

# Analysing Support for Text Layout on the Web

Richard Ishida  
Addison Phillips

# W3C Internationalization Initiative

W3C's goal is a Web for All, regardless of language, script or culture. The Web community has made tremendous progress in internationalizing the Web over recent decades, but as Web penetration in language communities increases, as usage scenarios grow, and as new applications such as digital publishing emerge, there remains more to do.

For the Web to truly work for stakeholders all around the world, there must be a collaboration of language experts, Web site designers, developers, and vendors who are active in moving the Web forward. To ensure a rapid response to the growth of the Web, the W3C wants to marshal the resources of organizations and experts who care about these problems and enlist their help in strengthening internationalization support for the Web.

To accelerate progress in this area, the W3C is also looking to supplement the core funding it receives from W3C Member fees so that it can increase in-house resources dedicated to this work.



The internationalization initiative will provide participants and funding to address three main aspects of the internationalization continuum:

- **Language enablement** appeals most directly to stakeholders (e.g., governments, publishers, community groups, etc.) who utilize the language.
- **Developer support** appeals most directly to tech companies that are building the infrastructure for a global Web and supporting W3C standards groups.
- **Author support** appeals to people creating Web content in their own language, as well as to companies who build or localize Web sites in many languages.

## The Web needs your help

Success in meeting these goals requires participation and funding from language, developer, and author communities, in order to expand the effort over and above what can be achieved with our core funding.



### Language enablement

The W3C wants to ensure that local requirements for language support on the Web are identified and addressed. Text layout is an area of particular interest, and is concerned with things such as rules for line-breaking & justification, local approaches to expressing emphasis or decorating text, localizing counter styles, supporting bidirectional text in markup, initial-letter styling, hyphenation, page layout, and so on. These typographic conventions are often very different from the Western norm in languages that use writing systems such as Arabic, Devanagari, Thai, Mongolian, and so forth.

In Web pages and in digital publishing, needs can be addressed by improving W3C standards for rendering text (such as CSS, WebVTT, SVG, etc.), and markup (such as HTML). The goal here is to ensure that the Web supports the native typographic features that users around the world are used to, and enables users to interact with the Web in line with long-standing print traditions. The [language matrix](#) captures an overview of where work is needed.

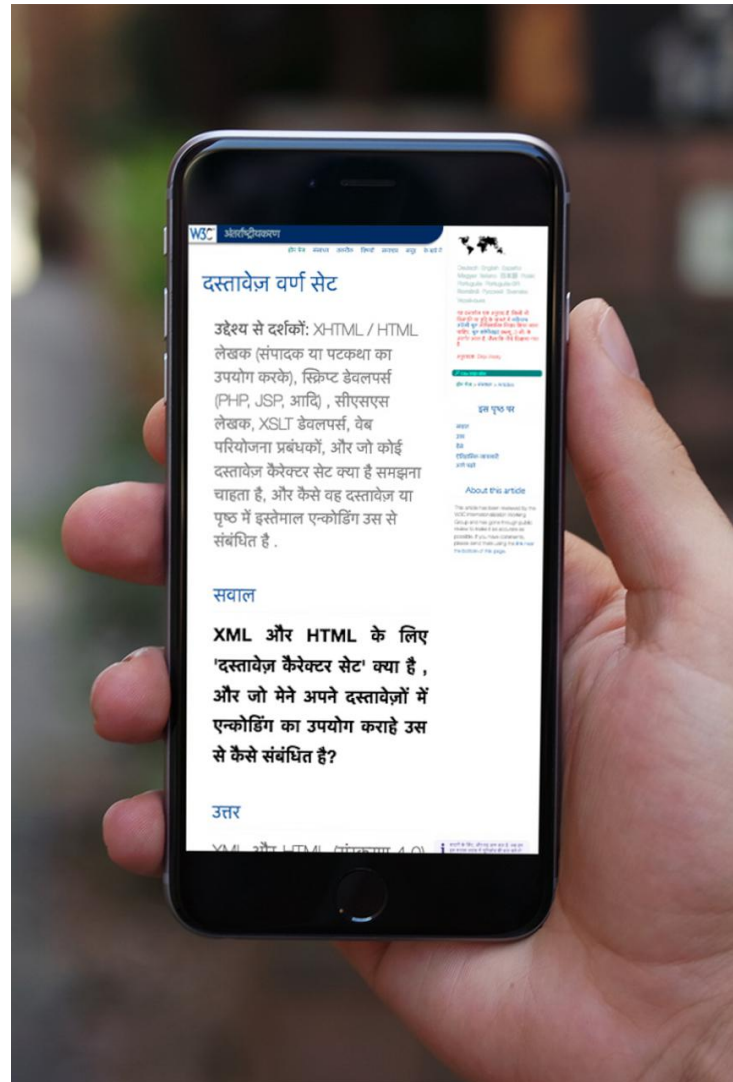
To achieve this goal, the W3C needs to assess current support for the world's languages on the Web, identify gaps, prioritize them, develop requirements, and then take steps to close the gaps. To do so, it needs to establish a network of experts who can advise on language-related requirements, and increase resources available to facilitate the work in this area.

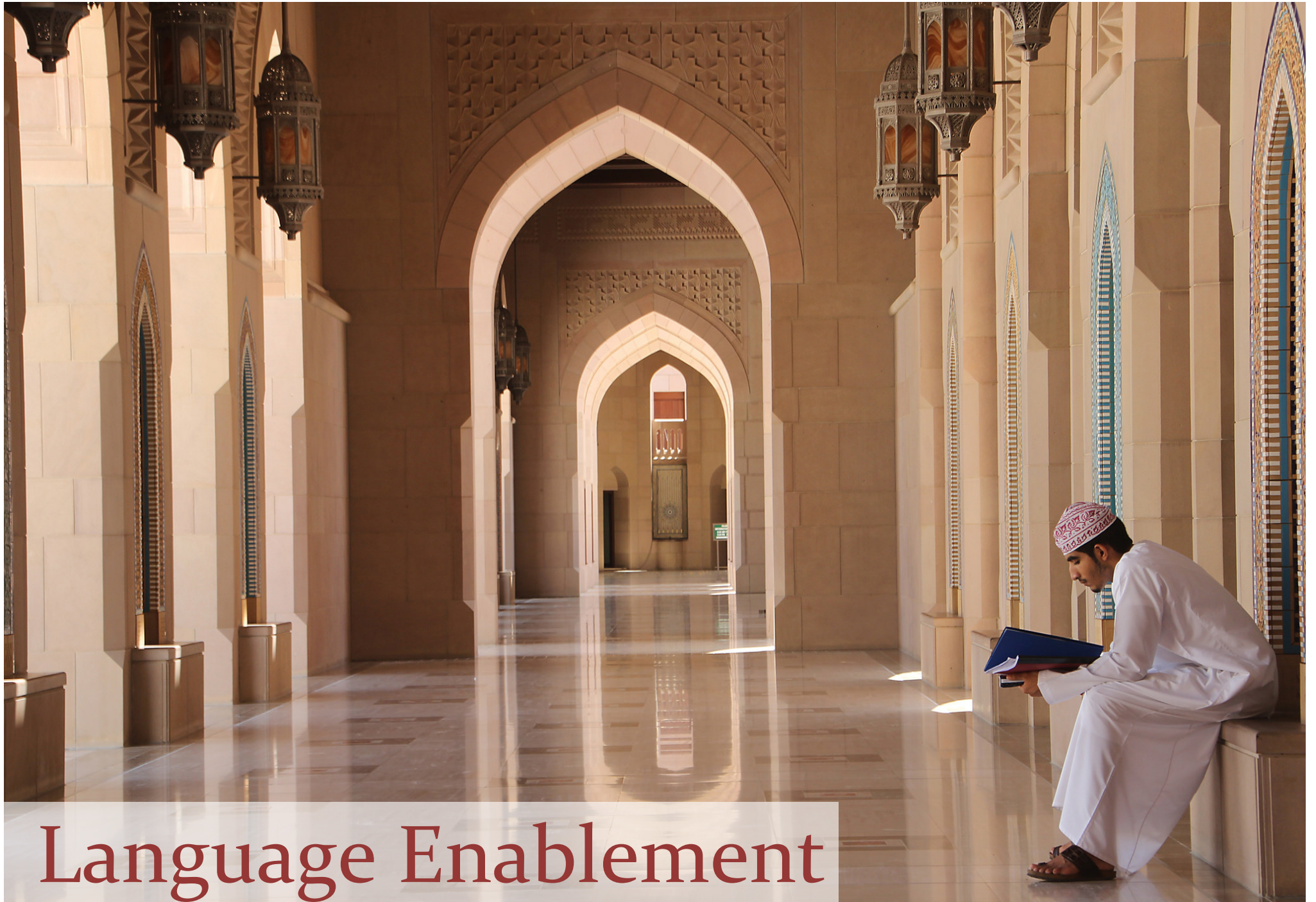
### Developer support

A core focus for the W3C is to support creators of specifications. of system-

# (Web for All)

- **Language enablement**
  - understand where the gaps are for users of the global Web
- **Developer support**
  - build standards & applications that support a global Web
- **Author support**
  - help people create content in their own language, or create content that will be localised





Language Enablement

# language matrix

[w3c.github.io/typography/gap-analysis/language-matrix](http://w3c.github.io/typography/gap-analysis/language-matrix)

script	language	Level	Encoding	Fonts	Font styles	Glyph control	Cursive text	Char transforms	Numbers & digits	Boundaries & selection	Quotation	Inter-char spacing	Ruby	Text decoration	Emphasis & highlight	Bidi text	Other inline	Line breaking	Hyphenation	Justification & alignment	Lists, counters	Initial letter style	Baselines, inline align	Other paragraph	Bidi layout	Vertical text	Notes, footnotes	Page numbers, running heads	Other pagination
Arab	Arabic*	0.15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Urdu*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Persian*	0.15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Pashto*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Sindhi*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Kurdish*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Saraiki*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Ethi	Amharic*	0.55	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Tigrinya*	0.55	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Gujarati*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Punjabi*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Korean*	0.67	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Hani	SChinese*	0.5	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	TChinese*	0.5	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Hebr	Hebrew*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?

For the 79 languages currently listed, we see the following problem areas. This page is updated as information becomes available.

- 33 languages need work for advanced publishing
- 25 languages need work for basic features
- 2 languages don't work well on the Web
- 47% of cells still need investigation.

# line-breaking

	Space as <b>word</b> separator	Other <b>word</b> separator	Syllable separator	No separator
<b>Wraps words</b>	Amharic (ethiopic)*, Arabic, Armenian, Bengali, Cherokee, Dhivehi (thaana), English (latin), English (deseret), Fula (adlam), Georgian, Greek, Gujarati, Hebrew, Hindi (devanagari), Inuktitut (UCAS), Kannada, Korean (hangul)*, Malayalam, Mandaic, Mandinka(n'ko), Oriya, Panjabi (gurmukhi), Russian (cyrillic), Sinhala, Syriac, Tamil, Tedim (pau cin hau), Telugu	Samaritan		Khmer, Lao, Myanmar, Thai
<b>Wraps syllables</b>	Eastern Cham, Korean (hangul)*, Sundanese		Vietnamese (latin), Tibetan	Balinese, Batak, Chinese, Javanese, Western Cham
<b>Wraps characters</b>		Amharic (ethiopic)*		Japanese, Vai

# word, space

Hindi

को छोड़नो और अपने देश को वापस  
आनो का अधिकार है।

this يحق لكل فرد أن يغادر أية بلاد بما في  
is English ذلك بلده كما يحق له العودة إليه.

Arabic

# word, no-space

Khmer

មនុស្សគ្រប់រូបមានសិទ្ធិចាកចេញពីប្រទេស  
ណាមួយ រួមទាំងប្រទេសរបស់ខ្លួនផង និងមាន  
សិទ្ធិរំលត្រឡប់មកប្រទេសរបស់ខ្លួនវិញ។

Thai

ตัวอย่าง|การเขียน|ภาษาไทย  
ตัวอย่าง|การ|เขียน|ภาษา|ไทย





# character

Japanese

す|べ|て|人|は、|自|国|そ|の|他|い|ず|れ|  
の|国|を|も|立|ち|去|り、|及|び|自|国|に|  
帰|る|権|利|を|有|す|る。|

きょう

# ambivalent

Korean

전|세|계|의 |월|드 |와|이드 |웹|으로 |만|들  
기!

전|세|계|의 |월|드 |와|이드 |웹|으로 |  
만|들|기!

