

Editing challenges on multi-script wikis

C. Scott Ananian
Wikimania 2017



August 12, 2017
Montréal, Québec, Canada

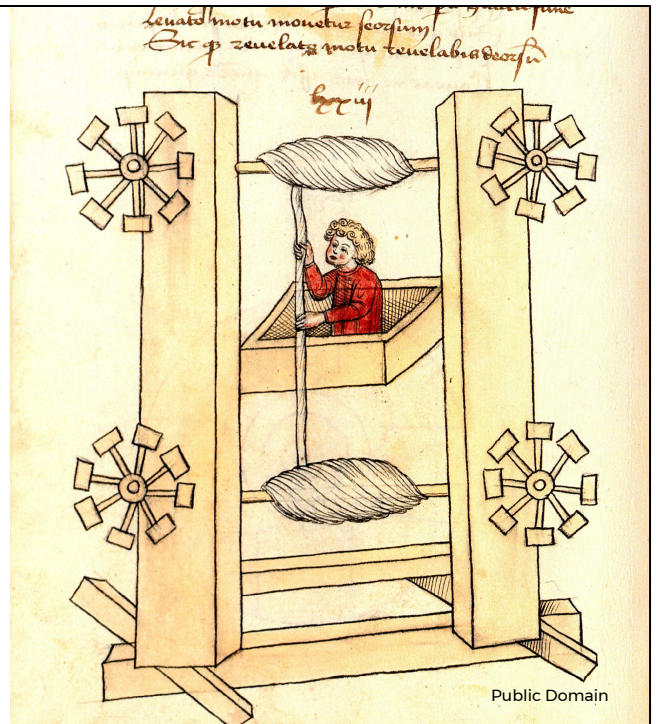
Hello. My name is C. Scott Ananian. I work for the Wikimedia Foundation, on the parsing team.

This talk is about wikis which use multiple writing systems to express their language, or in some cases the same writing system to express multiple closely-related languages. Let me show you what that look like.

English (en-US)

[[Elevator]]

Elevator design by the German engineer Konrad Kyeser (1405)

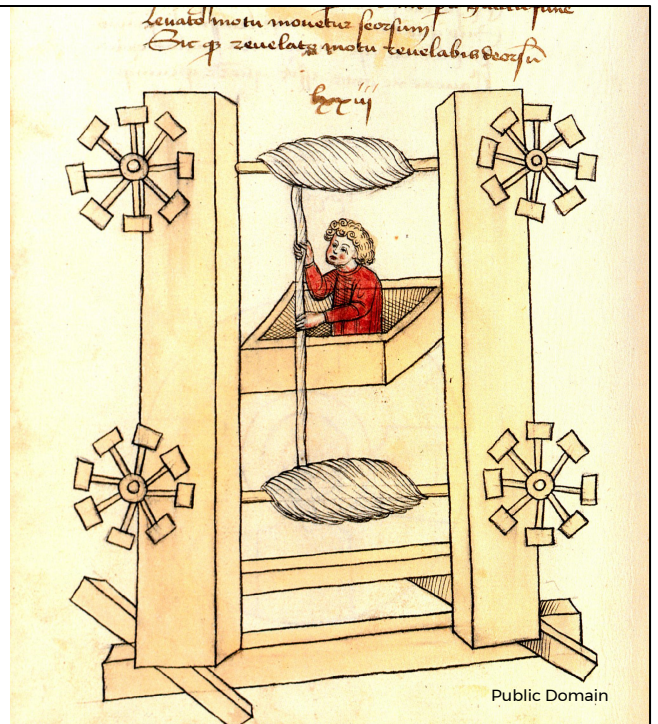


This is an illustration from the English Wikipedia article titled, "Elevator". For the benefit of those who don't speak American, an Elevator is a device for moving up and down from floor to floor. This is an early design by a German.

English (en-GB)

[[Lift]]

Lift design by the German engineer Konrad Kyeser (1405)



If you speak British English, of course, this isn't an "elevator" at all. This is a "lift". American and British English vary on a number of terms and pronunciations, which co-exist somewhat uneasily on English Wikipedia. And there are further variants found in Indian English and other places around the world. There is a long policy on Wikipedia about where and how to use regional terms. But, with a few exceptions, the different variants are mutually intelligible.

Serbian (sr-el)

[[Lift]]

Lift je uređaj za transport ljudi ili tereta među spratovima zgrada ili radnih platformi.



