A LaTeXperiment of hieroglyphic typesetting

Serge Rosmorduc
rosmord@lifac.ens-cachan.fr

May 15, 1998

1 License

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

If you use LaTeX, and want to typeset hieroglyphic texts, this package is for you. If you want to typeset hieroglyphs, and don't know LaTeX, then have a look at the file `EGypto.tex`, to get an idea of what LaTeX looks like.

You can use also use the (ugly, but getting better) fonts 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

As LaTeX works on so many machines, I can't really give you detailed informations. So I assume you know a little about it. 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, these files should be on your `TEXINPUTS` path. The directory where your own LaTeX sources are should do in most cases, anyway.

Then, put your fonts where they can be used. On Unix systems, you will need to put the `.tfm` files in one of the directories listed in the `TEXFONTS` environment variable, and the `pk` files in one of the directories listed in you `PKFONTS` environment variable. Or `TEXPKS`. Or something else. It depends on your printing program.

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). To do this you need to have the metafont program installed. Actually, if you have the GNU font utilities, I distribute all files needed to redraw the fonts.

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.

4 Typesetting hieroglyphs

I provide two ways of typesetting hieroglyphs. The first one only uses the power of LaTeX, and the second use a C program as a pre-processor.

4.1 The hard way

If you want to use only LaTeX, and nothing else, you must include the option `hierLtx` in your `\documentstyle` line, like this:

```
\documentstyle[hierLtx]{article}
```

To type a hieroglyphic text directly, you can use the `\hieroglyphe` macro. You type the signs like this:

```
\hieroglyphe{A/1}
```

to get:

Beware! the numbers aren't exactly the one in Gardiner's grammar. To find which number gives you which sign, you may use the `testfont` TEX source, like this:

let's suppose you want the sign list for "man". You type (on a unix box, but the idea is the same elsewhere. Your input is *slanted*, the computer's output is in typewriter font) :

tex testfont
```
This is TeX--XeT, Version 3.1415--1.0 (C version 6.1)
TeX--XeT Copyright (C) 1992 by Dante e.V.
(/usr/TeX/lib/texmf/tex/plain/testfont.tex
Name of the font to test = A
Now type a test command (\help for help):)
```
\table

\bye
```
[1]
Output written on testfont.dvi (1 page, 6088 bytes).
Transcript written on testfont.log.
```

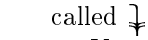
Then, you can print the table.

To type a complicated text, you can , like in the "manuel", type : '-' to separate two groups horizontally, and ':' to separate two groups vertically. Accolades allow you to group signs.

```
\hieroglyphe{{Q/3-X/1}:N/1}
```

nice, isn't it[1]?

4.2 The "Easy" Way

If you have a C compiler, you can build[2] the `sesh` program. I intended it to be called  , but it's a little hard to type on a keyboard.

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).

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.

[1] Actually, I'm cheating. See the file `Ltxex.tex` for real example.

[2] see the file `BUILD.txt`

example: if your LaTeX file, called `foo.htx`, is :

```
\documentstyle[hiero]{article}
\begin{document}
\begin{hieroglyph}
A1 \end{hieroglyph}
\end{document}
```

you should proceed like this :

sesh ¡ foo.htx ¿ foo.tex
latex foo.tex

and you should get a dvi file containing the sign . Beware that you can't use LaTeX comments between the beginning and the end of the `hieroglyph` environment.

4.2.1 Hieroglyphs

The "alphabetic" signs can be called like shown in table 1. The Letters A, i, a, w, b, p, f, m, n, r, h, H, x, X, z, s, S, q, k, g, t, T, d, and D are used both for the hieroglyphs and their transliteration. Signs are named, either after their transliterations (for example, `xpr` is) or after their Gardiner Code. See appendix B for a list of codes. For a complete list, see [**?**], which intends to be *the* format for encoding hieroglyphic texts.