Ethiopic

እያንዳንዱ ሰው ከማንኛውም  
አገር ሆነ ከራሱ አገር ወጥቶ  
እንዲገናኝ ወዲያ አገሩ የመመለስ  
መብት አለው።

እያንዳንዱ፡ ሰው፡ ከማንኛውም፡  
አገር፡ ሆነ፡ ከራሱ፡ አገር፡ ወጥቶ፡  
እንዲገናኝ፡ ወዲያ፡ አገሩ፡ የመመለስ፡  
መብት፡ አለው።

# special rules (CJK)

すべて人は、自国その他  
いずれの国をも立ち去り、  
及び自国に帰る権利を有  
する。



すべて人は、自国その他  
いずれの国をも立ち去  
り、及び自国に帰る権利  
を有する。



# special rules (Tibetan)

ལྷོ་དང་འདུན་པ་བཟང་བའི་རང་རིགས་ཀྱིས། །རྒྱལ་  
ཁའི་འཕྲིན་བཟང་ལས་དོན་འགྲུབ་ཕྱིར་འབད།



ལྷོ་དང་འདུན་པ་བཟང་བའི་རང་གས་  
ཀྱིས། །རྒྱལ་ཁའི་འཕྲིན་བཟང་ལས་  
དོན་འགྲུབ་ཕྱིར་འབད།







# line-breaking

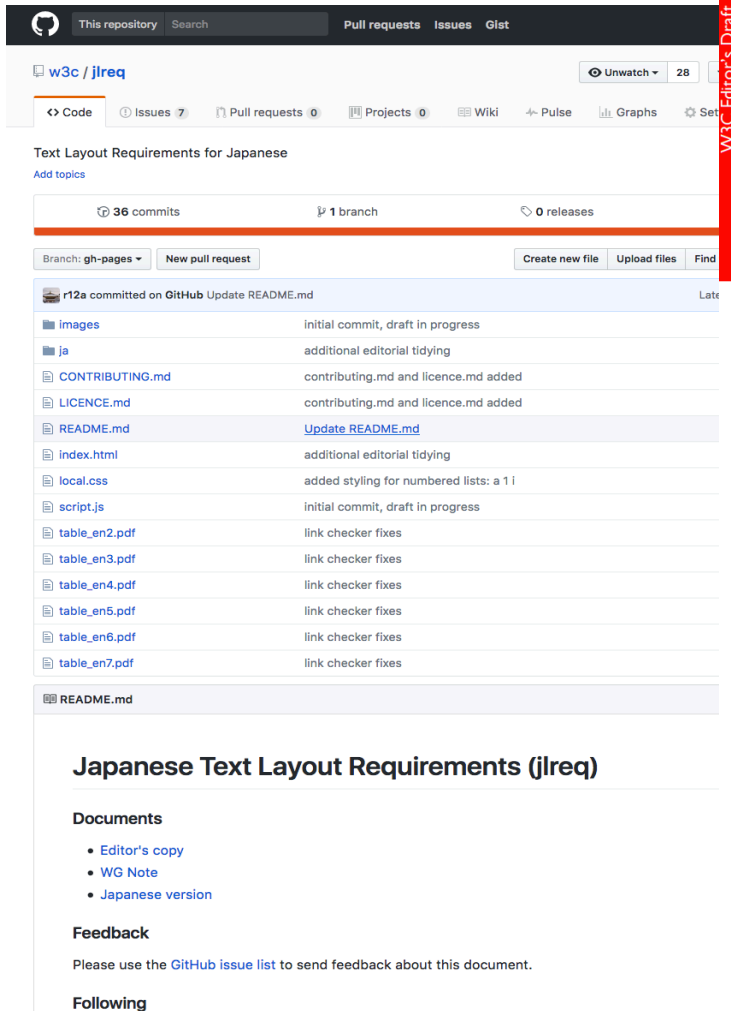
[w3.org/international/articles/typography/linebreak](http://w3.org/international/articles/typography/linebreak)

	Space as <b>word</b> separator	Other <b>word</b> separator	Syllable separator	No separator
<b>Wraps words</b>	Amharic (ethiopic)*, Arabic, Armenian, Bengali, Cherokee, Dhivehi (thaana), English (latin), English (deseret), Fula (adlam), Georgian, Greek, Gujarati, Hebrew, Hindi (devanagari), Inuktitut (UCAS), Kannada, Korean (hangul)*, Malayalam, Mandaic, Mandinka(n'ko), Oriya, Panjabi (gurmukhi), Russian (cyrillic), Sinhala, Syriac, Tamil, Tedim (pau cin hau), Telugu	Samaritan		Khmer, Lao, Myanmar, Thai
<b>Wraps syllables</b>	Eastern Cham, Korean (hangul)*, Sundanese		Vietnamese (latin), Tibetan	Balinese, Batak, Chinese, Javanese, Western Cham
<b>Wraps characters</b>		Amharic (ethiopic)*		Japanese, Vai



# jlreq

w3c.github.io/jlreq/



W3C Editor's Draft

Japanese Text Layout Requirements (jlreq)

Documents

- Editor's copy
- WG Note
- Japanese version

Feedback

Please use the [GitHub issue list](#) to send feedback about this document.

Following

The following are the basic elements of a page format. [Fig. 11](#) illustrates an example of a page format in vertical writing mode.

- Trim size and binding side (vertically set Japanese documents are **bound on the right-hand side**, and horizontally set documents are **bound on the left-hand side**. See [Fig. 12](#).)
- Principal **text direction** (vertical writing mode or horizontal writing mode).
- Appearance of the **kihon-hanmen** and its position relative to the trim size.
- Appearance of **running heads** and **page numbers**, and their positions relative to the trim size and kihon-hanmen.

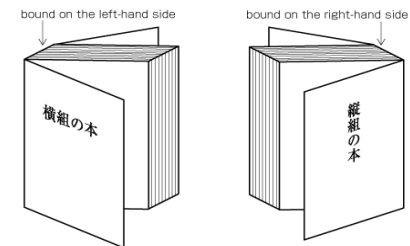


Fig. 12: Binding-side (bound on the right-hand side and bound on the left-hand side).

## NOTE

Establishing a **kihon-hanmen** may be seen as defining not only a rectangular area on a page, but also within that area an underlying, logical grid, to guide the placement of such things as characters, **headings**, and **illustrations**. However, once a kihon-hanmen is established, there is no absolute requirement to align characters with the grid, especially when setting characters inside a line. The only factors that influence the placement of characters are strong gravitational forces that (i) attract the first and last characters on a line to align with the border of the kihon-hanmen, and (ii) attract each line position to the line positions on which the kihon-hanmen is based.

It may help in understanding the basic concepts of Japanese layout and kihon-hanmen to think in terms of a slit-based model, rather than a grid-based model. Each slit is the full length of the lines on which the kihon-hanmen is based.

## 2.2.4 Elements of Kihon-hanmen

The **kihon-hanmen** is the hanmen style designed as the basis of a book. The following are the basic elements of the kihon-hanmen (see [Fig. 13](#)).

- Character size** and typeface name
- Text direction (vertical writing mode or horizontal writing mode)



# klreq

# clreq

ko en

## 2.3 'Letter Face Position in Character Frame' Standard | '글자틀 내 글자면 위치' 표준

1 ReSpec

Standardization of 'letter face position in character frame' of fixed width Hangul fonts improves the compatibility of the space between Hangul font characters. (The relation between each side's spaces remains even when the Hangul font is changed. It prevents a paragraph's left outline being scattered when the opening quotation mark or parenthesis at the line head has an unexpected space).

고정폭 한글 폰트의 '글자틀 내 글자면 위치'를 표준화는 한글 폰트 간의 글자사이 비율 호환성을 향상하기 위함이다.(한글 폰트를 변경해도 문장부호의 좌우 여백 관계가 그대로 유지되도록 한다. 글줄 시작에 위치한 열기 괄호·따옴표에 의도치 않은 여백이 생겨 단락의 왼쪽 외곽선이 흐트러지는 경우를 방지한다).



Fig. 3: Letter face position in the character frame | 글자틀 내 글자면 위치

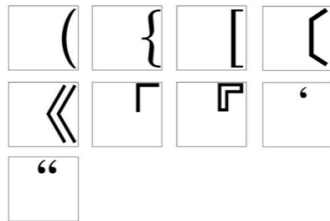
### ISSUE 3

See <https://github.com/w3c/klreq/issues/10>

### 2.3.1 Arrangement of 'Letter Face Position in Character Frame' for Full Width Parentheses | 전각 괄호의 '글자틀 내 글자면 위치' 지정

In horizontal writing, the letter face of a full width opening parenthesis is placed on the right end of the character frame, and the left space is considered a user controlled area. In vertical writing, the letter face of a full width opening parenthesis is placed on the bottom end of character frame, and the space is considered a user controlled area.

가로짜기의 경우 전각 열기 괄호의 글자면은 글자틀의 오른쪽(가로짜기)-아랫쪽(세로짜기) 끝에 두고, 왼쪽 여백은 사용자 조정 영역으로 간주한다. 세로짜기의 경우 전각 열기 괄호의 글자면은 글자틀의 아랫쪽 끝에 두고, 여백은 사용자 조정 영역으로 간주한다.



zh-hant zh-hans en

## 1. Bilingual Annotations.

1 1 ReSpec

### 中外文对照

### 中外文對照

Bilingual annotations aim to provide a Chinese translation of text in foreign languages or acronyms, or to offer the original text for words that have been translated into Chinese. This is mainly used for proper nouns, titles or those terms whose concepts are difficult to convey after translation. It is commonly found in translated works, mainly in light novels.

为外来语、首字母缩略词标注其中译，或对翻译名词标注其原文，多见于专有名词、作品名及译后概念较难传达的词汇。常见于译作，尤以轻小说为主。

為外來語、首字母縮略詞標注其中譯，或對翻譯名詞標注其原文，多見於專有名詞、作品名及譯後概念較難傳達的詞彙。常見于译作，尤以轻小说為主。

巧合的是，<sup>Keith Emerson</sup>基斯·愛默生彈奏的賦格部分這時剛好結束，曲子正要進入<sup>無盡的謎</sup>〈Endless Enigma〉的第二樂章。  
巧合的是，<sup>Keith Emerson</sup>基斯·愛默生彈奏的賦格部分這時剛好結束，曲子正要進入<sup>無盡的謎</sup>〈Endless Enigma〉的第二樂章。

Fig. 7: An example of positioning for bilingual annotations. | 中外文對照行間注的排版示例。 | 中外文對照行間注的排版示例。

## 2. Interlinear Comments.

### 行间批语

### 行間批語

Interlinear comments are ways to annotate the meaning of text fragments or a single word, and are so named for their interlinear positioning. They usually lie in the interlinear space and co-exist with the body text. Compared to other annotation methods, i.e. headnotes or footnotes, interlinear comments are more compact and stick better to the body. These kinds of comments are often found in ancient books, such as Rouge Inkstone, an early commentary of the novel Dream of the Red Chamber.

行间批语是为一段文本片段或单个词汇标注解释的排版方式，因共存于正文文本，显示于其行间而得名。行间批语比眉批、脚注等注释方法更具紧凑、依附性，多见于如《红楼梦》早期抄本的脂砚斋批语等古籍。

行間批語是為一段文本片段或單個詞彙標注解釋的排版方式，因共存於正文文本，顯示於其行間而得名。行間批語較眉批、腳注等注釋方法更具緊湊、依附性，多見於如《紅樓夢》早期抄本的脂硯齋批語等古籍。

# alreq

# elreq

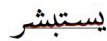
### 2.6.3 Arabic Script and Typography

1 ReSpec

Arabic script has some characteristics that are challenging for typographers and font designers. Examples below show some characteristics worth to be considered carefully. How could typography, which came late to the Arabic world, then follow the tradition of the many authors and artists who manually shaped the Arabic writing over decades? even in it's simplest *Naskh* style?

#### 1. Multi-level baselines

Letters may join through a finely inclined line



or two, square-ended lines



Multilevel baselines don't occur in all fonts. The above examples use the Arabic Typesetting font. Compare those examples to to more typical fonts:



#### 2. Multi-context joining

Rendering of letters depends not only on their place in the word (initial, medial, final) but also on their neighboring letters, i.e. the letter they join with. Each letter has a different appearance in each combination.



Fig. 18: Initial letter noon, showing many different forms.

Fonts don't always comply with or respect this kind of 'tuning'. To do so, fonts need many glyphs in order to adapt to each context. In more modern typefaces some of these connections are implemented by ligatures, but ligatures can't capture or cover all joining behavior.

In the two left most words, the initial noon differs in that one raises a kind of stroke. This property of raising a stroke is common for a number of letters (beh, teh, noon, theh) which are taller than their connected letters in order to be distinguished in some contexts, such as *بسل* vs. *سبل*, or to resolve ambiguity. See also the section about teeth letters below.

#### 3. Words as groups of letters

A word shape is not (only) a "horizontal" connections of letters, but of groups of letters (syntagmes).

Example two words in some nice Naskh font.



### 3.3.2 Justification When ETHIOPIC WORDSPACE is the Word Delimiter

1 ReSpec

Since the arrival of the printing press in Ethiopia in 1863 (Pankhurst, 1998), full justification of Ethiopic has been a common typesetting practice in Ethiopian, and later Eritrean, publishing houses. Earlier, Ethiopic justification rules are a feature of Hiob Ludolf's *Historia Aethiopia*, which is noted as the first use of movable type for Ethiopic script (Ludolf, 1861). Prior to letterpress typography, calligraphic manuscripts rendered on parchment also featured full, or approximately full, justification. Though the latter likely reflects the scribe's desire not to waste a millimeter of available lateral writing space.

The placement of Ethiopic wordspace presents a complication to the justification of Ethiopic text. Two placement styles developed in typeset literature which will be referred to here as "word bound" and "centered" styles. Additionally, the word spacing following an Ethiopic fullstop may (or may not) be governed by a special rule and in combination with the two wordspace spacing styles. These spacing rules are discussed in the following sections.



