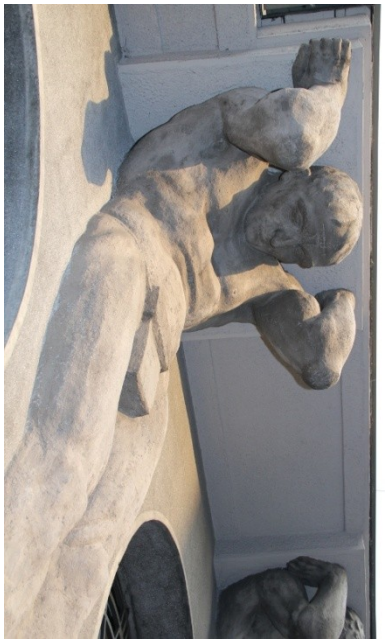


BANKA SLOVENIJE

BANK OF SLOVENIA

EUROSYSTEM

Easily translatable Shiny applications

Matjaž Jeran – Financial statistics

uRos 2018 on 14-September-2018

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Contents

- Introduction – why translatable applications?
- What to translate?
- Examples in FOSS: GNU gettext
- Principles shown on a simple R program
- Details about internationalization and localization
- Conclusions (and a demo)

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

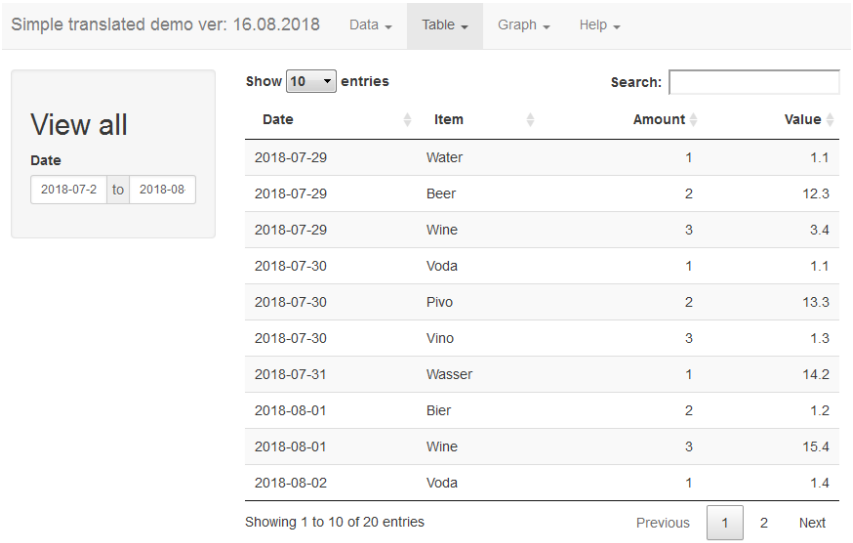
Introduction – Why?

- Why applications in more languages?
- Required by law or targeting international community: e.g. Canada, Switzerland, Belgium, Slovenia, EU, Eurostat, UN
- Desire: one code – many language implementations – simplify maintenance
- Shared development

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

What to translate?

- User interface: windows titles, drop-down menus, forms
- Data structure: input data - data.frames, lists, …
- Data contents: use of codelists?



Any examples to learn from?

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

- R, Rcmdr, rattle, unix, linux programs …
- Set of messages is read from the language file into messages matrix at start of execution
- **C – standard: gettext** and .po and .mo files - see R/library/translations R/library/Rcmdr/po
- Use of ISO-639 (languages) and ISO-3166 (countries)
- Lang. support for cases, singular, dual, plural
- Latest development: R package shiny.i18n

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Simple program

- Simple program
- Intervoven messages and code

```
print ("Hello world")
```

- Separating messages from the code
- All messages are collected at one place
- Code should not change

```
mesg <- "Hello world"
print (mesg)
```

Preparing for translation

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

- All messages of all languages collected in a matrix
- Language selector determines behaviour
- The same set of code for all languages

```
lang <- "en"

mesg <- matrix (
c ("Hello world", "Pozdravljen svet"),
nrow = 1, ncol = 2, dimnames =
list (
c ("Hello world"),
c ("en", "sl")))

print (
mesg ["Hello world", lang])
```

Adding another language eg. French?