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

4.2.2 Special signs

You won't type only hieroglyphs. So:

.. is a space

. is a half-space

// is

h/ is

v/ is

/ is

o is

O is

SIGN	TRANS-LITTERATION	CODE
	ꜣ	A
	i	i
	ꜥ	a
	w	w
	w	W
	b	b
	p	p
	f	f
	m	m
	n	n
	n	N
	r	r
	h	h
	ḥ	H
	ḫ	x
	ḫ	X
	z	z
	s	s
	š	S
	ḳ	q
	k	k
	g	g
	t	t
	ṯ	T
	d	d
	ḏ	D

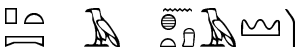
Table 1: Alphabetical signs

4.2.3 Grouping Signs

You can type adjacent signs with '-' , stack signs with ':', group signs with parenthesis, and use '*' to separate signs on the same level.

A few examples:

```
\begin{hieroglyph}
p*t:pt-..-A-..-n:(x:t)*U30-A-xAst-qmA
.end{hieroglyph}
```

gives 

you can type a '=' sign *after* a '-' or a ':' to indicate a grammatical ending. More important, a *space, or any number of spaces, tabulations, or newlines,* indicate a word ending[3] 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 and easier to read[4].

4.2.4 Sign Modifiers

\\ after a signs inverts it: `A1-A1\` writes

\s*n* where *n* is 1,2,3 or 4, diminishes the sign size:

```
A1-A1\s1-A1\s2-A1\s3-A1\s4
```

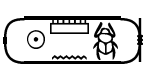
draws

\R*n* rotates a figure by the angle *n*. THIS IS NOT FINISHED and uses postscript. `anx\R30`

4.2.5 Cartouches and parenthesis

`<- and ->` draw a cartouche around the embedded text :

```
<-ra-mn:n-xpr->
```



`<S and >` draw a serekh :

[3] Note that in the "manuel", a word ending is *one* space. This is extremely inconvenient for LaTeX, and, in fact, for the general purpose of directly typing the text.

[4] You can use spaces directly to separate words, if you want, without using a '-'. but this won't be conform to the manuel encoding.

`G5 -<S-nTr-X:t->`

`<H >` draw a *ḥwt*-sign.

`<H-pA-i-i-A1-Htp:t*p>`

The letters *b, m and e,* used after the beginning sign, allow you to draw only the beginning, middle, or ending of construction. So, for example, `<Se-nTr->` writes .

The parenthesis are:

[[...]] for [text] ;

[{...}] for {text } ;

[&...&] for ⟨text ⟩ .

4.2.6 Shading and stacking

`-#-` ...`-#-` shades a text[5]: `-#-A1-A2-#-` writes .

You can shade a part of a group with `#//`, `#h/`, `#v/`, and `#/` : `A1#//` is ; `A1#h/` is ; `A1#v/` is , and `A1#/` is .

`#` alone allows to stack signs: `m#a` is .

4.2.7 Text Direction

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*).

BEWARE !!!! this file was not compiled with TeX–Xet, so the example bellow is false. You should have gotten a dvi version of this file (called `doc.dvi`), without this message.

Example: , made like that:

[5] The previous versions of sesh allowed one to do without the '-'. but I decided to revert to the manual, to be able to use the # to stack signs.

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

There is a more general right-to-left builtin, but it won't allow line-cutting: '+rl' and '+lr'. example :

```
+rl G5 -nTr-ms-w-t:Z2 +rl anx G5 -nTr-ms-w-t:Z2
```

. It will work in any case.

Finally, if your ultimate output is postscript, you can use packages like pstricks to invert left-to-right text and make it into right-to-left.

For Column-writing, the LaTeX macro `\EnColonne` can be used. Its first argument is the column width. The one given in the following example isn't bad.

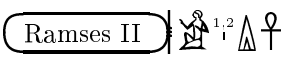
```
\EnColonne[1.2\Htm]{
\begin{hieroglyph}
#def wAt N31# sw*(t:di) -Htp -wp-wAt-wAt-wAt-!
E16-nb-tA:idb\s2*Z1-Dsr-r-xAst-!
\end{hieroglyph}
}
```

It is often necessary to use the `\s` construct to tune the sizes of the signs in column writing.

