New in ICU & CLDR

Markus Scherer (Google)
Steven R. Loomis (IBM) @srl295

Internationalization & Unicode Conference 42
September 2018 — Santa Clara, CA

Unicode Standard(s)

- Encodes all modern world languages
  - Lossless data exchange
  - Additional symbols & historic characters continue to be added
- Efficient processing; single-binary global software
- Unicode Locale Data (CLDR)
- Unicode Collation Algorithm
- IDNA, Security, regex, ...

But...

- Several large specs + Annexes
- 137,374 characters in 11.0
- > 80 character properties, many multi-valued
- > 75 languages with comprehensive locale data
- > 100 sort orders
- Significant update every year
- Formatting, line-break, regular expressions...

Internationalization, Localization & Locales

- Requirements vary widely across languages & countries
  - Sorting
  - Text searching
  - Bidirectional text processing and complex text layout
  - Date/time/number/currency formatting
  - Codepage conversion
  - … and so on
- Performance is key
  - It might be easy to do the right thing
  - It is hard to do with high performance + small footprint

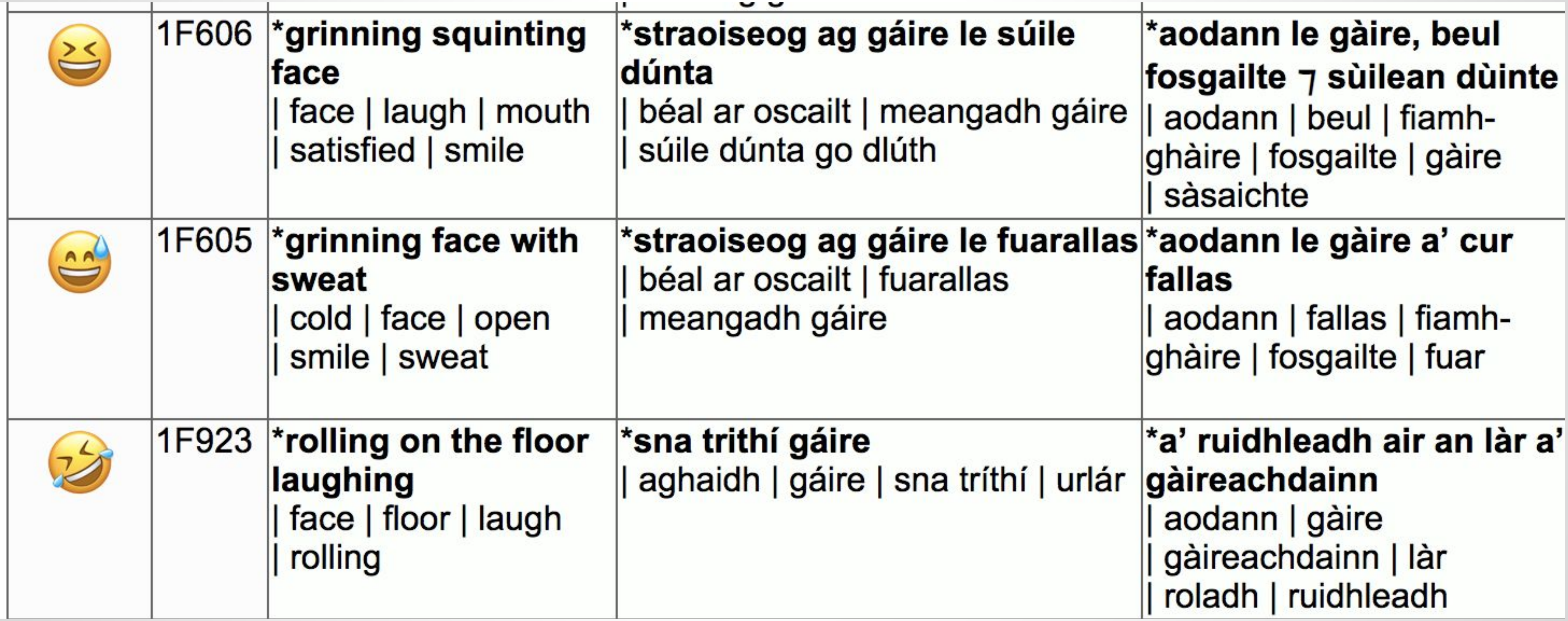
CLDR History

- Localization support requires data.
- ICU’s original data set from IBM data
- 2003: Discussion and comparison with other vendors → “**Common Locale Data Repository**” v1.0 under openi18n.org (Linux Foundation)
- 2004: v1.1+ under Unicode