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

- Adding more languages requires one more column in the messages matrix
- Adjusting languge selector
- The code remains the same

```
lang <- "fr"

mesg <- matrix (
c ("Hello world", "Pozdravljen svet", "Bonjour le monde"),
nrow = 1, ncol = 3, dimnames = list (
c ("Hello world"),
c ("en", "sl", "fr")))

print (
mesg ["Hello world", lang])
```

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Using any language

- Read message matrix from external translation file for selected language
- Tab delimited trans. file
- Structure of submaps named by languages and teritories
- Warning for non-synchronized code and messages!

```
lang <- "it"
mesg <- read.table (file =
file.path (lang, "mesg.txt"),
header = TRUE, sep = "\t",
stringsAsFactors = FALSE)
rownames (mesg) <- mesg [, 1]
print (
mesg ["Hello world", lang])
```

Contents of mesg.txt:

en	it
Hello world	Buon giorno il mondo

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Simulating gettext in R

- Non-synchronized code and messages can be treated by a function **gettxt (msg, lang)** that includes the error processing if the translation is missing
- R function gettext - API to GNU library – can we have our own .po file?

```
lang <- "it"
mesg <- read.table (file =
file.path (lang, "mesg.txt"),
header = TRUE, sep = "\t",
stringsAsFactors = FALSE)
rownames (mesg) <- mesg [, 1]
print (
gettxt ("Hello world", lang))
```

Contents of mesg.txt:

en	it
Hello world	Buon giorno il mondo

Shiny translatable data structures – str 1

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

- Spreadsheet input data
- Inheritance of names?
- Beware: data column names may be natural language dependent!

Date	Item	Price
	Water	

Datum	Opis	Cena

Date	Description	Prix
	L'eau	

- Safe coding: columns should be referred only by position - not by the name:
  not data$Price but data[, 3]
- Before display data use text matrix:
  - to rename column names to target language
  - Use names parameters in methods to display tables and graphs

Shiny translatable data structures – str 2

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

- Spreadsheet input data in predefined language
- Inheritance of names
- Code is not affected by insertion or deletion of columns

Date	Item	Price
	L'eau	

- read.table (…header=TRUE,)
- Columns could be referred by column name. Rename columns of a copy to target language just before display

```
data.colnames<- c(gettxt ("Date",lang), gettxt("Item"…

output$T22table <- DT::renderDataTable (
        {
          tmp <- subset (dat, subset = input$dates22 [1]
      <= Date & …)
          colnames (tmp) <- data.colnames
          return (tmp)
        } )
```

Internationalization and localization

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

- Cultural specifics: should be supported in R, shiny, javascript and operating system
- Number formatting is controlled by operating systems and R settings options (OutDec = ",")
- Package DataTables should have language plug-in (for non-English)
- The settings can be obtained and set by: Sys.getlocale() Sys.setlocale ()

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Definition of language

Can be defined on the server or on the client

- **server** – introductory web page with possibilities
- **client** – web page using javascript to read the browser language settings and transfer the value to Shiny

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Conclusions

- **Benefits**
  - Development and maintenance is reduced to one copy
  - Many translators not involved with programming
- **Risks**
  - Requires disciplined programming practice
  - More work if data structure changes
- **Some new features in R may be handy**

BANKA SLOVENIJE
BANK OF SLOVENIA
EUROSYSTEM

Working examples

- Multilingual Shiny applications in the Bank of Slovenia:
  - Financial accounts
  - Currency rates
  - Balance of payments for energy
  - Real estate market
- uRos 2018 – Shiny demo program