WIKIMEDIA
FOUNDATION



CC BY-SA 3.0 Peregrine981

As it turns out, Serbian sides with the Brits. I don't know how this is pronounced, but it is spelled the same as British English: lift. This is text from Serbian Wikipedia about elevators.

Serbian (sr-ec)

[[Лифт]]

Лифт је уређај за транспорт људи или терета међу спратовима зграда или радних платформи.



WIKIMEDIA
FOUNDATION



CC BY-SA 3.0 Peregrine981

But as it turns out, there are multiple ways to write Serbian. This is the same article text, written in Cyrillic script. LanguageConverter allows you to switch between these views of the article, depending on which script you prefer to read. As I understand it, many people in Serbia can read both. There are also spelling and vocabulary differences, orthogonal to the script choice.

Mandarin Chinese (zh-tw)

[[電梯]]

電梯，亦稱升降機、垂直
電梯。在馬來西亞、新加
坡和香港俗稱
「𨋖」(lift的譯音)，
是一種垂直運送行人或
貨物的運輸工具。



CC BY-SA 4.0 MNXANL

This is the same article on Chinese Wikipedia, as you would read it if you lived in Taiwan. It is written with what are called “Traditional” characters.

Mandarin Chinese (zh-cn)

[[电梯]]

电梯，亦称升降机、垂直电梯。在马来西亚、新加坡和香港俗称“𨋖”（lift的译音），是一种垂直运送行人或货物的运输工具。



CC BY-SA 4.0 MNXANL

But if you're on the mainland, this is how that same article appears. These are “simplified” characters, which were introduced in 1956 and 1964 by the mainland People's Republic of China government in an attempt to promote literacy. The Law of the People's Republic of China on the National Common Language and Characters implies simplified Chinese as the standard script, and relegates traditional Chinese to certain aspects and purposes such as ceremonies, cultural purposes (e.g. calligraphy), decoration, publications and books on ancient literature and poetry, and research purposes.

Unlike Serbian, in this case, few people are equally fluent in both traditional and simplified characters. There are thousands of characters which differ between the two orthographies, although note that not all of the characters change.

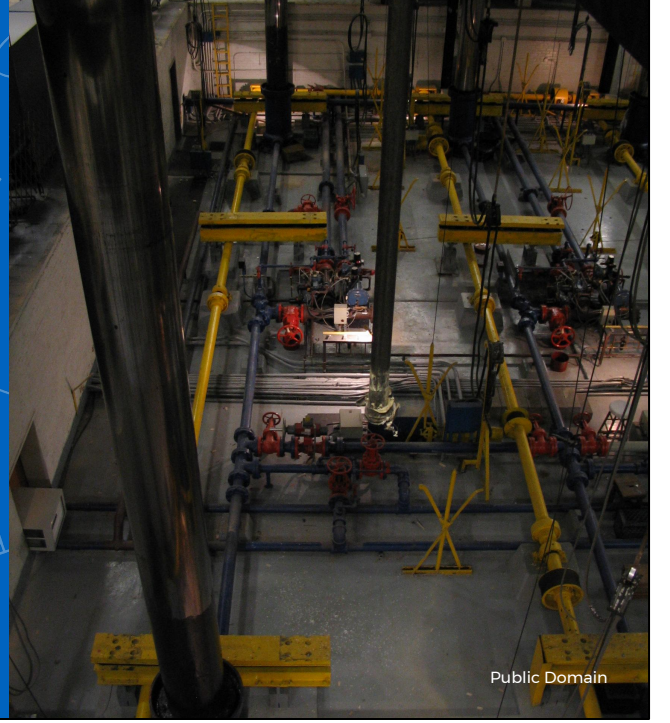
Hindi (hi)

[[उत्थापक]]

उत्थापक, उच्चालित्र अथवा एलिवेटर (lift या elevator) एक युक्ति है वस्तुओं एवं व्यक्तियों को उर्ध्व दिशा में चढ़ाने-उतारने के काम आती है। प्रायः किसी बहुमंजिला ऊँचे भवन, जलपोत एवं अन्य संरचनाओं में उत्थापक लगा होता है जो गोलों को या सामान आदि को एक मंजिल से दूसरी मंजिल या एक स्तर से दूसरे स्तर पर लाता और ले जाता है। उत्थापक प्रायः विद्युत मोटर द्वारा चलते हैं।



WIKIMEDIA
FOUNDATION



Public Domain

This is the Hindi article, written in Devanagari script.

