



How to squeeze a language tag into a Locale

Eike Rathke
Red Hat, Inc.
FOSDEM 2014-02-01

Agenda

- About the Speaker
- Attributes in the OpenDocument Format
- BCP 47 language tags
- Locales in LibreOffice
 - Challenge
 - Solution
- Summary
- Q&A



About the Speaker

- Eike Rathke, known on the net as erAck
- Based in Hamburg, Germany
- Worked on StarOffice from 1993 to 2000 for Star Division
- Worked on OpenOffice.org from 2000 to 2011 for Sun Microsystems and one other company
- Works on LibreOffice since 2011, employed by Red Hat, Inc.
- Areas of expertise:
 - Calc core, formula compiler and interpreter
 - number formatter/scanner
 - i18n framework, locale data
- Also mentor and knowledge spreader whenever possible
- Web site <http://erack.de/>



Attributes in ODF 1.0/1.1

- ▼ fo:language ISO 639 alpha language code
 - ▼ example fo:language='de'

- ▼ fo:country ISO 3166 alpha country code
 - ▼ example fo:country='DE'

Attributes in ODF 1.2

- ▼ fo:language
 - ▼ example fo:language='de'
- ▼ fo:script ISO 15924 alpha script code
 - ▼ example fo:script='Latn'
- ▼ fo:country
 - ▼ example fo:country='DE'

Attributes in ODF 1.2

- ▼ fo:language
 - ▼ example fo:language='sr'
- ▼ fo:script
 - ▼ example fo:script='Cyril'
- ▼ fo:country
 - ▼ example fo:country='RS'

Attributes in ODF 1.2

- ▼ fo:language
 - ▼ example fo:language='sr'
- ▼ fo:script
 - ▼ example fo:script='Latn'
- ▼ fo:country
 - ▼ example fo:country='RS'

Attributes in ODF 1.2

- ▼ *:rfc-language-tag BCP 47 language tag
 - ▼ example `style:rfc-language-tag='sr-Latn-RS'`
 - ▼ takes precedence over `fo:language`, `fo:script` and `fo:country`
- ▼ `fo:language`
 - ▼ example `fo:language='sr'`
- ▼ `fo:script`
 - ▼ example `fo:script='Latn'`
- ▼ `fo:country`
 - ▼ example `fo:country='RS'`

BCP 47 Language Tags

- ▼ BCP 47, Best Current Praxis
 - ▼ Currently RFC 5646, Tags for the Identification of Languages (previously RFC 4646, 3066, 1766)
 - ▼ For pointers see <http://www.langtag.net/> and my selection at http://erack.de/bookmarks/D.html#Language_Tags
- ▼ language ["-" script] ["-" region] *("-" variant) *("-" extension) ["-" privateuse]
 - ▼ For example *ca-valencia* for Catalan Valencian
 - ▼ Where *ca* is language and *valencia* is variant
- ▼ Tags are registered with IANA (Internet Assigned Numbers Authority)
 - ▼ Reuse of codes/tags for different languages or countries will not happen, like it was the case with ISO 3166 CS, first Czechoslovakia then Serbia and Montenegro

BCP 47 Language Tags

- ▼ Defined variants are possible, for example specify that a document uses
 - ▼ “German in the traditional spelling/orthography before 1996”
 - ▼ *de-DE-1901*
 - ▼ “British English with Oxford English Dictionary spelling”
 - ▼ *en-GB-oed*
 - ▼ important because it is mandatory for UN documents

Locales in LibreOffice

- ▼ Until LibreOffice 4.1 known locales use only ISO 639 alpha language codes and ISO 3166 alpha country codes
 - ▼ For example, *en-US* or *de-DE* or *pjt-AU*
 - ▼ (Pitjantjatjara in Australia in case you didn't know)
- ▼ UNO API uses the `com::sun::star::lang::Locale` struct
 - ▼ Designed after the Java locale `java.lang.Locale`
 - ▼ 3 struct members / fields
 - ▼ Language
 - ▼ Restricted to ISO 639 alpha codes
 - ▼ Country
 - ▼ Restricted to ISO 3166 alpha codes
 - ▼ Variant
 - ▼ “free-form” field, platform and application specific

The Challenge

- ▼ UNO API uses the `com::sun::star::lang::Locale` struct
 - ▼ interfaces provide or accept a `css::lang::Locale` in a method
 - ▼ interfaces derive from `css::lang::XLocalizable`
 - ▼ services implement the `css::lang::XLocalizable` interface
 - ▼ ~300-400 methods and interfaces
- ▼ ALL affected interfaces would be incompatible if `css::lang::Locale` would be changed or replaced by something different (e.g. designed after the `java.util.Locale` that implements BCP 47)
 - ▼ 90% of existing extensions would cease to run
 - ▼ => No-Go!

The Solution

- ▼ LibreOffice 4.2 defines a convention to transport language tags in `com::sun::star::lang::Locale`
 - ▼ If a locale can be expressed as ISO language and country codes only, use those in Language and Country fields
 - ▼ maintains compatibility with currently used locales
 - ▼ Else, a locale contains a script code or needs to be expressed as language tag, set the Language field to the *q/t* ISO 639-3 reserved for private use code (mnemonic Q-Language-Tag) and set the Variant field to the full language tag
 - ▼ The Country field may contain the corresponding ISO 3166 alpha country code, if any, or otherwise must be empty

Summary

- ▼ LibreOffice extension developers when evaluating and processing a Locale's content should prepare for the *q/t* code in the Language field and if present handle language tags in the Variant field:

```
if Language == "q/t"  
    tag = Variant  
else if Country == ""  
    tag = Language  
else  
    tag = Language + "-" + Country
```

- ▼ LibreOffice internal class *LanguageTag* from `include/i18nlangtag/languagetag.hxx` provides all necessary conversions from/to language tag / Locale / MS-LCID and methods to obtain language, script, country and variant



Thank you ...

- ▼ ... for using LibreOffice!
- ▼ ... for supporting LibreOffice!
- ▼ ... for hacking LibreOffice!



All text and image content in this document is licensed under the [Creative Commons Attribution-Share Alike 3.0 License](#) (unless otherwise specified). "LibreOffice" and "The Document Foundation" are registered trademarks. Their respective logos and icons are subject to international copyright laws. The use of these therefore is subject to the [trademark policy](#).