



Zunächst sollten alle wichtigen Dateien mit `git add` aufgenommen werden und anschließend alles gelöscht werden, was keine Quelldatei ist:

```
$ git clean -df
```

Nun müssen zu übersetzende Zeichenketten markiert werden. C-Code-Quelldateien, die Zeichenketten enthalten, welche übersetzt werden müssen, müssen dazu diesen Header einbinden:

```
#include <glib/gi18n.h>
```

Zum Markieren wird z.B. "My Text" ersetzt durch `_("My Text")`. Wenn zudem Kontextinformationen für die Übersetzer mitgegeben werden sollen, dann können andere Markierungen verwendet werden; diese sind in der GLib-Referenz erklärt.

.ui-Dateien werden so markiert:

```
<property name="..." translatable="yes">My Text</property>
```

Ebenso muss die `main()`-Funktion nun Code aufrufen, um die Übersetzungen zu benutzen. Dazu muss in `meson.build` eine Konfigurationsdatei deklariert werden, in der eine Konstante `GETTEXT_PACKAGE` gesetzt wird:

```
cdata = configuration_data()
cdata.set_quoted('DATADIR',
    join_paths(get_option('prefix'), get_option('datadir')))
cdata.set('GETTEXT_PACKAGE', 'mycopyshop')
configure_file(output : 'config.h', configuration : cdata)
```

In `main.c` muss folgendes eingebunden werden:

```
#include <locale.h>
#include <config.h>
```

und dann passende Aufrufe gemacht werden; diese sind auch in der GLib-Referenz zu finden:

```
setlocale (LC_ALL, "");
bindtextdomain (GETTEXT_PACKAGE, DATADIR "/locale");
bind_textdomain_codeset (GETTEXT_PACKAGE, "UTF-8");
textdomain (GETTEXT_PACKAGE);
```

Nun soll Meson aus den Markierungen Dateien mit den noch nicht übersetzten Texten erzeugen. Für diese wird ein Ordner `po` im Hauptprojektordner angelegt und in `meson.build` mit `subdir('po')` eingebunden.

Darin wird eine Datei `po/meson.build` erstellt:

```
i18n = import('i18n')
i18n.gettext('mycopyshop', preset : 'glib')
```

In der Datei `po/POTFILES` sollten alle Dateien mit zu übersetzenden Zeichenketten aufgelistet werden, zum Beispiel so:

```
# List of source files which contain
# translatable strings.
src/cshopapplication.c
src/cshopraw.c
src/cshopwindow.c
src/main.c
src/ui/menu.ui
src/ui/toolbar.ui
```

Damit ist die Internationalisierung des Projekts abgeschlossen. Zum Lokalisieren ins Deutsche muss in die Datei `po/LINGUAS` eine Zeile `de` eingefügt werden, für Englisch eine Zeile `en`, U.S.-Englisch `en_US` u.s.w.

Am Ende sieht `po/LINGUAS` so aus:

```
de
en
```

Die Zeilen sollten alphabetisch sortiert sein. Nun kann eine Grundlage für die Übersetzung gebaut werden:

```
$ mkdir builddir
$ cd builddir
$ meson ..
$ ninja mycopyshop-pot
```

Copyright-Informationen und „Report-Msgid-Bugs-To:“ sollte im Kopf der soeben erzeugten Datei `po/mycopyshop.pot` mit einem Texteditor eingetragen werden.

`msginit -l en --no-translator` bzw. `msginit -l de --no-translator` erzeugt dann entsprechende Dateien im `po`-Ordner, wo z.B. mit `gtranslator` die Übersetzung eingetragen werden kann (Deutsch und Englisch haben beide „`nplurals=2; plural=(n != 1);`“ laut `Gettext-Handbuch`). Alle neuen Dateien außer `builddir` sollten mit `git add` im Git-Repository gesichert werden.

Neubauen des Projekts erzeugt nun auch die übersetzte Fassung. Diese kann ausgeführt werden mit `LANGUAGE=en_US.UTF-8 LC_ALL=en_US.UTF-8 copyshop`.