A LaTeXperiment of hieroglyphic typesetting

Serge Rosmorduc
rosmord@lifac1.ens-cachan.fr

December 10, 1993

1 Licence

Sesh Nesout, a TeX package for hieroglyphic typesetting Copyright (C) 1993 Serge Rosmorduc

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 1, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

2 Introduction

This is a package that allows one to typeset hieroglyphs; one more ! It is now in a very early stage, but I want to come with a complete, easy-to-use, general program, that would allow one to build a database of Egyptian texts, and retrieve information in it. At the time being, I am quite far from this goal! Yet I think the current program can be amusing and even useful.

A second point is that there is also a hieroglyphic font with it. It should be bettered, but it exists. So, you can use it for your own programs. There are both a metafont and gsf (postscript type I) fonts. The fonts were made with the GNU fontutils.

3 Installation

You should copy the files EgypUtil.sty egypto.sty hierLtx.sty hiero.sty, and OT1diacr.fd where they belong. You might need to rename OT1diacr to T1diacr.fd. If you are using a Unix system, they should be on your TEXINPUTS path. The directory where your sources are should do in most cases, anyway.

Then, put your fonts where they can be used. On Unix system, it's usually somewhere in your TEXFONTS directory.

If you need another base resolution than 300dpi, you should get the metafont sources (HieroFonts) of the fonts and then make a 1.2, 1, 1/1.2, $1/(1.2)^2$ magnificated version of each hieroglyphic font (A-Z and Aa.mf); you should also build the diacritic fonts diacr7,8,9,10 and 12, and a magstep 0 version of the Egyp font (parts of cartouche and others).

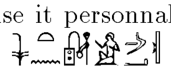
Then you should go to the SeshNesout directory and build the sesh program (read the README file). It is a text to latex converter, with a "*manuel de codage*"-like syntax[1].

4 Typesetting hieroglyphs

To type hieroglyphs, you need three things: the fonts, and the file `hiero.sty`, which is to be given as an option to the `\documentstyle` macro, and finaly the sesh program.

You might also want the *Inventaire des signes hiéroglyphiques en vue de leur saisie informatique,* [?] hereafter called *manuel de codage* (in English, French, and German).

4.1 Overview

There's a C program, `sesh` that follows more or less the "manuel de codage". It can be used to create hieroglyphic texts. I use it personnally as a filter in emacs. Its name is `sesh` (for Sesh Nesout Maâ ,  which was a bit long to type!

To use it, write your LaTeX source with hieroglyphic texts in a `hieroglyph` environment. Then, use sesh as a filter to obtain a LaTeXable result. As any polite program, sesh reads stdin and writes on stdout.

Signs are named, either after their translitterations (for example, `xpr` is ) or after their Gardiner Code. For a complete list, and documentation, see [?].

A few codes are used for signs variants; these are `pA'` and `k'` which write  and .

[1] If you'd rather do everything in LaTeX, make a list of each hieroglyphic font with "`tex testfont`", and use hierLtx.sty instead of hiero.sty. Yet this is a tedious way to go. It isn't supported anymore.

You can type adjacent signs with '-' , stack signs with ':', group signs with parenthesis, and use '*' to separate signs on the same level. <. . . .> writes a cartouche, '\' inverts a *sign*; and # . . .# shades a text.

For the moment, however, you should avoid using '*' and parenthesis for signs that are *not* stacked. I will change this.

I strongly recommend that you type the word-separation spaces. The package will then try to cut the lines at word-separations, which is nicer.

Example:

```
i-w -<-ra-mn:n-xpr-> -m-aH-a:Z1*pr-=f -#-mi-i-#
-r:a-ra-C2 -m -p*t:pt
```

If you use a version of TeX–Xet (a patch designed to allow you to type right-to-left text as well as left-to-right, with the benefit of the line-cutting system), It is possible to reverse a whole part of the text. Simply type +dg right-to-left, and +gd for left-to-right (this is *not* part of the *manuel*).

Example: , made like that:

```
+dg G7 E1:D40 xa:a m wAs-t:niwt nbty
+gd anx G7 E1:D40 xa:a m wAs-t:niwt nbty
```

Note that you can separate words with white spaces, without writing '-' before them. You can, anyway, follow the "manuel"

4.2 Macros

There is a limited macro facility (again, not part of the *manuel*). To define a macro, type

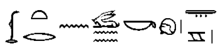
```
#def MACRONAME Body#
```

Note the second '#' sign. MACRONAME can be any unused sequence of letters, simple quote, and point.

This can be useful if a name appears often.