4.2.8 Some text

`+l .. +s` allows inserting of LaTeX text; `|....-` allows the insertion of superscript text:

```
<-+l Ramses II +s-> -A1-|1,2-di-anx
```



4.2.9 kerning

A non-manual feature of the present system allows to kern the signs. the sequence `\!` is a negative space that can be used both in horizontal context:

```
zA-\!\!ra:.
```

and in vertical one :

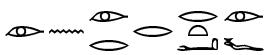
```
nD-D:\!\!Hr\s1
```

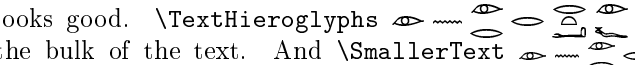
4.2.10 Size

The size of signs is controled by various LaTeX macros:

- `\DisplayHieroglyphs` write the hieroglyphic texts in its scope in a big size, with a large spacing between lines so that the reading is easy.
- `\TextHieroglyphs` does the contrary. It is better for in-text text.
- `\EnPetit{}` writes its argument in the same format as `\TextHieroglyphs` does.
- `\SmallerText` lowers the size of the signs.

`\DisplayHieroglyphs` This is an example :



looks good. `\TextHieroglyphs`  is better if you are in the bulk of the text. And `\SmallerText` is smaller and `\SmallerText` is even smaller

4.2.11 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 macros 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 baseline, 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 transliterations 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 transliteration 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 Transliteration

There are currently three ways to write some transliteration (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 useful for cross-references

   you can type a transliteration, 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

   iw.i 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}
         {example number}{page number in document}
```

(all on the same line)

3. the `\traduction{}{}` macro. It takes two arguments, the first being a transliteration, the second a translation. you have separates footnotes and, by default, the text is given in two columns which can spread over pages, which is useful to translate poetry. If you want the translation under the transliteration, 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: $^{10}|$.

\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. The first argument may contain a `hieroglyph` 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 transliteration (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{this is not\\to \\be taken seriously\\}
 \end{gramrule}
```

$$\textit{ir} + \begin{cases} \textit{infinitive} \\ \textit{prospective } \underline{\textit{s}}\textit{dm.f} \\ \textit{mrr.f} \end{cases} + \begin{array}{c} \textit{this is not} \\ \textit{to} \\ \textit{be taken seriously} \end{array}$$

6 To Do

- Capital letters for transliteration (currently, it is quite hard to type "*R*ʿ".
- A WYSYWIG interface, first using X.
- Improve the signs (has begun)
- fully and strictly implement the "manuel de codage". (but waiting for a possible new version.)

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.)

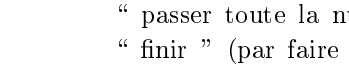
- Added hieroglyphs in `\dico....`;
- Added a possibility of reversed text in normal latex.

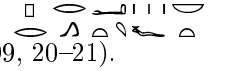
A Examples

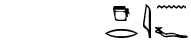
Here are a few examples, taken from various texts : Lefèbvre's *Grammaire de l'Égyptien Classique,* Gardiner's *Egyptian Grammar,* and Westcar.

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 ḥḳrw 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)* la fin il s'assit (*Leb.* 75).

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

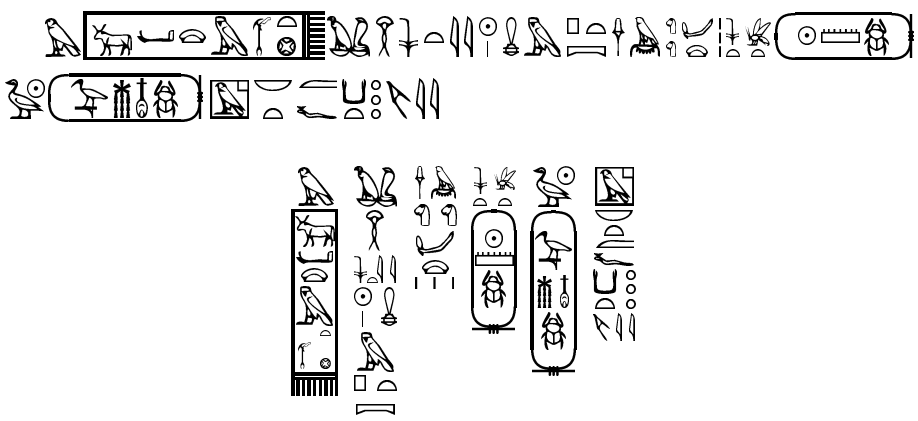
2. Il peut qualifier un substantif ...