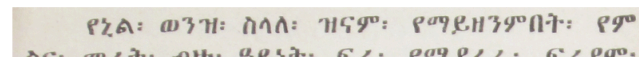
Fig. 25: Ethiopic Justification in *Historia Aethiopia* (Ludolf, 1681)

### 3.3.3 Justification with Word Bound Wordspace and Punctuation

In keeping with line justification for Latin script, the non-printed or "blank space" (space and gaps) between words is treated as stretchable. The width of the space symbol itself will be elongated to some aesthetic width value that may vary from space symbol to space symbol across a printed line. In Ethiopic justification, the blank space between the Ethiopic word separator and the words it separates is likewise allowed to stretch. This stretching of blank space may be either symmetrical ("centered") or asymmetrical but in the latter case space stretching is always between the right side of the separator and the following word –referred to here as "word bound".

In "word bound" justification the word separator, which may be either a punctuation symbol or U+1361 ETHIOPIC WORDSPACE [፡], appears to adhere to the word to the left as if it were its final character. Figures Fig. 26:

Ethiopic justification in word bound style (Erikson, 1921 (1913 EC)) and Fig. 25: Ethiopic Justification in *Historia Aethiopia* (Ludolf, 1681) both illustrate the word bound style.



# ilreq/IIP

# tlreq/mlreq

### 3.3 Initial letter styling

1 ReSpec

Drop initial is a typographic effect emphasizing the initial letter(s) of a block element with a presentation similar to a 'floated' element.

#### 3.3.1 Selecting initial letters

Initial letters in Indic scripts must be selected on the basis of orthographic syllables, rather than individual letter forms (see an example at the end of section 3, Text segmentation). A detailed definition of Indic syllables can be found in section 2, Indic Syllable boundaries. In Indian languages the size of the Initial Letter is determined by the number of the lines between top line of the syllable and lowest bit in the orthographic Indic syllable cluster where subjoined consonant and other diacritics appears.

#### 3.3.2 Typical drop initial usage in Indic scripts

Most of the Indic drop initial letters in magazines and newspapers use 2 to 4 line drops. Some examples are shown below.



Fig. 4: Examples of Indic Initial letters

The Sunken and raised Initial letter are not preferred in Indian languages. In examples of this kind, reference points on the drop cap must align precisely with reference points in the text. In Indic scripts the top reference point is the hanging base line for those scripts that have one, and the bottom alignment point is the text after-edge.

Initial letter wrap property is not applicable for Indian languages. No contour-filling is required in Indian languages.

Alignment of the top line of the non-highlighted characters is at the top of the thicker top line of the initial letter is commonly used in India. In some examples top lines of the initial letter and the following letters don't touch. This is due to variable technology/formats used by the publishers. It is preferred that both the top lines of initial letter and neighbouring text should touch. Here are some additional examples of initial highlighted letter and drop letter based on the Indic syllable definition.



zh en

### 2.3 Tibetan Syllables | 藏文音节

2 ReSpec

Word boundaries within a section are not indicated, only 'syllables', known as tsheg-bar /tsek bar/. Syllable boundaries are usually separated by the tsek character, U+0F0B TIBETAN MARK INTER-SYLLABIC TSHEG .

The pronunciation of Tibetan words is typically much simpler than the orthography, which involves patterns of consonants. These patterns reduce ambiguity and can affect pronunciation and tone.

The following diagram shows characters in all of the syllabic positions, and lists the characters that can appear in each of the non-root locations. The two-syllable word in the example is འགྲེམས་རྩོད་ 'grem-ston /d̥ɛm-ton/ (exhibition).



Fig. 6: Syllable composition in Tibetan

#### 2.3.1 Structural Rules | 结构规则

The primary consonant in a syllable is called the root consonant (or radical) (མིང་བཞི), and the other consonants in the syllable ( normally up to 6 in total) annotate or modify it. The following rules help identify the root:

要分析藏文结构必须先找出根字母，然后其他的部分根据结构规则就能找到。根字母的判断方法如下：

- 1. A consonant with a vowel is always the root, unless it is the phrase connector འི, and letters with superscripts or subscripts are root consonants.

一个辅音上有元音字母，那就是根字母，除非是 འི 如下上面是元音字母下面是辅音字母，在此中是根字母。



- 2. In a 2-consonant syllable with no vowel, the first consonant is always the root.

一个辅音上有上标字或者下标字那么这个辅音字母也是根字母。



# gap analysis

## TABLE OF CONTENTS

<b>1.</b>	<b>Introduction</b>
1.1	Work flow
1.2	Prioritization
<b>2.</b>	<b>Characters and phrases</b>
2.1	Encoding considerations
2.2	Fonts
2.3	Font styles
2.4	Glyph control
2.5	Cursive text
2.6	Quotations
2.6.1	Default quotation marks for q element
2.6.2	Default quotation marks in a new language section
2.6.3	Embedded quotations in a different language
2.7	Numbers, dates, etc.
2.8	Text boundaries & selection
2.9	Transforming characters
2.10	Inter-character spacing
2.11	Ruby annotation
2.11.1	Tabular markup
2.11.2	Double-sided ruby
2.11.3	Ruby alignment and other styling
2.12	Text decoration
2.13	Emphasis & highlights
2.13.1	Boten marks
2.14	Bidirectional text
2.15	Other inline features
<b>3.</b>	<b>Lines and Paragraphs</b>
3.1	Line breaking
3.2	Hyphenation
3.3	Justification
3.4	Counters, lists, etc.
3.5	Initial letter styling
3.6	Baselines & inline alignment
3.7	Other paragraph features
<b>4.</b>	<b>Layout &amp; pages</b>
4.1	Bidirectional layout
4.2	Vertical text
4.2.1	Vertical form controls
4.2.2	Upright text orientation
4.2.3	Tate-chu-yoko
4.2.4	Lists
4.2.5	Table cells
4.2.6	Logical properties
4.3	Notes, footnotes, etc.
4.4	Page numbering, running headers, etc.

## Japanese Gap Analysis

W3C Editor's Draft 12 June 2018



### This version:

<https://w3c.github.io/alreq/gap-analysis/>

### Latest published version:

<https://www.w3.org/TR/alreq-gap/>

### Latest editor's draft:

<https://w3c.github.io/alreq/gap-analysis/>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

[Richard Ishida \(W3C\)](#)

### Github:

[repository](#)

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Abstract

This document describes and prioritises gaps for the support of Japanese on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and ereaders. **This is a preliminary analysis.**

## Status of This Document

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.*

This document describes and prioritises gaps for the support of Japanese on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and ereaders. This document complements the document [Text Layout Requirements for the Japanese Script](#), which describes the requirements for areas where gaps appear. It is linked to from the [language matrix](#) that tracks Web support for many languages.

This document is currently an individual contribution.

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). Only send comments by email if you are unable to raise issues on github (see links below). All comments are welcome.

To make it easier to track comments, please raise separate issues or emails for each comment,

# gap analysis

## 2.11.2 Double-sided ruby

In order to get double-sided ruby to appear on either side of the base text in HTML, you need the CSS `ruby-position` property to work with the value `under`. It currently works per the standard only with Firefox. It works with the proprietary `-webkit` prefix in Chrome and Safari, but doesn't work with Edge. [See test results.](#)

Double-sided ruby doesn't appear often in Japanese text, but it is used, and should be available to content authors.

## 2.11.3 Ruby alignment and other styling

For background information read [Aligning annotations and bases.](#)

Fine control in HTML over the placement of ruby annotations relative to the base require the availability of the CSS property `ruby-align`. This only works in Firefox, and sometimes in Edge with proprietary syntax. [See results for major browsers.](#)

The impact of this is probably more important for advanced typographic layout.

Needs work for basic styling support.

## 2.12 Emphasis & highlights

*Bold and italic are not always appropriate for expressing emphasis, and some scripts have their own unique ways of doing it, that are not in the Western tradition at all. Does this script require support for emphasising or highlighting text that cannot be achieved currently? [See available information](#) or [check for currently needed data.](#)*

For requirements, see JLReq [Composition of Emphasis Dots.](#)

### 2.12.1 Boten marks

There are tests for boten mark support that show that they are not supported by Chrome 63.0.3239.132 or Edge, but they are supported by Firefox 58.0 and Safari 11.0.3.

The expected behaviour for basic support of boten marks is [specified in CSS.](#)

Given that there are alternative ways of showing emphasis, and although there are already two implementations, I mark this as an advanced need.

Additional requirements arose while the text was being written for CSS. They relate to text that is annotated with ruby at the same time as annotation marks. The conclusions were:

1. Emphasis marks go outside ruby.
2. If ruby only spans part of the emphasised word, emphasis marks stay as close as possible to the base.
3. Where a ruby annotation is hidden or empty, the emphasis marks should continue at the same height.

JLReq also requires that emphasis marks not appear over commas, full stops and brackets.

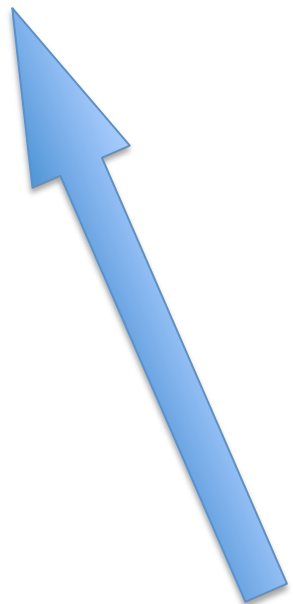
There are currently no tests for these behaviours, and no requirements in jlreq for the former (with ruby). **Need to take a closer look.**

JLReq calls out the usual practise of using sesame shaped boten for vertical text, and bullets for horizontal. Controls for this exist in CSS, and tests show that it is supported by the browsers that support boten.

The default side for boten is to the right of vertical lines and above horizontal lines, according to JLReq. CSS controls allow this positioning to be set by the author, and in fact it is the default in both browsers that support boten for text where a lang tag identifies it as Japanese.

Needs work for advanced level support.

language	Level	Encoding	Fonts	Font styles	Glyph control	Cursorive text	Char transforms	Numbers & digits	Boundaries & selection	Quotation	Inter-char spacing	Ruby	Text decoration	Emphasis & highlight	Bidl text	Other inline	Line breaking	Hyphenation	Justification & alignment	Lists, counters	Initial letter style	Besslines, inline align	Other paragraph	Bidl layout	Vertical text	Notes, footnotes	Other pagination	Page no. s, running heads
																												?
Arab	Arabic*	0.19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Urdu*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Persian*	0.19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Pashto*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Sindhi*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Kurdish*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Saraiki*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Uighur*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Beng	Bangla*	0.35	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Assamese*	0.35	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Manipuri*	0.35	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Chittagonian*	0.46	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Cans	Inuktitut*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Cree*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Cher	Cherokee*	0.78	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Cyrl	Russian*	0.89	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Ukrainian*	0.89	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Uzbek*	0.89	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Azerbaijani*	0.89	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Bulgarian*	0.89	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Serbian*	0.89	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Deva	Hindi*	0.26	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Marathi*	0.26	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Bhojpuri*	0.42	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Maithili*	0.42	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Awadhi*	0.42	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Nepali*	0.42	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Ethi	Amharic*	0.55	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	Tigrifna*	0.55	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Grek	Greek*	0.8	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Guju	Gujarati*	0.26	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Guru	Panjabi*	0.29	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Hang	Korean*	0.67	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Hani	SChinese*	0.5	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	TChinese*	0.5	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Hebr	Hebrew*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Java	Javanese*	0.69	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Japn	Japanese*	0.23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Khmr	Khmer*	0.73	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Knda	Kannada*	1	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
Lao	Lao*	0.7	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?



# sealreq network

w3c.github.io/sealreq/

GitHub navigation bar with search and menu items: Pull requests, Issues, Marketplace, Explore.

Repository header for w3c/sealreq showing 17 Unwatch, 8 Stars, and 0 Forks. Navigation tabs include Code, Issues (12), Pull requests (0), Projects (0), Wiki, Insights, and Settings.

## Southeast Asian layout task force

Add topics

Repository overview showing 32 commits, 1 branch, and 0 releases. A list of files is displayed, including gap-analysis, homepage, CONTRIBUTING.md, and README.md.

## Southeast Asian Layout task force

This group exists to allow a network of experts to share information about gaps and requirements for supporting Asian scripts on the Web and in eBooks.

The main problem we seem to face at the moment is that experts don't know how to tell the W3C what problems they have with support of their script on the Web, and the W3C doesn't know how to contact people who can help when they have a problem. This network of experts should help to significantly reduce that problem.

Some experts may go a step further, and contribute to a gap-analysis or requirements document for a given language.

Topics for discussion are suggested by the [gap-analysis template](#). This work supports the development of indicating hot-spots for language support.

### Documents

- Javanese Script Gap Analysis
- Khmer Script Gap Analysis
- Lao Script Gap Analysis

### Feedback

Please use the [GitHub issue list](#) to report issues for language support, for discussions, and to send feedback on documents.

### Following

GitHub navigation bar with search and menu items: Pull requests, Issues, Marketplace, Explore.

Repository header for w3c/sealreq showing 17 Unwatch, 8 Stars, and 0 Forks. Navigation tabs include Code, Issues (12), Pull requests (0), Projects (0), Wiki, Insights, and Settings.