Example:

```
\begin{hieroglyph}
#def 'tA tA:N23*Z1#
tr-t:r-n-wn:n:n-k-tp-Z1-'tA
\end{hieroglyph}
```



Note that, once defined, these macroes can be used in all subsequent hieroglyph environment.

A few useful definitions:

```
#def Hm' N42#
#def bdS A7#
#def bin G37#
#def arm D40#
#def knife T30#
#def copper N34#
#def di' D37#
#def hrw h-r:W*ra-Z1#
#def ink nw:k':A1#
#def inr O39#
#def iw' D54#
#def k' V31A#
#def nn M22-M22-n:n#
#def nxt A24#
#def rmn D41#
#def king A42#
#def sxA A2#
#def wnn wn:n:n#
```

4.3 LaTeX macros

If you want the small signs to be on the baseligne, type `\SurLigne` out of a hieroglyph environment; the default is to center signs. To switch back to centered signs, type `\Centrer`. In a cartouche, the text is always centered. You can handle the line-cutting algorithm by changing the two TeX macros `\Hrp` and `\Hitmts`; the first rules what is inserted between signs, the second between words. The default values tend to protect you from cutting a word, at the cost of some white space.

5 Typesetting translitterations and references

The commands here described are defined in the `egypto.sty` file.

5.1 Settings

\Montitre{...} allows you to define the document's title (To be used in Cross-references)

\eg to use the translitteration font

\def\SourceTexte{Name of the text} to tell LaTeX what text you are typing

\def\EXEMPLE{example} To use this style in english

5.2 Translitteration

There are currently three ways to write some translitteration (apart from `\eg`)

1. the translit environment: it takes 3 arguments:
   - name of the text
   - page number or recto/verso (exxs `{4}` or `{recto}`
   - line number(s) (could be column number) (exxs. `{4}` `{4-9}`
   - an alternative (possibly empty) reference in free text.

   One important thing is that TeX "understands" these numbers. i.e. there are commands to change the line number and the page number. such things are usefull for cross-references

   you can type a translitteration, and use "`\traduction`" to start typing the translation.

   Example:

   ```
   \begin{translit}{O. foobar XIV}{verso}{10-15}{LES 12,2-5}
     iw.i rx.kw mdw nTr
   \traduction
     I know the hieroglyphs
   \end{translit}
   ```

   O. FOOBAR XIV, verso 10–15 LES 12,2-5

   ỉw.ỉ rḫ.kw mdw nṯr

   I know the hieroglyphs

2. the exemple environment. a translit environment with the word "Exemple" in front of it, and numbered. To write "example", use `\def\EXEMPLE{Example}`

   these two environments write a line in "`*.dic`" where `*` is the name of your TeX file.

   "translit" writes:

   ```
   \Citation{name of the source}{references}{document's title}{
                         page number in document}
   ```

   "exemple" writes:

   ```
   \Exemple{name of the source}{references}{document's title}
           {exemple number}{page number in document}
   ```

(all on the same line)

3. the `\traduction{}{}` macro. It takes two arguments, the first being a translitteration, the second a translation. you have separates foonotes and, by default, the text is given in two columns wich can spread over pages, which is useful to translate poetry. If you want the translation under the translitteration, you can type

```
\def\EcritTraduction{\EcritTraductionEnLigne}
```

and

```
\def\EcritTraduction{\EcritTraductionEnColonne}
```

to switch back. In the column version, the second column is a verse-like environment

5.3 Varia

\affligne shows the line number above a vertical line: $\overset{10}{|}$.

\affpage shows the page number in a cartouche: (verso).

`\|` increases the line number and shows it.

`\*` increases the page number and shows it. (the line number becomes 1)

\numligne{VALUE} gives a value to the line number and shows it

\numpage{VALUE} the same for page number.

\dico{Y}{translation}{comments} Can be used to make an index of terms. ex:

```
\dico{XAa}{to free}{transitive verb}
```

writes in the .dic file :

```
\DicoIndex {XAa}{ to free }{transitive verb }{P. Leyde I 350}
{verso,13}{name of the text}{2}
```

that is, what you wrote, plus the references. It is to be used inside an environment.

5.4 Grammatical signs and al.

The zero-subject (i.e. the empty set) is bound to `\zero`

There is a environment for typesetting grammatical rules: its name is gramrule. a word typed there appears in slanted font, and '`~X␣`' writes X in translitteration (the space is mandatory).

The possib environment allows to type different cases, with an accolade in front of them. (`\\` to part the cases)

The pile environment allows to write some text in a column.