EXERCICE XXII

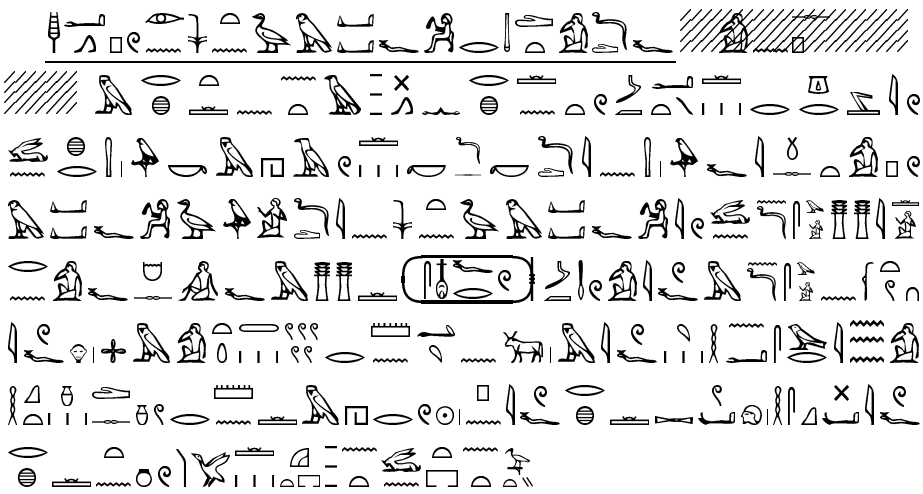
(*a*) *Translate into English :*

(1) (2)

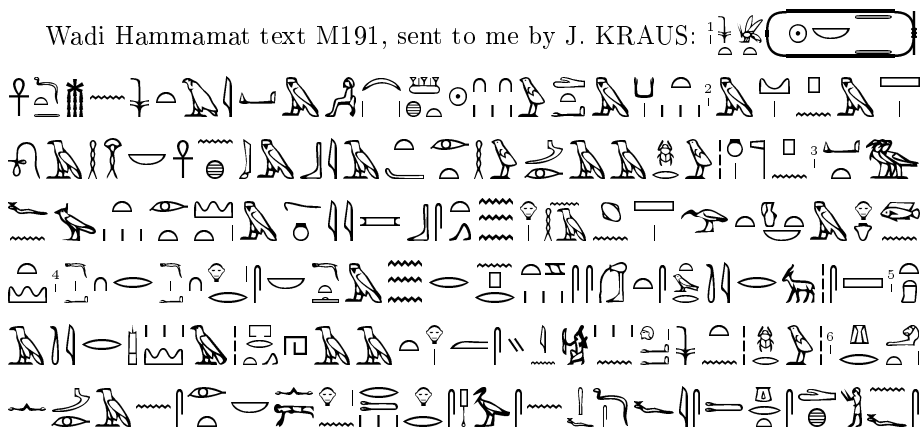
Un texte symétrique



Un extrait de Westcar



Wadi Hammamat text M191, sent to me by J. KRAUS:



B Table of signs

C Thanks

I want to thanks a few persons, whose help definitly helped improving this package. M. Louet and Corler, students in computer science at Brest University, wrote a X program for editing the fonts, which proved very useful for touching a few signs that were really ugly before. Juergen Kraus, egyptologist at Mainz university, has provided me with numerous bug reports and suggestions.