Issue filters and controls: Filters (is:issue is:open), Labels, Milestones, and a New Issue button.

Issue	Author	Labels	Projects	Milestones	Assignee	Sort
12 Open / 2 Closed						
1 Is inter-character spacing used in Lao?	r12a	lao, question				3
2 How is inter-character spacing used in Thai?	r12a	question, thai				
3 Drop caps in Javanese	adtbayuperdana	javanese				8
4 Ruby-like annotations in Thai & Burmese	ohbendy	myanmar, thai				7
5 Goal for requirements for Indonesian scripts	NorbertLindenberg	balinese, javanese, sundanese				5
6 Is inter-character spacing used for Khmer?	r12a	khmer, question				9
7 What quotation marks are used for Javanese script text?	r12a	javanese, question				1
8 Linebreaking CSS controls	mhosken					9
9 Tracking W3C specification work	r12a					
10 How is Khmer line-breaking handled on the Web?	r12a	khmer, question				5
11 How is line-breaking handled on the Web for Lao?	r12a	lao, question				6
12 Do Javanese & Balinese lines break at syllable or word boundaries?	r12a	balinese, javanese, question				19

# sealreq docs

W3C Editor's Draft

## Khmer Gap Analysis

W3C Editor's Draft 12 June 2018

### This version:

<https://w3c.github.io/sealreq/gap-analysis/khmr-gap>

### Latest published version:

<https://www.w3.org/TR/khmr-gap/>

### Latest editor's draft:

<https://w3c.github.io/sealreq/gap-analysis/khmr-gap>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

[Richard Ishida \(W3C\)](#)

### Github:

[repository](#)

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Abstract

This document describes and prioritises gaps for the support of Khmer written with the Khmer script on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and eReaders. **This is a preliminary analysis.**

## Status of This Document

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.*

This document describes and prioritises gaps for the support of Khmer written with the Khmer script on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and eReaders. It is linked to from the [language matrix](#) that tracks Web support for many languages.

The editor's draft of this document is being developed by the [W3C Internationalization Interest Group](#). It is published by the [Working Group Note](#).

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). Only send comments by email if you are unable to raise issues on github (see links below). All comments are welcome.

W3C Editor's Draft

## Lao Gap Analysis

W3C Editor's Draft 12 June 2018

### This version:

<https://w3c.github.io/sealreq/gap-analysis/laoo-gap>

### Latest published version:

<https://www.w3.org/TR/laoo-gap/>

### Latest editor's draft:

<https://w3c.github.io/sealreq/gap-analysis/laoo-gap>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

[Richard Ishida \(W3C\)](#)

### Github:

[repository](#)

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Abstract

This document describes and prioritises gaps for the support of Lao written with the Lao script on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and eReaders. **This is a preliminary analysis.**

## Status of This Document

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.*

This document describes and prioritises gaps for the support of Lao written with the Lao script on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and eReaders. It is linked to from the [language matrix](#) that tracks Web support for many languages.

The editor's draft of this document is being developed by the [W3C Internationalization Interest Group](#). It is published by the [Working Group Note](#).

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). Only send comments by email if you are unable to raise issues on github (see links below). All comments are welcome.

W3C Editor's Draft

## Javanese Script Gap Analysis

W3C Editor's Draft 12 June 2018

### This version:

<http://w3c.github.io/sealreq/gap-analysis/java-gap.html>

### Latest published version:

<https://www.w3.org/TR/java-gap/>

### Latest editor's draft:

<http://w3c.github.io/sealreq/gap-analysis/java-gap.html>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

[Richard Ishida \(W3C\)](#)

### Github:

[repository](#)

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Abstract

This document describes and prioritises gaps for the support of Javanese written with the Javanese script on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and eReaders. **This is a preliminary analysis.**

## Status of This Document

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.*

This document describes and prioritises gaps for the support of Javanese written with the Javanese script on the Web and in eBooks. In particular, it is concerned with text layout. It checks that needed features are supported in W3C specifications, in particular HTML and CSS and those relating to digital publications. It also checks whether the features have been implemented in browsers and eReaders. It is linked to from the [language matrix](#) that tracks Web support for many languages.

This document is an individual contribution, and is not currently a work item in any group, however, you can contact the [Internationalization Working Group](#) for more information. We welcome contributions to this and/or other documents.

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). Only send comments by email if you are unable to raise issues on github (see links below). All comments are welcome.





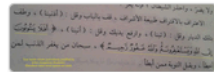
# type samples repo

[w3c.github.io/type-samples/](https://w3c.github.io/type-samples/)

## Type samples

This page lists pictures in the [type-samples github repository](#). There are 62 items.

### justification



**arab ar book justification**  
*justification* Two justification mechanisms.  
*Kashida* applied to a span of text inside a paragraph justified by inter-word spaces.



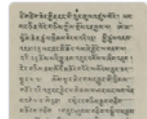
**hebr he book justification**  
*justification* Stretched characters in Hebrew.



**arab ar newsprint justification**  
*justification* Justification using simple baseline extensions.



**arab ar signage justification text-decoration**  
*text-decoration* Colour change between linked cursive characters.  
*justification* Arabic and latin text stretched to same size.



**tibt bo book justification**  
*justification* Line end padding with tsek marks.

### Select a feature

- All
- abbreviation (1)
- bidi-text (1)
- emphasis (3)
- font-style (1)
- fonts (4)
- glyphs-diacritics (4)
- hyphenation (2)
- initial-letter (5)
- justification (7)**
- lists (5)
- notes-footnotes (1)
- numbers (5)
- punctuation (4)
- quotations (2)
- ruby (3)
- text-decoration (6)
- vertical-text (8)

### Filter by script

- All
- arab (10)
- bopo (1)
- deva (2)
- ethi (3)
- grek (1)
- hang (2)
- hani (2)
- hebr (17)
- jpap (2)

### Filter by medium

- All
- book (23)

# text layout index

w3c.github.io/typography/

W3C Editor's Draft

## TABLE OF CONTENTS

1. Introduction
2. Characters & phrases
  - 2.1 Fonts
  - 2.2 Font styles
  - 2.3 Glyphs and diacritics
  - 2.4 Cursive text
  - 2.5 Transforming characters
  - 2.6 Numbers & digits
  - 2.7 Identifying boundaries of graphemes, words and larger groupings
  - 2.8 Punctuation
  - 2.9 Quotations
  - 2.10 Inline spacing
  - 2.11 Ruby annotation
  - 2.12 Text decoration
  - 2.13 Emphasis & highlights
  - 2.14 Bidirectional text direction
  - 2.15 Other inline features
3. Lines & paragraphs
  - 3.1 Line breaking
  - 3.2 Hyphenation
  - 3.3 Justification & line-end alignment
  - 3.4 Lists, counters, etc
  - 3.5 Initial letter styling
  - 3.6 Baselines & inline alignment
  - 3.7 Other paragraph features
4. Layout & pages
  - 4.1 Bidirectional layout
  - 4.2 Vertical text
  - 4.3 Notes, footnotes, etc
  - 4.4 Page numbering, running headers, etc
  - 4.5 More page layout and pagination
5. Changes Since the Last Published Version

## International text layout and typography index

W3C Editor's Draft 03 August 2018



### This version:

<https://w3c.github.io/typography/>

### Latest published version:

<https://www.w3.org/TR/typography/>

### Latest editor's draft:

<https://w3c.github.io/typography/>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

Richard Ishida (W3C)

### GitHub:

[repository](#)

Copyright © 2017-2018 W3C® (MIT, ERCIM, Keio, Beihang), W3C liability, trademark and document disclaimer. All rights reserved.

## Abstract

This document points browser implementers and specification developers to support typographic features of scripts or writing systems from around the world. It provides information in specifications, to tests, and to useful articles and papers that have been published from time to time.

## Status of This Document

This section describes the status of this document at the time of its publication. It is intended to provide a snapshot of the current state of the document. A list of current W3C publications and the status of their development can be found in the [W3C technical reports index](https://www.w3.org/publications/) at <https://www.w3.org/publications/>.

The information in this document helps to link users and developers to resources that address typographic needs around the world. It is expected that this document will be updated as new material becomes available or comes to our attention.

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please send comments by email if you are unable to raise issues on GitHub. Your comments are welcome.

To make it easier to track comments, please raise separate issues for each comment. Please point to the section you are commenting on using a URL for the section.

This document was published by the Internationalization Working Group as part of the W3C's work on Internationalization.

Comments regarding this document are welcome. Please send them to [ishida@w3.org](mailto:ishida@w3.org) (chives).

Publication as an Editor's Draft does not imply endorsement by the W3C. The document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

This document was produced by a group operating under the [W3C Patent Policy](#). W3C maintains a public list of [any patent disclosures](#) made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual

## 2.11 Ruby annotation

Ruby is used for phonetic and semantic annotations of East Asian text, including furigana, pinyin and zhuyin fuhao systems. In addition to positioning annotations along the correct side of the base text, there are many fine adjustments of the annotation and base text to support.

### Requirements

- *Chinese Layout Requirements: Interlinear annotations*
- (Chinese) *Bopomofo on the Web*
- (Chinese) *The Manual of the Phonetic Symbols of the Mandarin Script english chinese*
- *Japanese Layout Requirements: Ruby and Emphasis Dots* • *Positioning of Jukugo-ruby*
- (Japanese) *Use Cases & Exploratory Approaches for Ruby Markup*
- *Implementing Japanese Subtitles on Netflix: Rubies*

### GitHub resources

- [Requests for information](#)
- [Spec issues](#)
- [Browser bugs](#)
- [Type samples](#)

### Spec links

- [HTML5: The ruby element](#)
- [CSS3 Ruby](#)

### Tests

- [HTML5, the ruby element and its children](#)

# issue tracker

w3c.github.io/i18n-activity/textlayout/

W3C<sup>®</sup> Internationalization (i18n)  
Making the World Wide Web worldwidel

Learn Find Contact Join Follow

i18n site search:

## Text layout issue tracker

This page tracks issues related to international text layout on the Web. Issues listed may be requests to a local community for information about the behaviour of their writing system, or requests for improvements to either specs or browser implementations.

These issues are also linked to from the International text layout and typography index.

The mostly recently changed issues appear at the top of each section, and sections are ordered according to where the most recent changes occurred. The date indicates the last time there was a change to the tracker issue (not the issue in the other WG's repo). Each item links to a tracking issue in the i18n-activity github repo. Click on the link in that issue to follow the actual discussion.

You can filter the list in the URL. Add one or two of the filter names in the right-hand column after `?filter=`. If adding two, separate them with `+`.

There are 13 issues.

### lists

### ruby

465	Which side of the line do Mongolian ruby annotations appear normally?	■ ■	Jul 24, 2017
446	How to handle wide annotations in Mongolian ruby?	■ ■	Jul 20, 2017
445	Mongolian ruby requirements	■ ■	Jul 20, 2017
444	Which side of the line do Mongolian ruby annotations appear normally?	■ ■	Jul 20, 2017
437	Position of ruby, text-emphasis, under/overline, etc in vertical-LR writing mode	■ ■	Jul 20, 2017

### vertical-text

437	Position of ruby, text-emphasis, under/overline, etc in vertical-LR writing mode	■ ■	Jul 20, 2017
-----	--	-----	--------------

### text-decoration

464	Should traditional mongolian text in horizontal writing modes be horizontal or vertical?	■ ■	Jul 24, 2017
463	Text emphasis examples needed	■ ■	Jul 24, 2017
462	Underline and NNBS	■ ■	Jul 24, 2017
461	Mongolian text decoration styling	■ ■	Jul 24, 2017
440	Underline and NNBS	■ ■	Jul 20, 2017
437	Position of ruby, text-emphasis, under/overline, etc in vertical-LR writing mode	■ ■	Jul 20, 2017

### emphasis

437	Position of ruby, text-emphasis, under/overline, etc in vertical-LR writing mode	■ ■	Jul 20, 2017
-----	--	-----	--------------

Make list:

This repository **is:issue is:open label:text-de** Pull requests Issues Marketplace Explore

w3c / i18n-activity  19

Filters

12 Open  0 Closed Author Labels Projects Milestones

- Should traditional mongolian text in horizontal writing modes be horizontal or vertical?**    #464 opened on 24 Jul by r12a
- Text emphasis examples needed**    #463 opened on 24 Jul by r12a
- Underline and NNBS**    #462 opened on 24 Jul by r12a
- Mongolian text decoration styling**    #461 opened on 24 Jul by r12a
- How are underlines positioned in Arabic text?**    #457 opened on 20 Jul by r12a
- How are underlines positioned in Indic text?**     #456 opened on 20 Jul by r12a
- How are underlines positioned in Korean text?**    #455 opened on 20 Jul by r12a
- How are underlines positioned in Ethiopic text?**    #454 opened on 20 Jul by r12a
- Underline and NNBS**

### Useful links

Tracker issues on GitHub  
International text layout and typography index  
Review tracker

### Filter results

Click to show only items of this type:

- type-info-request
- spec-type-issue
- browser-type-bug
- Clear filters

Click to show only items of this script:

- arabic
- devanagari
- ethiopic
- han
- hangul
- hebrew
- japanese
- latin
- mongolian
- seasia
- tamil
- thai
- tibetan
- Clear filters