CLDR Structure

- UTS #35 Locale Data Markup Language
    - English: `<day type="tue">`**`Tuesday`**`</day>`
    - Spanish: `<day type="tue">`**`martes`**`</day>`
- Linguistic (textual) data + metadata
    - Patterns for formatting and parsing
    - Translations for languages, regions, etc.
    - Language & script metadata
    - Regional information
    - ISO & BCP 47 code support
    - Keyboard layouts
- Crowd sourcing and curation
    - Two voting cycles per year
    - 1 million votes in v34 by 384 submitters

CLDR emoji annotations



😆	1F606	***grinning squinting face** | face | laugh | mouth | satisfied | smile	***straoiseog ag gáire le súile dúnta** | béal ar oscailt | meangadh gáire | súile dúnta go dlúth	***aodann le gàire, beul fosgailte ⁊ sùilean dùinte** | aodann | beul | fiamh-ghàire | fosgailte | gàire | sàsaichte
😅	1F605	***grinning face with sweat** | cold | face | open | smile | sweat	***straoiseog ag gáire le fuarallas** | béal ar oscailt | fuarallas | meangadh gáire	***aodann le gàire a' cur fallas** | aodann | fallas | fiamh-ghàire | fosgailte | fuar
🤣	1F923	***rolling on the floor laughing** | face | floor | laugh | rolling	***sna trithí gáire** | aghaidh | gáire | sna tríthí | urlár	***a' ruidhleadh air an làr a' gàireachdainn** | aodann | gàire | gàireachdainn | làr | roladh | ruidhleadh

ICU Features

- Unicode text handling
- Charset conversions (200+)
- Charset detection
- Collation & Searching
- Unicode Locale Data (CLDR)
- Resource Bundles
- Calendar & Time Zones
- Unicode Regular Expressions
- ...

- Breaks: word, line, …
- Formatting
  - Date & time
  - Durations, intervals
  - Messages
  - Numbers, currencies
  - Measurement units
  - Plurals
- Transforms
  - Normalization
  - Casing
  - Transliterations

ICU Works Everywhere

- Mature, widely used set of C/C++ and Java libraries
  - Basis for Java 1.1 internationalization, but goes far beyond Java 1.1
- Very portable – identical results on all platforms/programming languages
  - C99/C++11 (ICU4C): many platforms/compilers
  - Java 7 (ICU4J): Oracle JRE, OpenJDK, IBM JRE, Android
  - PyICU & other wrappers
- Customizable & Modular

ICU Is Kept Up To Date

- Part of Unicode since 2016 (ICU-TC)
- ≥2 ICU releases per year
- Each ICU release supports the latest
  - Unicode version
    - Properties, Unicode collation, IDNA, spoof checker, line breaks, ...
  - CLDR version
  - Time zone database update
- TZ DB updates for past ICU versions
- Open source (since 1999) – but non-restrictive
  - Contributions from many parties (IBM, Google, Apple, Microsoft, Yahoo, ...)

Backwards Compatible

- C & Java binary compatible
- C++ source compatible across ICU versions
  - Occasional changes (const) for subclasses
- API rarely deprecated, kept functional if possible
- Updated data & behavior; bug fixes

Two Billion Devices

ABAS Software, Adobe, Amazon (Kindle), Amdocs, Android, Apache (Harmony, Lucene, Solr, PDFBox, Tika, Xlan, Xerces, ...), Appian, Apple, Argonne National Laboratory, Avaya, BAE Systems Geospatial eXploitation Products, BEA, BluePhoenix Solutions, BMC Software, Boost, BroadJump, Business Objects, caris, CERN, Debian Linux, Dell, Eclipse, eBay, EMC Corporation, ESRI, FreeBSD, Gentoo Linux, Google, GroundWork Open Source, GTK+, Harman/Becker Automotive Systems GmbH, HP, Hyperion, IBM, Inktomi, Innodata Isogen, Informatica, Intel, Interlogics, IONA, IXOS, Jikes, Library of Congress, LibreOffice, Mathworks, Microsoft, Mozilla, Netezza, Node.js, OpenOffice, Lawson Software, Leica Geosystems GIS & Mapping LLC, Mandrake Linux, OCLC, Oracle (Solaris, Java), Progress Software, Python, QNX, Rogue Wave, SAP, SIL, SPSS, Software AG, SuSE, Sybase, Symantec, Teradata (NCR), Trend Micro, Virage, webMethods, Wine, WMS Gaming, XyEnterprise, Yahoo!, ...

