Programs, debconf, man pages translation using gettext : a tutorial for skilled and less skilled Debian developers and translators

Christian Perrier

5th Debian Conference, Helsinki, Finland

This work is dedicated to René Cougnenc

Outline

What is in this tutorial?

- handling localisation in already internationalised software
- handling Debian specific localisation in packages
- a too short tutorial

What is in this tutorial?

- handling localisation in already internationalised software
- handling Debian specific localisation in packages
- a too short tutorial

What is in this tutorial?

- handling localisation in already internationalised software
- handling Debian specific localisation in packages
- a too short tutorial

What is NOT in this tutorial?

- internationalising software (NOT a gettext course)
- character encodings explanations
- it is INFORMAL...
- ...and badly prepared

A few reminders

- Internationalisation (i18n)
  - Get software ready to work with different languages in different countries
- Localisation (l10n)
  - Translate software and documentation to different languages
- Multilingualisation (m17n)
  - Get software ready to be used in several languages at the same time
- Globalisation (g11n)
  - The whole picture
- Bubullisation (b11n)
  - The pain for you to be here and listen to me talking tooo much

Outline

Let's find the places we'll get hurt

- GNU hello
  - a good example of an already i18n'ed software.
  - a good example of Debian package

Outline

Hello compilation and i18n

- Let’s dig in it NOW

LESSON Nr. 1

- Software author: thou shalt comment the source
- Translator: thou shalt whine and request comments

The mysterious PO files

- Usually lie in a po/ directory
- PO==Portable Object files
  - `msgfmt` compile to MO files
- Advantages:
  - text files
  - simple format
    - header: file contents and information about translator(s)
    - strings: the msgid/msgstr couple
- Caveat(s):
  - depends on English for string IDs

Important files in a po/ directory

- `LINGUAS`
  - Lists the supported languages (PO–>MO)
  - Depending on build process, also see `LINGUAS` or `ALL_LINGUAS` variables
- `POTFILES{.in}`
  - Defines WHICH files contain translatable material
  - Reference for (auto)build tools to list the supported languages
- The POT file
  - PO “template” file
  - Contains the translatable material (list of msgids)
- The PO files
  - One per language (ISO-639 code AND NOTHING ELSE)
  - Contains the translations
  - May use various encodings. DO NOT MESS THEM.

LESSON Nr. 2

- Thou shalt not change msgstr
- Thou shalt not change encoding without reason
- Thou shalt not use xx_YY file names except:
  - pt_BR/pt
  - zh_CN/zh_TW
  - pa_IN/pa_PK

Let's translate Hello to Klingon

- ISO code name?
  - `grep -i klingon /usr/share/iso-codes/iso_639.tab`
- `cp hello.pot tlh.po`
- Header fields

LESSON Nr. 3

- Properly fill in the header fields
- DO NOT change header fields

Status of translations

The magic `msgfmt` utility:

- `msgfmt -check`
- `msgfmt -statistics`
- Fuzzy? Untranslated?
- Changing upstream strings “fuzzies” the translations

Other useful utilities

- `msgcat`
  - Reformat and/or concatenate files
- `msgmerge`
  - Merge msgids of a POT file into a PO file
- `msgattrib`
  - Manipulate translation attributes (fuzzy/untranslated/obsoleted)
- `msgfilter`
  - Apply a filter to all translations of a translation catalog
- `msguniq`
  - Deal with duplicates msgid's
- `msguntypot`
  - NOT IN GETTEXT: deal with typo corrections

Outline

What's involved?

- `*.templates` file : the reference file(s)
- po directory
- `debconf-updatepo`
- `dh_installdebconf` magic

LESSON Nr. 4

- Write good English in templates
  - `debian-l10n-english`
- Use consistent style
  - Read the Developer's Reference!
- Make more use of common templates
  - www-config
  - dbconfig-common

A few tricks with debconf templates

- podebconf-display-po
  - Display translated debconf templates as they should be
- podebconf-report-po
  - Warn translators about needed updates

Outline

Why are man pages translations painful?

- Many translation projects outside projects
- Traditional model: maintenance is painful
- No relation between original and translation

Thou shalt use PO

- Translators are used to PO files
- Good tools exist
  - po4a
  - xml2po (gnome-doc-utils)
  - poxml

po4a: Martin and Denis save our souls

- Strings are extracted from the original
- Translated into the PO
- Reinject translations in a new document
-
  - po4a ROCKS

The journey of a translated man page

- Re-use an existing translation: `po4a-gettextize`
- Update the PO file wrt original: `po4a-updatepo`
- Complete/Correct/Break the PO file
- Generate the translated man page: `po4a-translate`
-
  - po4a ROCKS

po4a towards World Domination

- Very few packages use po4a for man pages
  - apt-show-versions
  - po4a
- Internally used in the French team
- Thou shalt be po4ayed
  - dpkg
  - apt
-
  - po4a ROCKS

LESSON Nr. 5

- Thou shalt switch thy man page to po4a
  - Ask for help on `debian-i18n` mailing list
-
  - po4a ROCKS

PO rUleZ

- Just Po It
-
  - po4a ROCKS