Urdu (ur)

[[رافعہ]]

انتصابی نقل و حمل کی **کل** جدید عمارتوں، جہازوں اور کاتوں میں استعمال ہونے والی تمام کھلی اور بند ساختوں اور لگاتار چلنے والے ان پٹوں کو بھی کہا جاتا ہے جو (**Elevator**: رافعہ یا (انگریزی بھاری چیزوں کو ایک جگہ سے دوسری جگہ پہنچاتے ہیں طاقت سے چلنے والے رافعہ جو عام طور پر **بھاری** سے کام کرتے تھے **انیسویں صدی** **ہیسوی** سے استعمال ہو رہے تھے جبکہ اس **صدی** کے **اواخر میں برقی رافعہ** عام ہو گیا۔

WIKIMEDIA FOUNDATION

Public Domain

This is the article on elevators in Urdu. It is written in Arabic script, right to left. Urdu and Hindi are different dialects of the Hindustani language, written in very different scripts: Arabic on the Pakistan side of the border, Devanagiri on the India side. We don't currently use LanguageConverter on these wikis, but I received an inquiry about it from Urdu Wikipedia, which would like to grow their wiki.

Punjabi is a similar case, with two smaller wikis split by the India-Pakistan border. There are *four* scripts used to write Punjabi: "Shahmukhi and Gurmukhī scripts are the most commonly ones used for writing Punjabi and are considered the official scripts of the language." But also Urdu and Devanāgarī.

It brings our wikis together

(Although fighting over orthography is part of the fun?)



..to bring our wikis together. Across arbitrary political or historical divisions, to share the work of creating repositories of free knowledge. (Flip back: three things: words, scripts, and if you stretch, languages)

Conversion in use on 11 wikis; wanted on ~35 more

[\[\[meta:Wikipedias in multiple writing systems\]\]](#)

- Chinese
- Serbian
- Kazakh
- Kurdish
- Inuktitut
- Anglo-Saxon
- Shilha
- Tajik (partial)
- Uzbek (partial)
- Gan (partial)
- Cantonese (client-side)



It's currently in use on 11 wikis, and there is a large backlog of other wikis which would like to adopt it to bridge differences.



But. There's a significant drawback: while helping readers, it can hinder editors.

Mixed script editing

Списак 118 познатих хемијских елемената [уреди]

Следећа табела садржи 118 познатих хемијских елемената.

- **Атомски број, име, и симбол** служе независно као јединствени идентификатори.
- **Имена** су она која су прихваћена од стране **IUPAC**; провизиона имена за недавно произведене елементе који нису формално именовани су дата у заградама.
- **Група, периода, и блок** се односе на позицију елемента у **периодном систему**. Бројеви група су у тренутно званично прихваћеној нотацији; за старије алтернативне нотације погледајте **Група периодног система елемената**.
- **Станје материје** (*Čvrsto, tečno, или gasovito*) се односи на standardne uslove **temperature i pritiska (STP)**.
- **Појављивање** прави разлику између елемената који се јављају у природи, категорисане као било *Praiskonski* или *Prolazni* (у смислу raspada), и *Sintetički* елементи који су произведени технолошким путем, и нису природно познати.
- **Опис** сумира својства елемента користећи опшрне категорије које су присутне у периодном систему: **актинoid, алкални метал, земноалкални метал, halogen, лантанoid, метал, metaloid, племенити gas, nemetal, и prelazni metal.**

This is a section of an [article on Serbian wikipedia](#). You can see that the source text begins in Cyrillic script and switches to Latin script halfway through. Even though everything is made consistent when you use LanguageConverter to read the page, the editor has to be fluent in both scripts (or dialects, or languages) in order to edit or proofread the page. This is relatively reasonable for American and British variants (although it may scare off new editors who naively violate our spelling policies), is acceptable for Serbian, but is a major barrier to editors on Chinese Wikipedia.