ICU in Apple/Google/IBM/Microsoft

Apple

- Mac OS X, iOS (iPhone, iPad, ...), watchOS, tvOS, Windows apps, Safari, iTunes, ...

Google

- Web Search, Android, Chrome/Chrome OS, Adwords, Google+, Google Maps, Blogger, ...

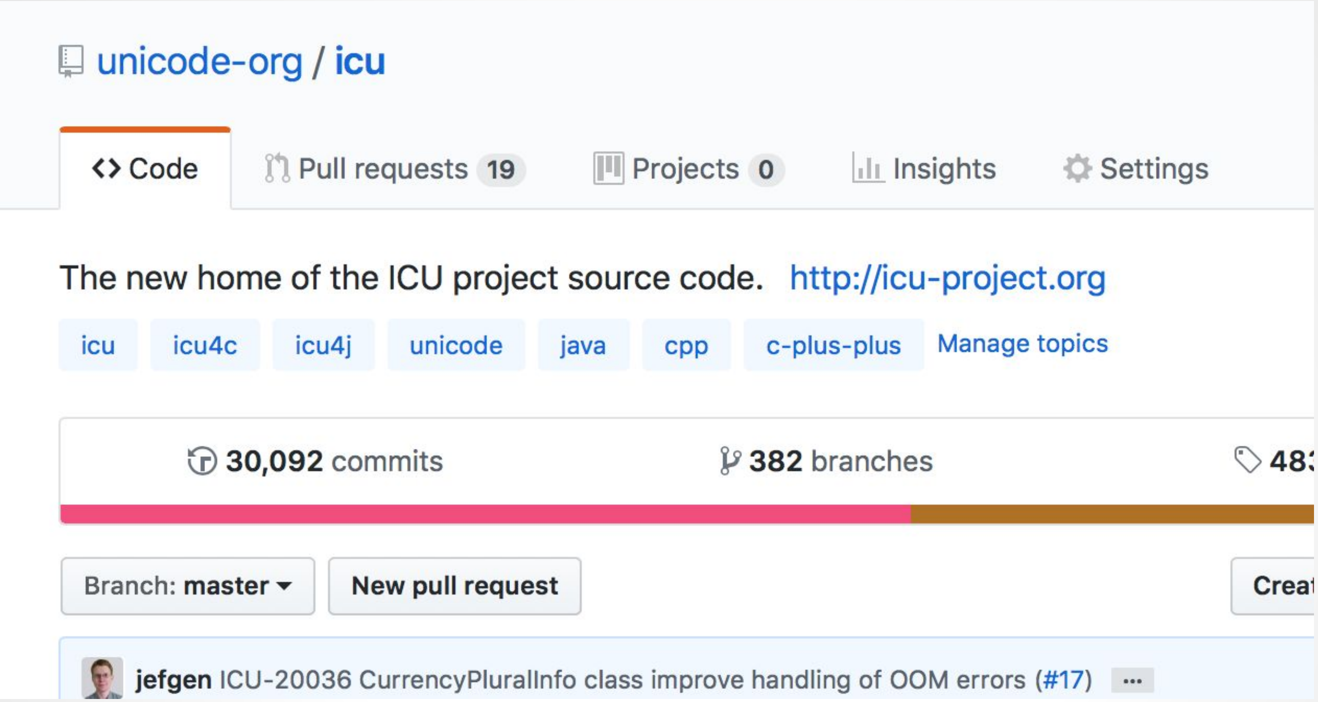
IBM

- DB2, Lotus, WebSphere, Tivoli, Rational, AIX, IBM i, z/OS, ...

Microsoft

- Windows Bridge for iOS, Windows 10 - Creators Update, Visual Studio 2017 [Electron], Visual Studio Code [Electron], ChakraCore

