

Unicode PRI #476—Background

CLDR Keyboards Subcommittee LDML (UTS#35) Part 7: Keyboards

Version	2023-05-15
Editors	Steven R. Loomis, and other CLDR Keyboard Subcommittee members
Timeline	2023-05-15 Draft Spec available: "Technical Preview" 2023-05-15 PRI review period opens 2023-07-15 PRI Review period closes

Table of Contents

Summary and Description of Issue	1
Document for Review	2
Feedback	3
Background	3

Summary and Description of Issue

CLDR Ticket for feedback: [CLDR-16601](#)

This document describes a major revision to the [CLDR LDML specification \(UTS#35, v43\)](#) to support keyboards, and uses data and structure that are consistent with that specification. The [Keyboard-SC](#), consisting of subject matter experts from across the industry, has been working for over 18 months to revise Part 7 of the LDML spec in order to better support keyboard-based text input requirements for the world's languages.

The Keyboard-SC determined that a rewriting of the specification was required, instead of attempting a backwards-compatible improvement. For details, see "[Comparison with v43 LDML Specification](#)".

The CLDR-TC and CLDR Keyboard-SC would appreciate feedback on whether there are specific changes or enhancements that should be made in the proposed specification. See the [Feedback section](#) for details.

Problem

Today, every platform must independently evaluate, prioritize, and implement all new or updated keyboard layouts, leading to major inconsistencies and delays especially where digitally disadvantaged languages are concerned. Consequently, language communities and other keyboard authors must see their designs developed independently for every platform/operating system, resulting in unnecessary duplication of technical and organizational effort.



Goals

“Keyboard 3.0” is designed from the ground up to be usable as a solution to support **both hardware and on-screen (touch) layouts for all platforms in a single source file for each language.**

With Keyboard 3.0, leading members of the language communities will be able to submit their layout once to CLDR, and it will be available to all platforms as part of the latest version of CLDR, making adoption much easier for platforms. Platform vendors will not need to develop and maintain their own keyboard layout data, especially for languages that they don’t yet support.

This work contributes to the goals of the [United Nations International Decade of Indigenous Languages](#) by improving the path for Digitally Disadvantaged Language communities to develop platform support for their languages. Users should see improvements in consistency between platforms, as layouts can be shared.

What this proposal includes

- Proposed Draft Update to the specification, UTS#35 part 7
- XML DTD and XSD schema files
- Sample keyboard layout files and test data

Status

The document is currently open to review and comment by the public.

Document for Review

- The document for review, UTS#35 Part 7 may be accessed at:
 - <https://www.unicode.org/review/pri476/spec/tr35-keyboards.html>
 - <https://www.unicode.org/review/pri476/spec/tr35-keyboards.pdf>

- (Please review this Part 7 as a new document.)
- DTDs and sample data files are available at:
 - <https://www.unicode.org/review/pri476/data/>
- Charts are available at:
 - <https://www.unicode.org/review/pri476/charts/>

Feedback

- Feedback is due by July 15th, 2023
- To give feedback via Jira or to view existing feedback, visit <https://unicode-org.atlassian.net/browse/CLDR-16601> (Sign-in to Jira is required to comment.)

To contribute direct suggestions via GitHub, open a pull request against the [keyboard-preview](#) branch.

Background

History of the LDML Keyboard Format

- In 2012, the original LDML keyboard format was designed to describe keyboards for comparative purposes, such as comparisons of character repertoire and layout. It was also intended to be used as a basis for interchanging keyboard data, but did not achieve that purpose.
- The CLDR Keyboard Subcommittee was formed and has been meeting since mid-2020. It quickly became apparent that the v43 LDML format was insufficient for implementing new keyboard layouts.
- The v43 CLDR specification required revision in order to be more useful to implementers.

Comparison with Previous [v43 LDML Specification](#)

A number of deficiencies in the v43 specification necessitated an incompatible change to the specification.

- The v43 key specification contained platform-specific details such as longPress-status and dead-key settings.
- The v43 key layout model was designed around physical ISO key locations such as A01, A02, etc. This is problematic for touch layouts which require a relative layout not associated with hardware locations. The proposed model is designed around a "bag" of reusable keys, each with a unique identifier and specific Unicode output. The keys are arranged either for on-screen use (for virtual/touch layouts) or according to one of the industry standard hardware layouts.

- The v43 layer model was oriented around modifier keys, and did not allow for "virtual" layers needed by touch layouts. The v43 layout model also did not specify key width, positioning, or spacing.
- The v43 transform specification was found to be ambiguous in its output and thus challenging to implement. The specification for transforms was extensively rewritten, the model was changed significantly to match expectations and experience from various platforms and keyboard designs. The new transform specification is largely a subset of common regular expression syntaxes, so that existing regular expression engines may be used to implement the LDML specification. In particular, look-ahead ("after") has been removed.
- The v43 specification separated layouts into per-platform directories with separate layout information. By contrast, this proposal specifies single cross-platform files representing each layout.
- The v43 specification was inconsistent in naming and presented challenges to authoring by language experts. Most of the v43 data files were extracted mechanically. This proposal gives consideration to authoring by language experts. Elements and attributes were renamed in order to be more self-consistent and easily understood.

Backwards incompatibility is not expected to be a problem as the proposed specification has a different focus. Implementations that need to use the previous format may continue to use CLDR version 43 and prior for keyboards.

Review Note: Due to the extensive nature of changes, a textual comparison with the [v43](#) specification is not included in this review document. Please review Part 7 in its entirety.

What is out of scope for this proposal

- This revision is intentionally not backwards compatible with v43 and prior versions of the CLDR LDML Keyboard specification, due to a major change in focus. [See the “Comparison” section, below.](#)
- Input Method Editors (IME) (such as used for Han ideographs) or other input methods (such as handwriting or chorded input) are not currently in scope for this format. Such methods often require the display of multiple possible alternatives.
- Localized names for keycaps (e.g., the German name for "Return") are out of scope for this section of LDML. They would belong in Part 2.
- Platform-specific keys such as Fn, Numpad, IME swap keys, and cursor keys are out of scope. This proposal covers only the character-emitting keys.
- Round-tripping of existing platform file formats is not guaranteed. Existing formats may require features which are not supported by this format.
- It is not a goal for this format to replace pre-existing platform layouts. It is intended for newly-created or newly-updated layouts. That is, a platform could either convert the data into their own format, or support this format natively.