# notifications

## w3c/sealreq (+0/-0/3)

2 issues received 3 new comments:

- #17 [Is inter-character spacing used in Lao?](#) (2 by laonux, jclark) lao question
- #2 [Do Javanese & Balinese lines break at syllable or word boundaries?](#) (1 by adtbayuperdana) balinese javanese question

## w3c/csswg-drafts (+1/-0/9)

1 issues created:

- #2975 [\[css-text-4\] hyphenate-character doesn't just put hyphen at end of line](#) (by r12a) i18n-sealreq

3 issues received 9 new comments:

- #2975 [\[css-text-4\] hyphenate-character doesn't just put hyphen at end of line](#) (7 by r12a, Crissov, kojiishi, litherum) i18n-tracking
- #2976 [\[css-text-4\] Dealing with unusual line-break/hyphenation rules](#) (1 by r12a) i18n-sealreq i18n-tlreq i18n-tracking
- #2809 [\[css-text-4\] hyphenate-character doesn't accept just a character](#) (1 by r12a) css-text-4 i18n-tracking

# Language Enablement



Language matrix

Expert networks

Gap-analysis

Layout requirements

Text layout index

Tracker tools & notifications



Developer support

# spec reviews

[github.com/w3c/i18n-activity/projects/1](https://github.com/w3c/i18n-activity/projects/1)

The image shows a screenshot of a GitHub project board for 'spec reviews'. The board is organized into five columns representing different stages of the review process:

- 6 Early review opportunities**:
  - DCAT-Rev > FPWD (2018-05-08) #557 opened by aphillips (FPWD)
  - CSS Logical Properties and Values Level 1 (16 May 2017) #427 opened by r12a (FPWD)
  - Resource Timing 2016-04-26 #284 opened by r12a (FPWD)
  - CSS Object Model 2016-03-23 #285 opened by r12a (FPWD)
  - CSS Ruby #264 opened by r12a (WD)
  - Media Stream Track Content Hints > FPWD (2018-07-03) #576 opened by aphillips (FPWD, media-capture-and...)
- 2 Review requested**:
  - CSS Writing Modes Level 4 > Ongoing #520 opened by aphillips (css-writing-modes)
  - CSS Painting API Level 1 > 2018-10-09 #581 opened by aphillips (LC, css-paint-api)
- 2 In review**:
  - Verifiable Credentials Data Model 1.0 > 2018-09-20 #578 opened by aphillips (FPWD, needs-review)
  - CSS Generated Content 2016-06-20 #283 opened by r12a (FPWD)
- 20 Awaiting comment resolution**:
  - HTML 5.3 > 2018-05-25 #539 opened by aphillips (LC, html)
  - CSS Align > 2017-06-30 #422 opened by aphillips (LC)
  - TTML2 > 2017-09-30 #428 opened by aphillips
  - CSS Text Decoration > Ongoing #514 opened by aphillips (CR)
  - Accessibility (A11Y) Reqs for People with Low Vision > 2106-03-17 #286 opened by r12a (FPWD)
  - Web Authentication, > 2016-08-16 #268 opened by r12a (WD)
  - Browser Payment API > 2017-02-22 #347 opened by aphillips (LC)
  - Review request: changes to CR of DNT (tracking-dnt) #487 opened by r12a (CR)
  - UTR #53, Unicode Arabic Mark Ordering Algorithm #494 opened by r12a (FPWD)
  - Intersection Observer review #485 opened by LJWatson (FPWD)
- 26 Completed**:
  - ORDL Information Model (+1 more) > 2017-05-15 #397 opened by aphillips (LC)
  - Basic Card Payment, 2016-04-21 #269 opened by r12a (WD)
  - Gamepad API > 2018-06-27 #569 opened by LJWatson
  - WebVTT #289 opened by r12a (CR, LC)
  - Device and Sensors Reviews (6 docs) > 2017-12-31 #512 opened by aphillips
  - IMSC 1.0.1 > 2017-05-07 #376 opened by aphillips (CR)
  - DOM 4.1 > 2018-03-01 #527 opened by xfq
  - Semantic Sensor Network Ontology > Urgent #421 opened by aphillips (LC)
  - ARIA > 2017-04-30 #322 opened by aphillips (LC)
  - UI Events KeyboardEvent key Values, >2016-11-27 #267 opened by r12a (LC)
  - Screen Orientation #265 opened by r12a

# review comments

## Review comment tracker

This page tracks comments made by the i18n WG on the specs of other WGs. It only tracks issues the i18n WG has not closed. (Issues may remain open in the i18n track if closed by the other WG in their repo.)

The mostly recently changed issues appear at the top of each section, and sections are ordered according to where the most recent changes occurred. The date indicates the time there was a change to the *tracker* issue (not the issue in the other WG's repo). Each item links to a tracking issue in the [i18n-activity github repo](#). Click on the link in *that* to follow the actual discussion.

There are 255 issues.

### webauthn

589	truncation to 64-byte upper limit doesn't mention character boundaries	Sep 5, 201
207	Examples should include non-ASCII [editorial]	May 10, 20
587	Fix #593 - Refer to RFC 8266 for RP-controlled UI strings #878	May 10, 20
208	Display name content rules?	Sep 27, 20

### css-ruby

588	Add over-most-under-last value to ruby-position & text-emphasis-position for captioning	Sep 4, 201
555	[css-ruby] Ruby text and browser minimal font size #1917	Apr 30, 201
99	i18n-ISSUE-491: Change collapse to merge for ruby-merge	Apr 30, 201
98	i18n-ISSUE-359: Drop ruby-merge in favour of a specific jukugo value	Apr 30, 201
550	[css-ruby] Alignment of bopomofo #1907	Apr 30, 201
548	[css-ruby][css-text-decor] Underline position for Japanese text with ruby #1918	Apr 30, 201
549	[css-ruby] text-orientation of bopomofo annotation #1916	Apr 30, 201
252	[css-ruby-1] Generating Parentheses needs more thought	Jul 24, 201
253	[css-ruby-1] Proportional or fullwidth parens?	Jul 24, 201
255	Descriptions of space-between and space-around	Jul 24, 201
256	Treatment of multiple Latin words in space-between & space-around	Jul 24, 201
257	Multiple latin words alignment	Jul 24, 201
258	Default styling for chinese annotations	Jul 24, 201
259	Default ruby centring for more than just zh	Jul 24, 201
336	The default ruby-position for vertical-lr text may not be over	Jul 24, 201

## truncation to 64-byte upper limit doesn't mention character boundaries #973

Open aphillips opened this issue on 27 Jun · 20 comments



aphillips commented on 27 Jun

Member + 👤 ...

<https://w3c.github.io/webauthn/#dictionary-pkcredentialentity>

When referring to the `name` the spec says:

```
Authenticators MUST accept and store a 64-byte minimum length for a name member's value.
Authenticators MAY truncate a name member's value to a length equal to or greater than 64
bytes.
```

Note that the specification does not require truncation on a Unicode character boundary. Arbitrary truncation at a 64-byte limit on a multibyte encoding such as UTF-8 can corrupt the last character in the string. The spec should require that the truncation occur on a character boundary (is there a reason you didn't use character count instead of byte count in the first place?)

Assignees

No one—assign yourself

Labels

i18n-comment

type:technical

Projects

None yet

Milestone

L2-WD-00

Notifications

Subscribe

You're not receiving notifications from this thread.

8 participants



Lock conversation



aphillips commented on 27 Jun • edited

Member + 👤 ...

Hmm... also I think it doesn't mean to say "64-byte \*minimum\* length". I suspect it means to say "maximum" there. PS> Please add the i18n-comment label.

equalsJeffH added i18n-comment type:technical labels on 27 Jun



equalsJeffH commented on 28 Jun • edited

Member + 👤 ...

[this issue is related to issue #593 and PR #951]

@aphillips wrote:

```
Note that the specification does not require truncation on a Unicode character boundary
```

I was wondering whether/when you'd bring this up.

I've done some modest research on this topic of "unicode string truncation" (due to the text you cite above) and apparently it is more complex than simply performing truncation on a Unicode character boundary -- it apparently ought to properly be done on [extended grapheme cluster](#) (EGC) boundaries.

I found detailed analysis here: <https://hoitech.github.io/truncate-presentation/> ..and a library: <https://github.com/hoitech/Unicode-Truncate>, but nothing regarding "unicode string



# counting characters

ASCII	I	n		t	h	e		l	o	v	e	l	i	e	s	t		t	o	w	n		o	f		a	l	l	,		w	h	e	r	e		t	h
	49	6E	20	74	68	65	20	6C	6F	76	65	6C	69	65	73	74	20	74	6F	77	6E	20	6F	66	20	61	6C	6C	2C	20	77	68	65	72	65	20	74	68
Cyrillic	В			с		а		м		о		м			п		р		е		к		р		а		с		н		о		м			г	🔍	
	D0	92	20	D1	81	D0	B0	D0	BC	D0	BE	D0	BC	20	D0	BF	D1	80	D0	B5	D0	BA	D1	80	D0	B0	D1	81	D0	BD	D0	BE	D0	BC	20	D0	B3	D0
Han	在			最				美				丽			的			城			镇			,				那			里			的			房	🔍
	E5	9C	A8	E6	9C	80	E7	BE	8E	E4	B8	BD	E7	9A	84	E5	9F	8E	E9	95	87	EF	BC	8C	E9	82	A3	E9	87	8C	E7	9A	84	E6	88	BF	E5	B1
Emoji		🙈			😬				😬				😬			😬			😬				😬			🐱			😬				😬				🔍	
	F0	9F	99	8A	F0	9F	99	81	F0	9F	98	A2	F0	9F	98	A0	F0	9F	98	A7	F0	9F	98	8E	F0	9F	98	BD	F0	9F	98	89	F0	9F	98	84	F0	9F

Specifications that limit the length of a string *MUST* specify which type of unit (grapheme clusters, code points, code units) the length limit uses.

Specifications that limit the length of a string *SHOULD* specify the length in terms of Unicode code points (and not bytes!).

If a specification specifies a length limit, it *SHOULD* specify that any string that is truncated include an indicator that the string has been altered, such as ellipses.

# counting characters

ASCII	I	n	t	h	e	l	o	v	e	l	i	e	s	t	t	o	w	n	o	f	a	l	l	,	w	h	e	r	e	t	h							
	49	6E	20	74	68	65	20	6C	6F	76	65	6C	69	65	73	74	20	74	6F	77	6E	20	6F	66	20	61	6C	6C	2C	20	77	68	65	72	65	20	74	68
Cyrillic	В		с		а		м		о		м		п		р		е		к		р		а		с		н		о		м		г	👉				
	D0	92	20	D1	81	D0	B0	D0	BC	D0	BE	D0	BC	20	D0	BF	D1	80	D0	B5	D0	BA	D1	80	D0	B0	D1	81	D0	BD	D0	BE	D0	BC	20	D0	B3	D0
Han	在		最		美		丽		的		城		镇		,		那		里		的		房		👉													
	E5	9C	A8	E6	9C	80	E7	BE	8E	E4	B8	BD	E7	9A	84	E5	9F	8E	E9	95	87	EF	BC	8C	E9	82	A3	E9	87	8C	E7	9A	84	E6	88	BF	E5	B1
Emoji	👨		😬		😞		😬		😬		😬		😬		😎		🐱		😬		😬		👉															
	F0	9F	99	8A	F0	9F	99	81	F0	9F	98	A2	F0	9F	98	A0	F0	9F	98	A7	F0	9F	98	8E	F0	9F	98	BD	F0	9F	98	89	F0	9F	98	84	F0	9F



# counting characters

ASCII	I	n		t	h	e		l	o	v	e	l	i	e	s	t		t	o	w	n		o	f		a	l	l	,		w	h	e	r	e		t	h
	49	6E	20	74	68	65	20	6C	6F	76	65	6C	69	65	73	74	20	74	6F	77	6E	20	6F	66	20	61	6C	6C	2C	20	77	68	65	72	65	20	74	68
Cyrillic	В			с		а		м		о		м			п		р		е		к		р		а		с		н		о		м			г	👉	
	D0	92	20	D1	81	D0	B0	D0	BC	D0	BE	D0	BC	20	D0	BF	D1	80	D0	B5	D0	BA	D1	80	D0	B0	D1	81	D0	BD	D0	BE	D0	BC	20	D0	B3	D0
Han	在			最				美				丽			的			城			镇			,				那			里			的			房	👉
	E5	9C	A8	E6	9C	80	E7	BE	8E	E4	B8	BD	E7	9A	84	E5	9F	8E	E9	95	87	EF	BC	8C	E9	82	A3	E9	87	8C	E7	9A	84	E6	88	BF	E5	B1
Emoji		👨			😬				😬				😬			😬			😬			😬			🐱			😬			😬			😬			👉	
	F0	9F	99	8A	F0	9F	99	81	F0	9F	98	A2	F0	9F	98	A0	F0	9F	98	A7	F0	9F	98	8E	F0	9F	98	BD	F0	9F	98	89	F0	9F	98	84	F0	9F



