Formatting Dates Correctly: Genitive Month Names in `strftime()`

State of the work in progress

Rafał Lużyński
rluzynski@fedoraproject.org
UPDATE: Just before showing these slides in public I learned that this problem probably does not apply to the Czech language. Please ignore any references to the Czech language in the following slides.

Other languages should be reviewed, too.
Question: Who uses GNOME or: MATE, Cinnamon, Ubuntu Unity, etc… AND Czech locales or: Belarusian, Catalan, Croatian, Finnish, Greek, Lithuanian, Polish, Russian, Slovak, Ukrainian.

UPDATE

Czech probably is not affected Czech locales
Question: Who uses GNOME

or: MATE, Cinnamon, Ubuntu Unity, etc…

AND

UPDATE: Czech probably is not affected

or: Belarusian, Catalan, Croatian, Finnish, Greek, Lithuanian, Polish, Russian, Slovak, Ukrainian (and several more…)

?
What’s wrong here?
What’s wrong here:

UPDATE

Czech probably is correct

18:52

19:38:45

27 stycznia

23:40

dijous 27 de abril

d'abril
We need genitive cases!

- Some (most?) Slavic languages have different suffixes for different cases
- The same: Baltic languages, Finnish, Greek
- The rules are too complex to be resolved programmatically
- Some Romance languages use “de” to create a genitive case but need “d’” if the word begins with a vowel
About 20 languages affected

- Armenian
- Asturian
- Belarusian
- Catalan
- Croatian
- Czech
- Finnish
- Greek
- Kashubian
- Lithuanian
- Ossetian
- Polish
- Russian
- Scottish Gaelic
- Silesian
- Slovak
- Sorbian (Upper, Lower)
- Ukrainian
- Walloon
- ...anyone else?
About 20 languages affected
Is this severe at all?

Yes. Linux desktops promote bad grammar.

This makes them unsuitable for schools.
Suggestion:

If we need genitives then why not just reword all months names to genitive?
Here is what Blogspot did:

**UPDATE** for those who don’t know: this is all incorrect. Nominative cases are required here.
We need both cases!

Czech probably is correct here

This is incorrect

This is correct

UPDATE
Czech probably is correct here
Why the bug?

- All GNOME/GTK+ applications use this function:

```c
gchar *g_date_time_format (GDateTime *datetime,
                          const gchar *format);
```

- It is inspired by `strftime()`:

```c
size_t strftime(char *s, size_t max, const char *format,
                const struct tm *tm);
```
Format specifiers

- `%b` – abbreviated month name,
- `%B` – full month name,
- `%m` – month (decimal number),
- `%Om` – month (alternative numeric system),
- and so on...

But there are no genitive cases!
Implementations

Both these functions internally use nl_langinfo():

- MON_1 – localized January,
- MON_2 – localized February,
- ABMON_1 – localized Jan,
- ABMON_2 – localized Feb,
- and so on...

Again no genitive cases!
So it’s a bug in glibc!

https://sourceware.org/bugzilla/show_bug.cgi?id=10871
Solution

- Add `ALTMON_1...ALTMON_12` items to `nl_langinfo()`
- Add `%OB` format specifier to `strftime()` and anything derived, inspired etc.
- Let `%OB` return the same string as `nl_langinfo(ALTMON_1...ALTMON_12)`
Solution

- Let `nl_langinfo(MON_...)` and `strftime("%B")` return the *genitive* case.
- Let `nl_langinfo(ALTMON_...)` and `strftime("%OB")` return the *nominative case* (the same as `nl_langinfo(MON_...)` and `strftime("%B")` return now).

Wait, *WHAT*?!
Why this *incompatibility*?

- *BSD family* (including FreeBSD, OpenBSD, OS X, iOS) do the same since 1990s
- POSIX also agreed for the same solution in 2010 to be included in a future release:
  
  http://austingroupbugs.net/view.php?id=258
  (but has not yet included it in any release)

- Otherwise we would never be compatible with POSIX and BSD
- How should we implement `g_date_time_format()` from glib2?
  Compatible with glibc? Compatible with POSIX? Compatible with OS X? Platform dependent (nonportable)?
Why this *incompatibility*?

- Month names are probably more often used to display dates than standalone
- This approach will automatically fix all applications which display dates incorrectly
- Also, unfortunately, will break some which display months standalone (e.g., calendars)
Near future

- `nl_langinfo (ALTMON_...)` and `strftime (%OB)` will be added to glibc
- But only provided that it is not yet defined which of `MON_x/ALTMON_x` and `%B/%OB` is nominative and which is genitive
- In case of `strftime()`, language communities may choose different approaches
- We want to hear feedback from translators, application developers, users,...
Why not go one step further?

- Do we also need `strftime ("%Ob")` (abbreviated alternative month name)?
Why not go one step further?

- Yes, we need it at least for Russian:

<table>
<thead>
<tr>
<th>Nominative:</th>
<th>Genitive:</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>мар</td>
<td>мар</td>
</tr>
<tr>
<td>апр</td>
<td>апр</td>
</tr>
<tr>
<td>май (red)</td>
<td>мая (green)</td>
</tr>
<tr>
<td>июн</td>
<td>июн</td>
</tr>
<tr>
<td>июл</td>
<td>июл</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Why not go one step further?

- No other system supports it
- Fedora will be First™ :-) (again...)
Who does it correctly

*BSD family (FreeBSD, OpenBSD, OS X, iOS):

- `nl_langinfo(nl_item item)` accepts `ALTMON_1...ALTMON_12`
- `strftime()` supports "%OB"
Who does it correctly

Microsoft:

- `GetDateFormat()` and `GetDateFormatEx()` automatically select genitive form when both "d" and "MMMM" appear in the format string
  Do you want to see the case where it does not work?

.NET Framework:

`System.Globalization.DateTimeFormatInfo` supports `MonthGenitiveNames` and `AbbreviatedMonthGenitiveNames`
Who does it correctly

LibICU
(International Components for Unicode):

Date format string includes:

- "M",
- "MM",
- "MMM",
- "MMMM" – month names in full date context (genitive)
- "L",
- "LL",
- "LLL",
- "LLLL" – month names standalone (nominative)
Who does it correctly

KDE and QT:
(based on libicu)
Who does it correctly

Android

- Written in Java
- `java.text.SimpleDateFormat` internally based on ICU
- This means: able to handle nominative and genitive months names correctly!
- Sometimes locales are incomplete
- Sometimes applications use it incorrectly
Who does it correctly

Ukrainian locales in glibc (sic!)

- Dirty hack
- "%OY", "%Om", "%Od", "%OH", "%OM", "%OS" were supposed to use alternative numeric symbols
- They defined alternative digits as: "0", "січня", "лютого", "березня", and so on...
- Result: "%Om" displays the month name in a genitive case
- Fallout: "%OY", "%Od", "%OH", "%OM", "%OS" also display months
- nl_langinfo() remains not fixed
Why not yet finished?

- It’s not easy to tweak in glibc
- 200+ locales and zillions of applications on multiple hardwares, don’t break any of them!
- No reviewers from Eastern Europe so far

Contributors needed!