Example of use:

```
\begin{gramrule}
~ir + \begin{possib}
       infinitive\\
       prospective ~sDm.f \\
       ~mrr.f \\
       \end{possib}
    + \pile{whatever\\you \\want\\}
 \end{gramrule}
```

$$\textit{ỉr} + \begin{cases} \textit{infinitive} \\ \textit{prospective } \textit{s\underline{d}m.f} \\ \textit{mrr.f} \end{cases} + \begin{array}{c} \textit{whatever} \\ \textit{you} \\ \textit{want} \end{array}$$

6 To Do

- Capital letters for translitteration (currently, it is quite hard to type "*Rʿ*".
- A WYSYWIG interface, first using X.
- Improve sesh. Improve the macro mecanism. Allow one to type some words in translitteration. Write a translitteration program. Implement additional codes. Add a startup file with definitions.
- Improve the signs (both baselines and look.)
- Add hieroglyphs in `\dico....`;
- fully and strictly implement the "manuel de codage".
- Add a possibility of reversed text in normal latex (not a priority. I think a built-in-tex support for right-to-left writings is a must for TEX, and that these kinds of extensions should be used).

7 Recently Done

- A right-to-left font, and a support for tex–xet.
- Improved sesh — I now use an hash table for the signs. The *manuel* is more or less supported, and should be so very quickly.
- Hieroglyphic typesetting improved. The sizes are better choosed, the signs can be merged with text, and are now centered on the base line.
- Can use NFSS2 now. (but still support the old system.)

8 Exemples

Here are a few examples, taken from various texts : Lefèbvre's *Grammaire de l'Égyptien Classique*[2], Gardiner's *Egyptian Grammar,* and Westcar.

Le pseudoparticipe

Il exprime particulièrement l'**état** ou la **condition** de qqn, ou de qq.ch. Deux cas sont à distinguer, selon que le substantif (ou pronom) auquel est apposé le pseudoparticipe est sujet ou objet du verbe de la phrase.

1. le pseudoparticipe peut qualifier un substantif (ou pronom) *sujet* de certains verbes, comme *wrš* “ passer tout le jour à ”, *sḏr* “ passer toute la nuit à ”, *prỉ* au sens de “ devenir ” ; *ḏr* “ finir ” (par faire telle ou telle chose). Ex.: *n sḏr s ḥkrw r dmỉ.ỉ* jamais un homme ne passa la nuit à avoir faim dans ma ville (*Menthuw.* 11). Litt. à l'état d'avoir faim.

   S'il est sourd et que sa bouche ne puisse plus s'ouvrir, tous ses membres deviennent faibles (*Ebers* 99, 20–21).

   ḏr.ỉn.f ḥms(w) à la fin il s'assit (*Leb.* 75).

   Comparer l'emploi, comme auxiliaires, devant *sḏm.n.f,* de *prỉ, sḏr* et *ḏr,* §331.

2. Il peut qualifier un substantif ...

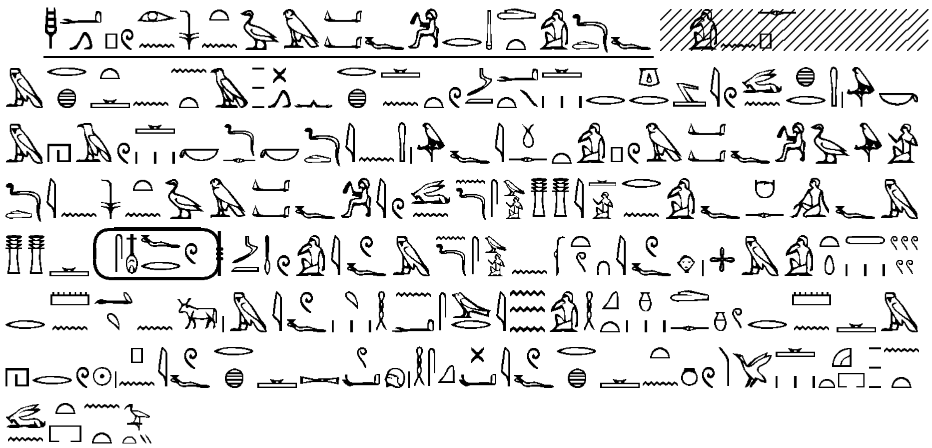
EXERCICE XXII

(*a*) *Translate into English :*

[2]Please, note that the style french.sty is not used there (and should be used) to allow people to typeset this text without it.

Un texte symétrique

Un extrait de Westcar



References

[1] Jan Buurman, Nicolas Grimal, Michael Hainsworth, Jochen Hallof, and Dirk Van Der Plas. *Inventaire des signes hieroglyphiques en vue de leur saisie informatique.* Mémoires de l'Académie des Inscriptions et Belles Lettres. Institut de France, Paris, 1988.