1F939 JUGGLING



1F3FD EMOJI MODIFIER FITZPATRICK TYPE-4



200D ZERO WIDTH JOINER



2640 FEMALE SIGN



FE0F VARIATION SELECTOR-16



1F468 MAN



200D ZERO WIDTH JOINER



1F469 WOMAN



200D ZERO WIDTH JOINER



1F467 GIRL



200D ZERO WIDTH JOINER



1F467 GIRL

# counting characters

ASCII	I	n		t	h	e		l	o	v	e	l	i	e	s	t		t	o	w	n		o	f		a	l	l	,		w	h	e	r	e		t	h		
	49	6E	20	74	68	65	20	6C	6F	76	65	6C	69	65	73	74	20	74	6F	77	6E	20	6F	66	20	61	6C	6C	2C	20	77	68	65	72	65	20	74	68		
Cyrillic	В			с		а		м		о		м			п		р		е		к		р		а		с		н		о		м			г	👉			
	D0	92	20	D1	81	D0	B0	D0	BC	D0	BE	D0	BC	20	D0	BF	D1	80	D0	B5	D0	BA	D1	80	D0	B0	D1	81	D0	BD	D0	BE	D0	BC	20	D0	B3	D0		
Han	在			最				美				丽			的			城			镇																	房	👉	
	E5	9C	A8	E6	9C	80	E7	BE	8E	E4	B8	BD	E7	9A	84	E5	9F	8E	E9	95	87	EF	BC	8C	E9	82	A3	E9	87	8C	E7	9A	84	E6	88	BF	E5	B1		
	F0	9F	A4	B9	F0	9F	8F	BD	E2	80	8D	E2	99	80	EF	B8	8F	F0	9F	91	A8	E2	80	8D	F0	9F	91	A9	E2	80	8D	F0	9F	91	A7	E2	80	8D	F0	9F



# counting characters

ASCII	I	n		t	h	e		l	o	v	e	l	i	e	s	t		t	o	w	n		o	f		a	l	l	,		w	h	e	r	e		t	h		
	49	6E	20	74	68	65	20	6C	6F	76	65	6C	69	65	73	74	20	74	6F	77	6E	20	6F	66	20	61	6C	6C	2C	20	77	68	65	72	65	20	74	68		
Cyrillic	В			с		а		м		о		м			п		р		е		к		р		а		с		н		о		м			г	👉			
	D0	92	20	D1	81	D0	B0	D0	BC	D0	BE	D0	BC	20	D0	BF	D1	80	D0	B5	D0	BA	D1	80	D0	B0	D1	81	D0	BD	D0	BE	D0	BC	20	D0	B3	D0		
Han	在			最				美				丽			的			城			镇																	房	👉	
	E5	9C	A8	E6	9C	80	E7	BE	8E	E4	B8	BD	E7	9A	84	E5	9F	8E	E9	95	87	EF	BC	8C	E9	82	A3	E9	87	8C	E7	9A	84	E6	88	BF	E5	B1		
	F0	9F	A4	B9	F0	9F	8F	BD	E2	80	8D	E2	99	80	EF	B8	8F	F0	9F	91	A8	E2	80	8D	F0	9F	91	A9	E2	80	8D	F0	9F	91	A7	E2	80	8D	F0	9F



When specifying a length limitation, specifications *SHOULD* set the maximum length in a way that accommodates users whose basic units require multiple code point sequences.

Specifications *SHOULD NOT* limit the size of data fields unless there is a specific practical or technical limitation.

# 'specdev' guidelines

W3C Editor's Draft

## Internationalization Best Practices for Spec Developers

W3C Editor's Draft 07 August 2018



### This version:

<https://w3c.github.io/bp-i18n-specdev/>

### Latest published version:

<https://www.w3.org/TR/international-specs/>

### Latest editor's draft:

<https://w3c.github.io/bp-i18n-specdev/>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

Richard Ishida ([W3C](#))

### GitHub:

[repository](#)

Copyright © 2014-2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Status of This Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.

This document provides advice to specification developers about how to incorporate requirements for international use. What is currently available here is expected to be useful immediately, but is still an early draft and the document is in flux, and will grow over time as knowledge applied in reviews and discussions can be crystallized into guidelines.

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). Only send comments by email if you are unable to raise issues on github (see links below).

To make it easier to track comments, please raise separate issues or emails for each comment, and point to the section you are commenting on using a URL for the dated version of the document. All comments are welcome.

This document was published by the [Internationalization Working Group](#) as an Editor's Draft.

Comments regarding this document are welcome. Please send them to [www-international@w3.org](mailto:www-international@w3.org) (archives).

Publication as an Editor's Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this document as other than work in progress.

This document was produced by a group operating under the [W3C Patent Policy](#). W3C maintains a public list of any patent disclosures made in connection with the deliverables of the group; that page also includes instructions for disclosing a patent. An individual who has actual knowledge of a patent which the individual believes contains [Essential Claim\(s\)](#) must disclose the information in accordance with [section 6 of the W3C Patent Policy](#).



W3C Editor's Draft

## 4.13 Truncating or Limiting the Length of Strings

Some specifications, formats, or protocols or their implementations need to specify limits for the size of a given data structure or text field. This could be due to many reasons, such as limits on processing, memory, data structure size, and so forth. When selecting or specifying limits on the length of a given string, specifications or implementations need to ensure that they do not cause corruption in the text.

*#char\_trunc\_no\_limit*

Specifications **SHOULD NOT** limit the size of data fields unless there is a specific practical or technical limitation.

There are many reasons why a length limit might be needed in a specification or format. Generally length limits correspond to underlying limits in the implementation, such as the use of fixed-size fields in a database or data store, the desire to fit into practical boundaries such as packet size, or some other implementation detail related to storage allocation or efficiency.

When truncating strings, it's necessary to decide what units to use when counting the size of the string. In many cases this is beyond the control of the specification, since the truncation is occurring for some preordained reason, such as the size of a fixed-length field in a file format or database. However, when the choice is available, some general guidelines can be applied.

Below you can see the effect of truncating strings in different scripts on an arbitrary code unit boundary. In this case, each string is encoded in UTF-8 and truncated after the 38th byte. There are several things to notice here.

ASCII	I	n	t	h	e	l	o	v	e	l	i	e	s	t	t	o	w	n	o	f	a	l	l	,	w	h	e	r	e	t	h							
	49	6E	20	74	68	65	20	6C	6F	76	65	6C	69	65	73	74	20	74	6F	77	6E	20	6F	66	20	61	6C	6C	2C	20	77	68	65	72	65	20	74	68
Cyrillic	Б	с	а	м	о	м		п	е	к	р	а	с	х	о	м		р																				
	D0	92	20	D1	81	D0	B0	D0	BC	D0	BE	D0	BC	20	D0	BF	D1	80	D0	B5	D0	BA	D1	80	D0	B0	D1	81	D0	BD	D0	BE	D0	BC	20	D0	B3	D0
Han	在	最	美	丽	的	城	镇																															
	E5	9C	A8	E6	9C	80	E7	BE	8E	E4	B8	BD	E7	9A	84	E5	9F	8E	E9	95	87	EF	BC	8C	E9	82	A3	E9	87	8C	E7	9A	84	E6	88	BF	E3	B1
Emoji	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉	👉
	F0	9F	99	8A	F0	9F	99	81	F0	9F	98	A2	F0	9F	98	A0	F0	9F	98	A7	F0	9F	98	8E	F0	9F	98	BD	F0	9F	98	89	F0	9F	98	84	F0	9F

First, as the number of bytes-per-character goes up, the number of characters available inside the byte count limit goes down. ASCII has four times the available characters as emoji, three times as many as languages such as Chinese, and roughly twice as many as the Cyrillic example.

Second, in each of the non-ASCII example the byte boundary for truncation falls in the middle of a character. The resulting "dangling bytes" are rendered as U+FFFD and the byte sequence itself is not valid UTF-8. If the byte sequence were then serialized into a file format, such as JSON or CSV, the surrounding file might not be wholly valid or the file processor might generate errors because the byte sequence itself is invalid. Unlike many legacy character encodings, UTF-8 is highly patterned, so the longest broken character sequence that can result from mid-character truncation is one character. By contrast, in many legacy encodings, a file or document containing a mid-character truncated string can be wholly changed or rendered unintelligible after that point.

*#char\_trunc\_units*

Specifications that limit the length of a string **MUST** specify which type of unit (extended grapheme clusters, Unicode code points, or code units) the length limit uses.

*#char\_trunc\_unit\_rec*

Specifications that limit the length of a string **SHOULD** specify the length in terms of Unicode code points.

*#char\_trunc\_byte\_boundary*

If a specification sets a length limit in code units (such as bytes), it **MUST** specify that truncation can only occur on code point boundaries.

*#char\_trunc\_indicator*

If a specification specifies a length limit, it **SHOULD** specify that any string that is truncated include an indi-



# self-review checklist

[w3.org/international/techniques/developing-specs](http://w3.org/international/techniques/developing-specs)



118n site search:

[Learn](#) [Find](#) [Contact](#) [Join](#) [Follow](#)

## Internationalization techniques: Developing specifications

This page provides checklists for specification developers, editors and reviewers who want to take account of internationalization issues during the development of a spec. Where a checklist item is followed by a `more` link, click on that for more information. The page also lists links to useful resources on the W3C Internationalization Activity site and elsewhere that may help.

This page is generated from the document Internationalization Best Practices for Spec Developers. It is just one of several techniques indexes, each of which focus on a particular type of user.

[Collapse all](#) • [Expand all](#)

▶ Language	resource
▶ Text direction	text_direction
▶ Characters	characters
▶ Resource identifiers	seo_resid_non_ascii
▶ Markup & syntax	markup
▶ Typographic support	typography
▶ Local dates, times and formats	locale
▶ Navigation	navigation

You can link to this page and open specific items by using the `open` parameter in the URL. For example, `developing-specs.en?open=characters&open=char_choosing` will automatically open the sections `characters` and `choosing character encodings`. The necessary parameter values are shown to the right of each heading. These are links, to help you create a URL for sharing. The query `?open=all` expands all sections.

# string metadata



bdi يتم تحقيق ذلك بإضافة العنصر المضمن HTML5 في

```
<textarea dir = "auto" > </textarea >
```



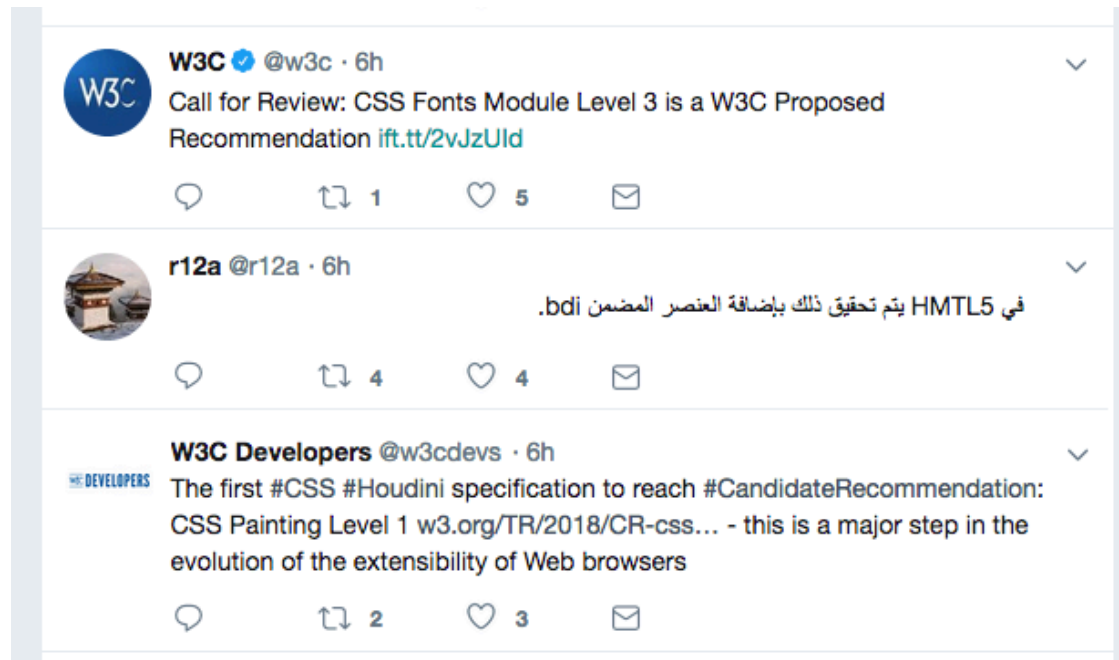
في HTML5 يتم تحقيق ذلك بإضافة العنصر المضمن bdi.



# string metadata



# string metadata



The image shows a screenshot of three tweets from a social media platform. The first tweet is from W3C (@w3c) and is in English. The second tweet is from r12a (@r12a) and is in Arabic. The third tweet is from W3C Developers (@w3cdevs) and is in English. Each tweet includes a profile picture, a name, a handle, a timestamp, and engagement icons for replies, retweets, likes, and direct messages.

