O'Connor
- O'Neill
- O'Shaughnessy

The proposed sort order, with secondary differences, would be:

- O'Connor
- O'Connor
- O'Neill
- O'Neill
- O'Shaughnessy
- O'Shaughnessy

Similarly, in Hebrew, the current sort order is:

- רמב"ם
- רמב"ן
- רמב"ם
- רמב"ן

Proposed:

- רמב"ם
- רמב"ם
- רמב"ן
- רמב"ן

In [text search based on the UCA](#), even using a very loose search mode (strength=primary), searching for one of these name variants will not find the other.

Details

[L2/22-124 UTC #172 properties feedback & recommendations](#) p. 30 item Coll1: Hoist Hebrew tailoring from CLDR into DUCET

→ **[172-A103] Action Item** for Markus Scherer, PAG: Continue the discussion about the desired sort order of Geresh and Gershayim in CLDR and in the DUCET. See [L2/22-124](#) item Coll1.

This proposal includes both “high” and “low” quotation marks. While these may not look similar, they share the ASCII characters as commonly used fallbacks. Consensus in PAG discussion was to include the “low” marks.

For the [Hebrew](#) and [Yiddish](#) languages, CLDR modifies the sort order so that the Hebrew characters only have a small (diacritic-like, “secondary”) difference from their ASCII look-alikes.

The Chrome browser's find-in-page search (ctrl+F) uses a UCA-based implementation with strength=primary. However, *Chrome hacks the pattern string so that the ASCII quotes and some of their look-alike characters match*. It does this by replacing each Geresh or single quotation mark with the ASCII apostrophe, and each Gershayim or double quotation mark with the ASCII double quote. Details: [CLDR-15946](#)





Given the strength=primary setting used here, this is equivalent to making these characters primary-equal.

Non-goal

This proposal does not address the similar case of free variation between curly modifier letters and quotation marks, such as:

					
02BB	2018	2019	02BC	201B	02BD
MODIFIER LETTER TURNED COMMA	LEFT SINGLE QUOTATIO N MARK	RIGHT SINGLE QUOTATIO N MARK	MODIFIER LETTER APOSTRO PHE	SINGLE HIGH-REV ERSED-9 QUOTATIO N MARK	MODIFIER LETTER REVERSE D COMMA

And

			
201C	201F	02EE	201D
LEFT DOUBLE QUOTATION MARK	DOUBLE HIGH-REVE RSED-9 QUOTATION MARK	MODIFIER LETTER DOUBLE APOSTROPHE	RIGHT DOUBLE QUOTATION MARK