Parsoid support

Visual Editor support



CC by SA 4.0, C. Scott Ananian

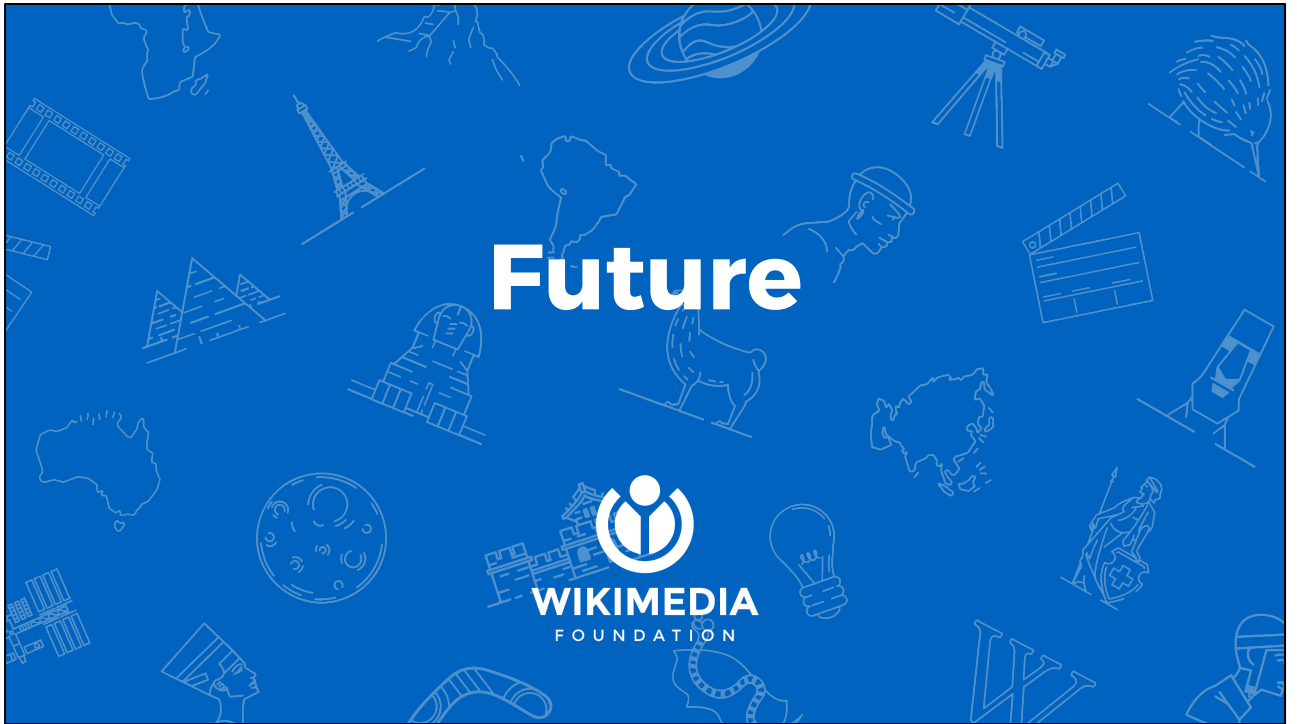
Two weeks ago basic support for LanguageConverter markup rolled out in Parsoid and Visual Editor. This ensures that it is safe to use VisualEditor on these wikis with LanguageConverter turned on, and there are patches awaiting review to allow basic editing support of LanguageConverter markup.

Reading variants

- Google indexing
- Kiwix offline reader
- Android app
- PDF
- REST API



This doesn't actually do conversion in Visual Editor, the text will still appear mixed (as it does in the wikitext editor). The next near-future step is to implement our existing converters on top of the Parsoid output, which will allow reading content in your preferred variant. This will be useful to all of the downstream clients which use Parsoid output: google indexing, the kiwix offline reader, the android mobile app, and PDF export. The converted text will be available via the REST API.



But we can do more. There are four specific “challenges” we’d like to address.

Source language annotation

Current implementation relies on character set to implicitly mark source language.

For reliable operation in some variants, we need explicit hints.

Ideally we can use an [annotation service](#), and not direct wikitext markup.

Initial work will focus on wikis where articles are stored in a consistent variant; for example, articles on yue.wiki stored in yue-hant variant.

Лифт => Lift

I think I have learned that the best way to lift one's self up is to help someone else.

— [Booker T. Washington](#)



For example, in Chinese wiki, not all of the characters change between simplified and traditional. In English you can't use character set at all. For example, in this quote from Booker T. Washington I can't determine whether "lift" should be converted to "elevator" for an American reader unless I know whether the source text is American or British.

But we can simplify this process on a few wikis, which already have a convention that the edited text is written in a single variant. For example, on Cantonese wiki they edit articles exclusively in traditional characters.

Single-variant editing

Parsoid/Visual Editor use **Selective Serialization** in order to convert only the edited portion of a page back to wikitext.

Same technology can be used to edit in your native variant but avoid disrupting the variant used in unedited portions when saving changes.

Please lift those boxes and place them in the elevator.

Please `-{lift}`- those boxes and place them in the lift.



The holy grail here is single-variant editing. [...] Note that we need to use our source language information to protect the first “lift” when a British English editor is working on this sentence, otherwise both instances of the word might be converted to “elevator” for an American reader.