**W3C** @w3c · 6h  
Call for Review: CSS Fonts Module Level 3 is a W3C Proposed Recommendation [ift.tt/2vJzUld](https://www.w3.org/2018/07/20-css-fonts-l3-cr/)

**r12a** @r12a · 6h  
في HTML5 يتم تحقيق ذلك بإضافة العنصر المضمن `bdi`.

**W3C Developers** @w3cdevs · 6h  
The first #CSS #Houdini specification to reach #CandidateRecommendation: CSS Painting Level 1 [w3.org/TR/2018/CR-css...](https://www.w3.org/TR/2018/CR-css-painting-1/) - this is a major step in the evolution of the extensibility of Web browsers

# string metadata

 #bidi في HTML5 يتم تحقيق ذلك بإضافة العنصر المضمن .bdi



 **W3C** @w3c · 6h  
Call for Review: CSS Fonts Module Level 3 is a W3C Proposed Recommendation [ift.tt/2vJzUld](http://ift.tt/2vJzUld)

  1  5 

 **r12a** @r12a · 6h  
bdi. يتم تحقيق ذلك بإضافة العنصر المضمن HTML5 في

  4  4 

 **W3C Developers** @w3cdevs · 6h  
The first #CSS #Houdini specification to reach #CandidateRecommendation: CSS Painting Level 1 [w3.org/TR/2018/CR-css...](http://w3.org/TR/2018/CR-css...) - this is a major step in the evolution of the extensibility of Web browsers

  2  3 

# string metadata

## Requirements for Chinese Text Layout 中文排版需求

W3C Editor's Draft 30 July 2018

### This version:

<https://w3c.github.io/clreq/>

### Latest published version:

<https://www.w3.org/TR/clreq/>

### Latest editor's draft:

<https://w3c.github.io/clreq/>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editors:

[董福興 \(Bobby TUNG\)](#) (特邀专家 (Invited Expert))

[陳奕鈞 \(Yijun CHEN\)](#) (特邀专家 (Invited Expert))

[刘庆 \(Eric Q. LIU\)](#) (特邀专家 (Invited Expert))

[陈慧晶 \(Hui Jing CHEN\)](#) (特邀专家 (Invited Expert))

[薛富侨 \(Fuqiao XUE\)](#) (W3C)

[Richard Ishida](#) (W3C)

### Former editors:

[李安琪 \(Angel LI\)](#) (W3C)

[梁海 \(Hai LIANG\)](#) (特邀专家 (Invited Expert))

[吴小倩 \(Xiaoqian WU\)](#) (W3C)

[张爱杰 \(Aijie ZHANG\)](#) (中国移动通信集团公司 (China Mobile))

```
{  
  name: "梁海 (Hai LIANG)",  
  company: "特邀专家 (Invited Expert)",  
  w3cid: 12345  
},
```



huijing commented 19 days ago

Member + 😊 ...

The CSS applies to elements with the `lang="zh-hans"` and `lang="zh-hant"` attributes respectively. However, I noticed that the generated elements at the top of the document, version and editor names etc., do not have the language attribute on them, is there any way we can add them, otherwise it seems that some Windows machines default to Yu Gothic (which is a Japanese font not a Chinese font).

梁海 梁海

# in development

W3C Editor's Draft

## Requirements for Language and Direction Metadata in Data Formats



W3C Editor's Draft 25 July 2018

### This version:

<https://w3c.github.io/string-meta/>

### Latest published version:

<https://www.w3.org/TR/string-meta/>

### Latest editor's draft:

<https://w3c.github.io/string-meta/>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editors:

[Addison Phillips](#) (Invited Expert)

[Richard Ishida](#) (W3C)

### Github:

[repository](#)

Copyright © 2017-2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Abstract

This document describes the best practices for identifying language and base direction in data formats used on the Web.

## Status of This Document

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.*

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). Only send comments by email if you are unable to raise issues on github (see links below). All comments are welcome.

To make it easier to track comments, please raise separate issues or emails for each comment, and point to the section you are commenting on using a URL for the dated version of the document.

This document was published by the [Internationalization Working Group](#) as an Editor's Draft.

Comments regarding this document are welcome. Please send them to [www-international@w3.org](mailto:www-international@w3.org) ([archives](#)).

Publication as an Editor's Draft does not imply endorsement by the W3C Membership. This is a draft document and may be updated, replaced or obsoleted by other documents at any time. It is inappropriate to cite this doc-



W3C Editor's Draft

## Character Model for the World Wide Web: String Matching



W3C Editor's Draft 27 July 2018

### This version:

<https://w3c.github.io/charmod-norm/>

### Latest published version:

<https://www.w3.org/TR/charmod-norm/>

### Latest editor's draft:

<https://w3c.github.io/charmod-norm/>

### Bug tracker:

[File a bug \(open bugs\)](#)

### Editor:

[Addison Phillips](#) (Invited Expert)

### Github:

[repository](#)

Copyright © 2004-2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

## Abstract

This document builds upon on *Character Model for the World Wide Web 1.0: Fundamentals* [CHARMOD] to provide authors of specifications, software developers, and content developers a common reference on string identify matching on the World Wide Web and thereby increase interoperability.

## Status of This Document

*This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the [W3C technical reports index](https://www.w3.org/TR/) at <https://www.w3.org/TR/>.*

### NOTE

This version of the document represents a significant change from the [earlier editions](#). Much of the content is changed and the recommendations are significantly altered. This fact is reflected in a change to the name of the document from "Character Model: Normalization".

### NOTE

#### Sending comments on this document

If you wish to make comments regarding this document, please raise them as [github issues](#). When reviewing the document, please refer to the latest [editor's copy](#). Only send comments by email if you are unable to raise issues on github (see links below). All comments are welcome.

To make it easier to track comments, please raise separate issues or emails for each comment, and point to the section you are commenting on using a URL.

This document was published by the [Internationalization Working Group](#) as an Editor's Draft.



# Developer support



String metadata (bidi & language)

Review radar

Self-review checklist

Guidelines for developers



Author support

# articles

## Articles, best practices & tutorials

You can also find resources using the [Technique index](#) and [Topic index](#), which provide more fine-grained access to information.

### Getting Started

- [Getting Started with the W3C I18n site](#)
- [Introducing Character Sets and Encodings](#)
- [Language on the Web](#)
- [Internationalization Quick Tips for the Web](#)

### Characters

- [Handling character encodings in HTML and CSS \(tutorial\)](#)
- [Character encodings for beginners](#)
- [Character encodings: Essential concepts](#)
- [Choosing & applying a character encoding](#)
- [Declaring character encodings in HTML](#)
- [Declaring character encodings in CSS](#)
- [The byte-order mark \(BOM\) in HTML](#)
- [Normalization in HTML and CSS](#)
- [Characters or markup?](#)
- [Changing an HTML page to Unicode](#)
- [Using character escapes in markup and CSS](#)
- [Document character set](#)
- [Setting the HTTP charset parameter](#)
- [Setting charset information in .htaccess](#)
- [Checking HTTP Headers](#)
- [Checking the character encoding using the validator](#)
- [HTML, XHTML, XML and Control Codes](#)
- [Missing characters and glyphs](#)
- [Who uses Unicode?](#)
- [Migrating to Unicode](#)

### Language

- [Working with language in HTML \(tutorial\)](#)

> [عربي](#) [Български](#) [Dansk](#)  
[Deutsch](#) [Ελληνικά](#)  
[Español](#) [Français](#) [हिन्दी](#)  
[Magyar](#) [Italiano](#) [日本語](#)  
[한국어](#) [Nederlands](#) [Polski](#)  
[Português](#) [Português-BR](#)  
[Română](#) [Русский](#)  
[Svenska](#) [ไทย](#) [Türkçe](#)  
[Українська](#) [简体汉语](#)  
[繁體中文](#)

### On this page

- [Getting Started](#)
- [Characters](#)
- [Language](#)
- [Markup & text](#)
- [Text direction](#)
- [Styling & layout](#)
- [Forms](#)
- [Navigation](#)
- [Web addresses](#)
- [Cultural issues](#)
- [Other](#)



# vertical text guidelines

## Styling vertical Chinese, Japanese, Korean and Mongolian text

**intended audience:** CSS developers, and anyone who needs guidance on how to produce vertically-oriented text for Chinese, Japanese and Korean using CSS.

Updated 2017-03-13 13:53

This article explains how to use CSS to produce vertical text for languages such as Chinese, Japanese, Korean, and Mongolian. The CSS specification contains a lot of implementation-specific information. This article draws out the basic information that content authors need to create the more common features of vertical text. It also compares the theory and practice about what is possible.

Each section explains how you would mark up your content according to the CSS spec (which is usually simple and straightforward), but then looks at what you currently have to do to achieve the same result in browsers that don't implement the standard property and value names. For more information, see [Browser support](#).

### Basic setup

Where it is supported, most of what you need should be achievable by applying the `writing-mode` property to the content that you want to be set vertically.

In Japanese, Chinese and Korean, lines start at the right side of the figure box and progress to the left. Latin script text typically runs down the page, with the letters rotated clockwise, while the Han characters remain upright. Any graphic also remains upright.

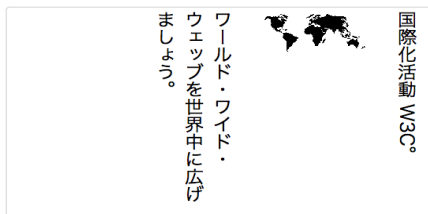


Fig. 1. Chinese and Japanese vertical text lines run right to left.

W3C Internationalization  
Home > Resources > Articles

On this page

- Basic setup
- Other values of `writing-mode`
- Changing the glyph orientation

```
CSS
figure {
  writing-mode: vertical-rl;
}
.upright {
  text-orientation: upright;
}

HTML
<figure>
  <p>「<span class="upright">i</span>」は、浅葱の双子の兄であり、共犯者だ。</p>
</figure>
```

```
<p>「<span class="upright">i</span>」は、浅葱の双子の兄であり、共犯者だ。</p>
```

To make the letter 'i' upright, you would then use this CSS declaration:

```
.upright {
  text-orientation: upright;
}
```

Test in your browser: [standard syntax](#) • [proprietary syntax](#)

doesn't convert any characters to fullwidth. If you need to do so, you could use the `text-orientation` property, if the browser supports it (see the next subsection).

transforms. If you transform the text into fullwidth characters, that may actually work on its own, since fullwidth characters are displayed upright by default.



Fig. 7. Using a fullwidth transform to make Latin letters stand upright.

'国际化活动 W3C' in the above example initially ran down the page, but applying the fullwidth transform using the following CSS makes them stand upright.

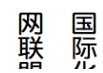
```
{
  text-transform: full-width;
}
```

Test in your browser: [standard syntax](#) • [proprietary syntax](#)

appropriate for initialisms, but is not necessarily useful for all types of upright text, and only that *this technique only works for Latin characters without accents!*

**width characters.** Another way to achieve this is to just use fullwidth characters, such as '国际化活动 W3C'. These will automatically be displayed upright by default. You don't need any special CSS in this case.

This approach is also only works for Latin script text that doesn't include accents (since those characters for which full-width variants exist).



For Webkit-based browsers you need to use the proprietary `-webkit-text-orientation` property. However, Webkit seems to have a problem centering the letter in the vertical line.

`text-transform: full-width` currently only works in Gecko. Text in other scripts will not have full-width equivalents, and so will need to use the next approach.



# ruby markup & styling guidelines

## Mono vs. group vs. jukugo

The most common approach when creating ruby is to associate each base character with a single ruby annotation, ie. [mono ruby](#). (All of the earlier examples illustrate mono ruby.)

Mono ruby makes it easy to handle line breaks when justifying text, since the browser can split the line between any two base characters. It also maps base characters and annotations precisely, and allows styling to apply the fine rendering control you may need.

[Group ruby](#), on the other hand, assigns a single annotation to a sequence of base characters, and these base characters can no longer be split at the end of a line. Situations where group ruby is appropriate include sequences of base characters that are associated with a single phonetic sound, or semantic ruby that applies to a whole word, or even a phrase.

Here is an example that shows group ruby on the left, and mono-ruby on the right. The two characters on the left are pronounced 'kyo', which is an indivisible sound. Note the difference in how the annotations are distributed in relation to the base characters.

きょう かいぎ  
今日の会議

To mark up group ruby you simply put more than one base character in the `rb` tag, as shown in the following code sample.

```
<ruby> <rb>今日</rb> <rb>きょう</rb> </ruby> <ruby> <rb>会</rb> <rb>かい</rb> <rb>議</rb> <rb>ぎ</rb> </ruby>。
```

[See live demo.](#)

If you want to apply [jukugo](#) rules to your ruby text, you should mark up the content in the same way as mono ruby, using the tabular model<sup>1</sup>, and use one `ruby` element per compound noun.

You don't need to worry about the overlaps in the markup. That will be taken care of by CSS. As previously mentioned, the markup simply establishes the correspondances between base characters and annotations.

## Bopomofo

Bopomofo, or zhuyin fuhao, characters used in ruby with Traditional Chinese characters are marked up in exactly the same way as mono ruby. No special markup is needed.

第十屆  
第十屆

\* The CSS Ruby specification doesn't say it yet, but the current intention is to apply jukugo styling only to tabular markup. Unfortunately, as described above, tabular markup is not well supported at the moment, but this is also academic because jukugo ruby styling is still not well defined, much less implemented, at the moment.

i The red coloring of the ruby text here is just to better show the position of the annotations in this example. There would not normally be a color difference. Note, in particular, how the tone marks appear to the right of the other bopomofo characters, even though they are not combining characters.