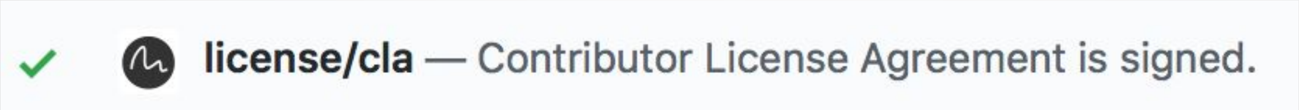
ICU Source Code on GitHub…



The new home of the ICU project source code. http://icu-project.org

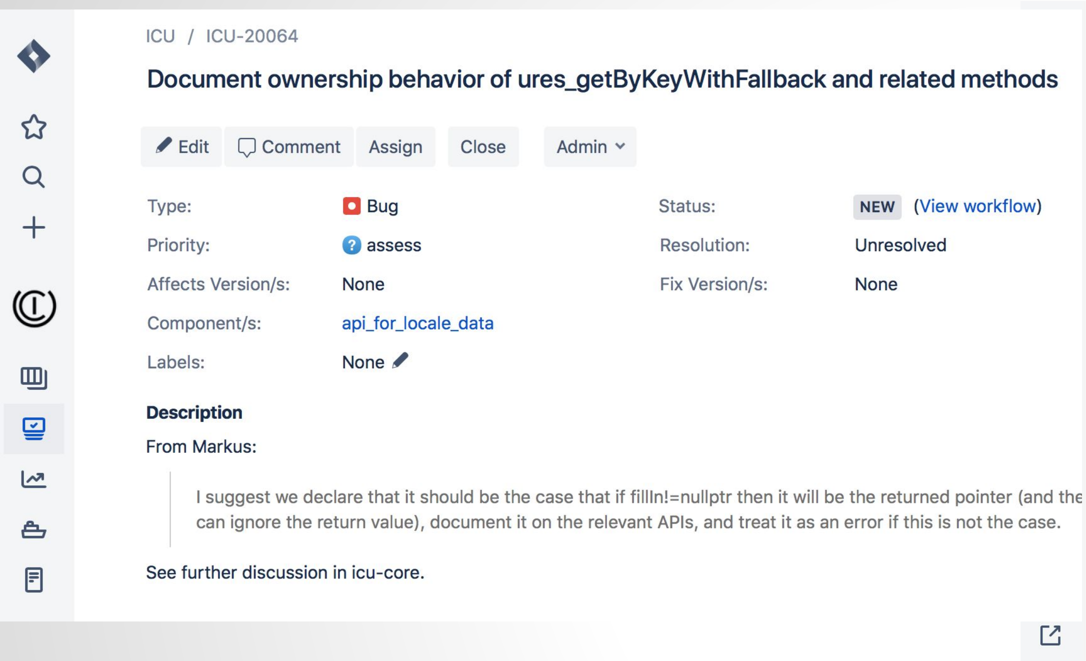
jefgen ICU-20036 CurrencyPluralInfo class improve handling of OOM errors (#17) …

- Moved from SubVersion to git
  https://github.com/unicode-org/icu
- Pull Requests welcome, provided that:



✓ license/cla — Contributor License Agreement is signed.

…and ICU tickets now on Jira



ICU / ICU-20064

Document ownership behavior of ures_getByKeyWithFallback and related methods

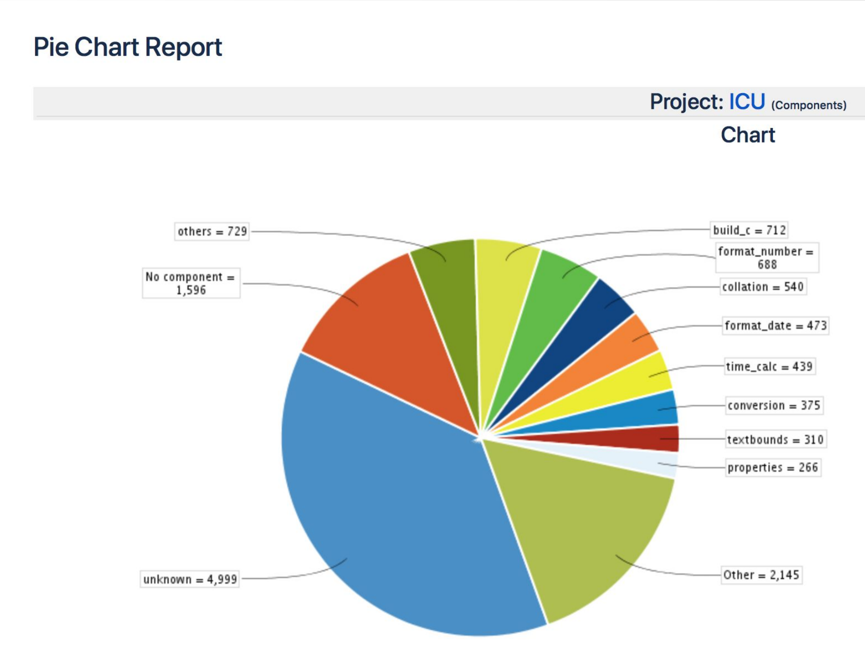
Type:	Bug	Status:	NEW (View workflow)
Priority:	assess	Resolution:	Unresolved
Affects Version/s:	None	Fix Version/s:	None
Component/s:	api_for_locale_data		
Labels:	None		

Description

From Markus:

> I suggest we declare that it should be the case that if fillIn!=nullptr then it will be the returned pointer (and the can ignore the return value), document it on the relevant APIs, and treat it as an error if this is not the case.