Glossaries

Word-based conversion dictionaries can get large and accumulate topic-specific entries.

Chinese wikipedia uses [templates](#) and a [custom gadget](#) to add new conversion rules.

Improve support by bringing this into core as article [glossaries](#).

[[Glossary:Harry Potter]]

**-{en-uk:Sorcerer;
en-us:Philosopher}-
-{en-uk:Minister for Magic;
en-us:Minister of Magic}-
-{en-uk:sherbet lemon;
en-us:lemon drop}-**



Remember we said that LanguageConverter converts words as well as scripts. These dictionaries can get large. [...]

Glossaries might look much like Categories, so instead of adding a template you'd add something like **[[Glossary:Harry Potter]]** to an article about the Harry Potter series of books.

That glossary would define a set of rules specific to that series of books. "Sorcerer's stone" in the UK becomes "Philosopher's stone" in the US, etc.

Hoisting these glossaries out of templates could also significantly improve performance of rendering.

Content Translation

LanguageConverter can be viewed as a simple implementation of machine translation.

At the moment we only use machine translation to generate the initial revision of articles.

A future [Content Translation Tool](#) could be used to keep parallel texts in different languages (or variants) in sync.



LanguageConverter: works on words, scripts, and *in some cases* languages. Indeed you can think about LanguageConverter as a special case of machine translation, one where the automatic translation is surprisingly likely to be entirely accurate and correct. We have a tool for applying machine translation to articles on Wikipedia: the Content Translation tool. Currently it only applies to creating the initial revision of articles. I would like to eventually see LanguageConverter migrated into the ContentTranslation framework and Content Translation grow the ability to maintain edits in sync.

TL;DR



CC by SA 4.0, C. Scott Ananian

Ok. This nice dog hasn't been paying attention and wants me to sum up.

Summary

- Not everyone uses Latin letters!
- Some texts which look different are really quite similar.
- LanguageConverter is a tool for sharing content when texts are closely related.
- **We can make editing easier on wikis using LanguageConverter!**
- These wikis could be good incubators for more general cross-wiki machine translation tools.





THANK YOU

[\[\[commons:File:Editing_challenges_on_multi-script_wikis.pdf\]\]](#)



WIKIMEDIA
FOUNDATION

Example markup

Marking up text which has variants:

- Bidirectional rules:
-`{zh-hans:computer; zh-hant:ELECTRONICBRAIN;}`-
- Unidirectional rules:
-`{HUGEBLOCK=>zh-cn:macro;}`-
- Disable conversion:
-`{R|SimpTrad}`-

Also a bunch of stateful options for adding/removing rules (more on this later), and for working around title limitations.

See https://www.mediawiki.org/wiki/Writing_systems/Syntax

Glossary Templates

From [[zh:鋼鐵人3]] ([[en:Iron Man 3]]):

```
{{noteTA
```

```
|G1=Movie
```

```
|G2=Show
```

```
|G3=美国漫画
```

```
|1=zh-hans:罗伯特; zh-tw:勞勃; zh-hk:羅拔;
```

```
|2=zh-hans:奧德利奇·齊連安; zh-tw:奧德奇·齊禮安; zh-hk:奧  
德奇·齊禮安;
```

```
|3=zh-hans:羅德斯; zh-tw:羅德; zh-hk:羅德;
```

```
}}
```



Transliteration code

```
class SrConverter extends LanguageConverter {
    public $mToLatin = array(
        'a' => 'a', 'ǎ' => 'b', 'b' => 'v', 'r' => 'g', 'd' => 'd',
        'h' => 'd', 'e' => 'e', 'x' => 'ž', 'z' => 'z', 'u' => 'i',
        [...]
        'X' => 'H', 'Ц' => 'C', 'Ч' => 'Č', 'Џ' => 'Dž', 'Ш' => 'Š',
    );
    public $mToCyrillics = array(
        'a' => 'a', 'b' => 'б', 'c' => 'ц', 'č' => 'ч', 'ć' => 'h',
        'd' => 'д', 'dž' => 'џ', 'đ' => 'ђ', 'e' => 'e', 'f' => 'ф',
        [...]
        'Nj' => 'Њ', 'n!j' => 'њ', 'N!j' => 'Hj', 'N!J' => 'HJ'
    );
};
```



Word conversion code

```
$zh2Hant = array(  
'佩' => '佩',  
'佉' => '佉',  
[...]  
';克制' => ';剋制',  
'? 克制' => '? 剋制',  
);  
$zh2Hans = array(  
'佩' => '佩',  
'佉' => '佉',  
[...]
```