## Aligning annotations and bases

When the ruby text annotation is longer than the ruby base it belongs to, or vice-versa, there can be several different ways of dealing with the extra space that is lying around. The CSS Ruby spec deals with this mostly through the use of the `ruby-align` property.

It's best to use this property on the `ruby` element.

If you are working with inter-character bopomofo ruby, none of this is relevant, since the positioning of the bopomofo characters and tone marks is fixed.

## Aligning to one edge

うきよえ むかしばなし  
浮世絵 昔話

If you want to align the edges of the annotation and the base, you can use the `start` value of `ruby-align`. (An older version of the Ruby CSS spec also included an `end` value, but that has been removed from the current version.)

```
ruby { ruby-align: start; }
```

[See live demo.](#)

i The examples in this section are slightly unusual in that they use group ruby and distort phrym annotations in ways you might not normally, but which clearly illustrate the point being made. We also add red lines to help clarify positioning.

! If there is a `lang` attribute indicating that the content is Chinese (zh-...), then shorter text is centered. Otherwise, it acts as expected.   
! only works if you use the value `left`.  
See test results for major browsers.

## Centering shorter items of text

うきよえ むかしばなし  
浮世絵 昔話

If you want the shorter of the ruby base or ruby text to be centered, with the characters set solid, rather than aligned with an edge, it will hardly be a surprise to learn that you just need to set the value of `ruby-align` to `center`.

```
ruby { ruby-align: center; }
```

[See live demo.](#)

! See test results for major browsers.

# i18n test suite

[w3.org/international/tests](http://w3.org/international/tests)

## Internationalization tests

This page groups together pages being developed by the W3C Internationalization Working Group to assess internationalization support of user agents. It is an ongoing effort.

Note that these tests do not only test conformance with W3C standards. In some cases the tests also allow exploration of the behavior of user agents in ways not described by the standards.

### CSS Counter Styles 3

#### Custom styles

- 3.1. Counter algorithms: the system descriptor
  - Basic tests
- 3.3. Symbols before the marker: the prefix descriptor
  - Prefixes and suffixes
- 3.4. Symbols after the marker: the suffix descriptor
  - Prefixes and suffixes

#### Simple predefined counter styles

- 6.1. Numeric
  - Simple numeric
- 6.2. Alphabetic
  - Simple alphabetic

#### Complex predefined counter styles

- 7.1. Longhand East Asian
  - Limited range Asian
- 7.2. Ethiopic numeric
  - Ethiopic

### CSS Ruby 1

#### Ruby Box Model

- 2.4 Autohiding Base-identical Annotations
  - Ruby autohide

#### Ruby Formatting Properties

- 4.1 Ruby Positioning: the ruby-position property
  - ruby-position
- 4.2 Sharing Annotation Space: the ruby-merge property
  - ruby-merge
- 4.3 Ruby Text Distribution: the ruby-align property
  - ruby-align

### Custom Counter Styles

proprietary/writing-mode-vrl-forms-003-prop.html

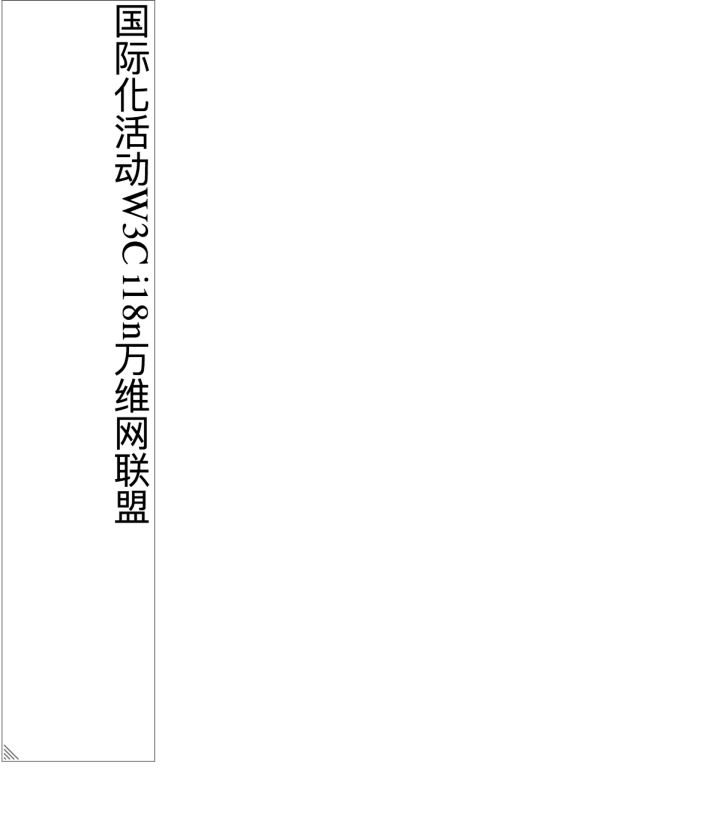
writing mode:vertical-rl, textarea

Pass Fail Partial Skip Firefox

Previous • Next

Link to spec  
i18n tests

Test passes if the input field is vertical and the text inside it has default vertical settings.



writing-mode:vertical-rl will display a textarea field vertically, and with vertical text inside.

Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:61.0) Gecko/20100101 Firefox/61.0

Score How to score and send results

Show results

# i18n test suite

w3.org/international/tests



i18n site search:

Home > Resources > Tests

## On this page

vertical-rl

- Basics
- Glyphs
- Text wrap
- Images
- Forms
- Tables

vertical-lr

- Basics
- Glyphs
- Text wrap
- Images
- Forms
- Tables

Links

## About this article

This article was published without public review. If you have comments, please send them using the link near the bottom of this page.

## Summarized test results: CSS3 Writing Modes, vertical text

**intended audience:** users, HTML coders, script developers, CSS coders, Web project managers, and anyone who wants to know whether browsers support the CSS Ruby spec.

Updated 2016-12-02 12:07

These tests check whether user agents correctly apply the `writing-mode` property per the CSS3 spec for the `vertical-lr` and `vertical-rl` values. They are just essential tests. More detailed tests for edge cases and finer aspects of rendering can be found in the CSS test suite.

To see the test, click on the link in the left-most column. To see detailed results for a single test, click on a row and look just above the table. The detailed results show the date(s) the test result was recorded, and the version of the browser tested.

Any dependencies are shown in notes above the table, and notes below the table will usually provide any additional useful information, including an explanation of why a result was marked as 'partially successful'.

### Key:

pass	fail	partially successful
------	------	----------------------

The proprietary test results are for either prefixed implementations, using `-webkit` or `-ms`, or for the nightly version of Firefox, or for non-standard writing-mode values in Internet Explorer.

If `writing-mode-vrl-001` or `writing-mode-vlr-001` fails, or either of the corresponding `-prop` tests, the remaining tests for the section can be ignored.

## vertical-rl

### Basics

Test link	Assertion	Firefox	Chrome	Opera	Safari	Edge	IE	Android	UC
<a href="#">writing mode:vertical-rl, vertical lines</a> <small>writing-mode-vrl-001.html</small>	writing-mode:vertical-rl will display a line of text vertically.	pass	pass	pass	fail	pass	fail	fail	fail

## vertical-rl

### Basics

Test link	Assertion	Firefox	Chrome	Opera	Safari	Edge	IE	Android	UC
<a href="#">writing mode:vertical-rl, vertical lines</a> <small>writing-mode-vrl-001.html</small>	writing-mode:vertical-rl will display a line of text vertically.	pass	pass	pass	fail	pass	fail	fail	fail
<a href="#">writing mode:vertical-rl, line wrap</a> <small>writing-mode-vrl-002.html</small>	writing-mode:vertical-rl will wrap lines from right to left.	pass	pass	pass	fail	pass	fail	fail	fail
<a href="#">writing mode:vertical-rl, alignment</a> <small>writing-mode-vrl-003.html</small>	writing-mode:vertical-rl will cause lines to display from the right side of the enclosing box.	pass	pass	pass	fail	pass	fail	fail	fail

Links: Section 3.1 • Related tests

### Proprietary syntax

Test link	Assertion	Firefox	Chrome	Opera	Safari	Edge	IE	Android	UC
<a href="#">writing mode:vertical-rl, vertical lines</a> <small>writing-mode-vrl-001-prop.html</small>	writing-mode:vertical-rl will display a line of text vertically.	pass	pass	pass	pass	pass	pass	pass	pass
<a href="#">writing mode:vertical-rl, line wrap</a> <small>writing-mode-vrl-002-prop.html</small>	writing-mode:vertical-rl will wrap lines from right to left.	pass	pass	pass	pass	pass	pass	pass	pass
<a href="#">writing mode:vertical-rl, alignment</a> <small>writing-mode-vrl-003-prop.html</small>	writing-mode:vertical-rl will cause lines to display from the right side of the enclosing box.	pass	pass	pass	pass	pass	pass	pass	pass

Links: Section 3.1 • Related tests

## Glyphs

Test link	Assertion	Firefox	Chrome	Opera	Safari	Edge	IE	Android	UC
<a href="#">writing mode:vertical-rl, default Han orientation</a> <small>writing-mode-vrl-005.html</small>	By default, writing-mode:vertical-rl will display Chinese characters upright.	pass	pass	pass	fail	pass	fail	fail	fail
<a href="#">writing mode:vertical-rl, default Latin orientation</a> <small>writing-mode-vrl-007.html</small>	writing-mode:vertical-rl will display Latin characters rotated 90° right by default.	pass	pass	pass	fail	pass	fail	fail	fail
<a href="#">writing mode:vertical-rl, default Arabic orientation</a> <small>writing-mode-vrl-008.html</small>	By default, writing-mode:vertical-rl will display Arabic characters rotated 90° right.	pass	pass	pass	fail	pass	fail	fail	fail
<a href="#">writing mode:vertical-rl, default Arabic direction</a> <small>writing-mode-vrl-009.html</small>	By default, writing-mode:vertical-rl will display Arabic characters progressing up the page.	pass	pass	pass	fail	pass	fail	fail	fail
<a href="#">writing mode:vertical-rl, Arabic joining</a>	By default, writing-mode:vertical-rl will display Arabic characters using cursive joining.	pass	pass	pass	fail	pass	fail	fail	fail

# i18n checker

[validator.w3.org/i18n-checker](http://validator.w3.org/i18n-checker)

## ▼ Detailed report

### ▼ ❌ Conflicting character encoding declarations

#### Explanation

The following character encoding declarations are inconsistent:

- `<meta charset="iso-8859-1"/>`
- `<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>`

Browsers will apply precedence rules to determine the character encoding to use for the page, but this may not be the encoding you intended.

#### What to do

Change the character encoding declarations so that they match. Ensure that your document is actually saved in the encoding you choose.

#### Further reading

- [Character encodings explained](#)
- [Choosing a character encoding](#)
- [Changing the encoding of a document](#)

↑ TOP

### ▶ ❌ Multiple encoding declarations using the `meta` tag

### ▶ ❌ Content-Language `meta` element used

### ▶ ❌ A language attribute value was incorrectly formed

### ▶ ❌ A language subtag is invalid

### ▶ ❌ A `lang` attribute value did not match an `xml:lang` value when they appeared together on the same tag.

### ▶ ⚠ Non-UTF-8 character encoding declared

### ▶ ⚠ Non-preferred name used for legacy character encoding

### ▶ ⚠ Found Unicode code points for directional controls

### ▶ ⚠ Unpaired directional controls found

### ▶ ⓘ `b` tags found with no class attribute



The W3C validators are hosted on server technology donated by HP, and supported by community donations.

[Donate](#) and help us build better tools for a better web.



Flattr

[Home](#) [About](#) [GitHub](#) [Feedback](#) [Version 2.0.3](#)

[Internationalization \(I18n\) Activity](#)  
*Making the World Wide Web world wide!*

COPYRIGHT © 2013-2016 W3C® (MIT, ERCIM, KEIO, BEIHANG), ALL RIGHTS RESERVED.  
W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY.  
YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS.



↑ TOP

# Author support



[Internationalisation articles](#)

[Techniques index](#)

[Internationalisation test suite](#)

[Internationalization Checker](#)



# Summary

# What we talked about

- **Language enablement**

understand where the gaps are for users of the global Web

- **Developer support**

build standards & applications that support a global Web

- **Author support**

help people create content in their own language, or create content that will be localised

- language matrix
- layout requirements
- gap-analysis research & documentation
- text layout index & sample repo
- issue tracking & notification
  
- reviews & discussion of issues raised
- in-depth research on issues like string metadata, case-folding, etc.
- 'specdev' guidelines
- self-review checklist
  
- HTML & CSS task-oriented dos & don'ts
- articles for content developers
- tests & their results
- Internationalization Checker



# Key message

The W<sub>3</sub>C isn't a Genie in a lamp that solves all your problems. It brings people together & facilitates work, but this is **your** Web.

To produce change we need people **like you** to step up and provide guidance and work through issues.

**Get involved** and help us ensure that it meets local needs around the world.





Next steps...

# next steps

- **Join** a layout network as follower or contributor.
- **Contribute** to creating gap-analysis docs.
- **Learn** about strings and internationalization.
- **Use** the information in our articles and test results.
- **Check** your pages with the i18n checker.
- **Support** the sponsorship program.



Thank you

[w3.org/International/talks/iuc42](http://w3.org/International/talks/iuc42)