See further discussion in icu-core.



Pie Chart Report

Project: ICU (Components)

Chart

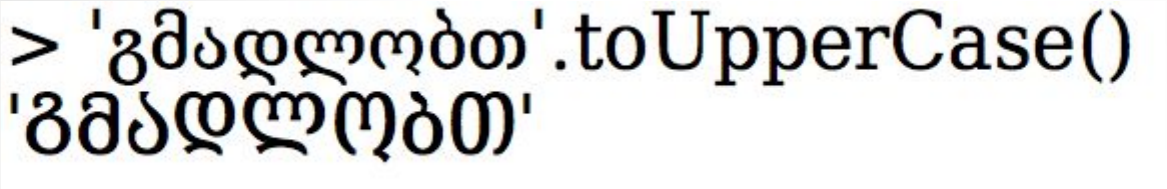
- All Trac content migrated to Atlassian Cloud https://unicode-org.atlassian.net
- Trac ticket #1234 becomes ICU-1234

What’s new in ICU & CLDR

- Unicode 11.0
  - Quickly made available via CLDR 33.1 & ICU 62
- CLDR 33, 33.1 & 34
- ICU 61, 62, 63

What's new in Unicode 11

- 684 new characters
- Capital letters for Georgian



```
> 'გმადლობთ'.toUpperCase()
'ᲒᲛᲐᲓᲚᲝᲑᲗ'
```

- 7 new scripts: Hanifi Rohingya, Medefaidrin, 5 historic
- 5 urgently needed CJK unified ideographs
  - 3 chemical names
  - 2 for Japan government
- + Mayan Numerals, historic Sanskrit, Gurmukhi, ...

Unicode properties API

New support for standard properties

- Extended_Pictographic (emoji)
- Indic_Positional_Category
- Indic_Syllabic_Category
- Vertical_Orientation

Via old API: (code point, property) → value

Via new API: property → (c.p. → value)

Custom properties: sets, and now maps & tries

CLDR 33 & 34

- Unicode 11.0
    - Also UCA & Unihan 11.0
    - Emoji 11 annotations
- Emoji keywords
- Odia & Assamese
- Base-Arabic ASCII digits option
- Keyboard layouts

Arabic digits

Arabic language native digits ٠١٢٣٤٥٦٧٨٩

- Customary in many countries
- Default for base language

ASCII digits 0123456789

- Customary in some countries
- More widely understood
- Better for UIs with single Arabic option

Now single-value change to switch “ar” default

- Does not change regional variants

Number & currency formatting & parsing

New number & currency parsing implementation

- sync C & J, bug fixes

Skeletons > patterns, e.g., MessageFormat

```
"Number of files:
  {num, number, :: group-min2}."
```

C: UNumberFormatter (skeleton, not fluent)

New NumberRangeFormatter

```
NumberRangeFormatter.with()
   .identityFallback(RangeIdentityFallback.
       APPROXIMATELY_OR_SINGLE_VALUE)
   .numberFormatterFirst(NumberFormatter.with().
       unit(MeasureUnit.METER))
   .numberFormatterSecond(NumberFormatter.with().
       unit(MeasureUnit.KILOMETER))
   .locale(ULocale.UK)
   .formatRange(750, 1.2)
   .toString();
```

→ "750 m - 1.2 km"

BreakIterator

Easier to write custom rules:

- “Safe” rules no longer used (ignored)

Updates for Unicode 11

Full conformance with UAX #14 line break

Finnish line break → root (data size -150kB)

Under the hood

Number formatting via double-conversion lib

Bug fixes: ErrorProne, FindBugs, Coverity

CI: ThreadSanitizer & AddressSanitizer

Java 7 minimum, but supporting Java 11+

C++11

- No more default `using namespace icu;`
- C++11 compiler required
- ICU 58 UChar: uint16_t / wchar_t / char16_t
- ICU 59 UChar: always char16_t in C++
  - Works seamlessly with u"string literals"
  - Breaking change!
- Conversion helper types & classes available
- UTF-8 source code
  - `UnicodeString(u"Πατάτα")`

Upcoming

2019q1

- Simultaneous Unicode 12, CLDR 35, ICU 64

References

http://site.icu-project.org/

- Downloads, bug reports
- User Guide
- Demonstrations
- Mailing lists (design & support)

http://cldr.unicode.org/

- Downloads, bug reports
- Specs & process

This presentation: https://goo.gl/jpjU4e