# Distribution ArabicTEX

Yannis Haralambous*

20 avril 1995

*Cette distribution est dédiée*
*à ceux qui luttent contre le*
*fanatisme réligieux et le rascisme.*

## 1 ArabicTEX et Ω

*ArabicTEX n'est qu'un avant-goût de la distribution arabe Ω qui est en préparation. En effet, les polices utilisées consistent en les 256 premiers codes des polices Ω (16-bits). Les codes restants serviront pour les autres langues utilisant l'alphabet arabe (ArabicTEX ne couvre que l'arabe, le persan et l'ourdou). De plus, toute l'analyse contextuelle sera effectuée au sein d'Ω et on n'aura plus besoin de préprocesseur.*

*Un module de compatibilité « ArabicTEX → Ω arabe » assurera la compilation correcte des fichiers préparés pour ArabiTEX par Ω.*

## 2 Présentation générale

La composition de l'arabe présente deux problèmes fondamentaux : (a) le sens de l'écriture, (b) les propriétés contextuelles des caractères. Pour résoudre le premier problème, qui devient encore plus complexe lorsqu'on mélange des textes droite→gauche avec des texte gauche→droite, on se sert de TEX--XET, l'extension bidirectionnelle de TEX. Le deuxième problème est résolu à l'aide du préprocesseur ArabicTEX (`ARABITEX.EXE` sous DOS), qui lira le texte avant TEX--XET et le modifiera.

Voici donc le schéma de travail :

$\boxed{\text{fichier } \texttt{.tex}} \to \boxed{\text{fichier } \texttt{.tmp}} \to \boxed{\text{fichier } \texttt{.dvi}}$

où la première flèche représente ArabicTEX, et la deuxième, TEX--XET. Les extensions des fichiers sont laissés au choix de l'utilisateur. Il est utile de garder les fichiers avant conversion, puisque le texte arabe y est plus lisible.

*Adresse personnelle : 187, rue Nationale, 59800 Lille, Fax : (33) 20.40.28.64, Internet `haralambous@univ-lille1.fr`

# 3 Conventions de saisie

Pour saisir le texte arabe, on utilise une transcription. D'autres filtres peuvent être écrits pour passer des codages arabes (ASMO et autres) à notre transcription ; on peut même envisager de modifier le source de ArabicTEX si cela peut être utile : il s'agit d'un simple fichier Lex.

Voici la transcription proposée pour l'arabe, le persan et l'ourdou[1] :

## 3.1 Consonnes

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ا | `A` | ٹ | `'t` | ر | `r` | ط | `T` | گ | `g` | و | `U` |
| أ | `'a` | ج | `j` | ز | `z` | ظ | `Z` | ل | `l` | ؤ | `'u` |
| إ | `'i` | ح | `H` | ژ | `zh` | ع | `` ` `` | م | `m` | ى | `I` |
| آ | `'A` | خ | `kh` | ڑ | `'r` | غ | `gh` | ن | `n` | ي | `y` |
| ب | `b` | چ | `ch` | س | `s` | ف | `f` | ں | `'n` | ئ | `'y` |
| ت | `t` | د | `d` | ش | `sh` | ق | `q` | ه | `h` | ء | `\|\|` |
| ث | `th` | ذ | `dh` | ص | `S` | ڤ | `v` | ة | `"h` | ے | `E` |
| پ | `p` | ڈ | `'d` | ض | `D` | ك | `k` | ۀ | `e` | | |

Notes :

- le alif avec wesla ٱ se transcrit `"A` ;
- la lettre wa و se transcrit *aussi* par `w` : l'utilisateur a le choix d'utiliser '`U`' ou '`w`' pour transcrire cette lettre ;
- de même, la lettre ta marbouta ة se transcrit *aussi* par `"t` : l'utilisateur a le choix d'utiliser '`"h`' ou '`"t`' pour transcrire cette lettre ;
- le tiret devant `h` sert uniquement à distinguer entre des paires de lettres et des transcriptions à deux lettres : par exemple, كه (transcrit '`k-h`') et خ (transcrit '`kh`'). Il peut être placé devant n'importe quelle consonne et ne changera pas la sortie.

## 3.2 Double consonnes

Pour produire un shadda, il suffit d'écrire deux fois la consonne. Ainsi, `Damma"h` produira ضَمَّة. Pour avoir deux lettres identiques séparées on utilisera le tiret : تتحرّك est obtenu par `t-tHrrk`.

[1] Voir en 3.5 une convention spéciale, appliquée en mode ourdou.

Cette précaution est inutile lorsqu'on voyellise : `tataHrrak` donnera bien تَتَحرَّك puisque la voyelle '`a`' se place entre les deux lettres `t`.

### 3.3 Voyelles

Voici la transcription des voyelles :

| | |
|---|---|
| fatha | `a` |
| kasra | `i` |
| damma | `u` |
| soukoun | `<>` |
| fatha vertical | `a\|` |
| alif avec wesla | `"A` |
| alif avec madda | `'A` |
| fathatan | `aN` |
| kasratan | `iN` |
| dammatan | `uN` |

### 3.4 Exemples

Texte non voyellisé :

*حرف مطبعى*

قطعة معدنية يعلوها وجه حرف طباعى بارز، تستخدم في طباعة الحروف « التيبو » للطبعمنها مباشرة. قد تؤخذ عنها قوالب معدنية أو مطاطية أو بلاستيك، الخ، أو قد تنقل عنها أشكال أخرى للاستفادة منها في عمليلت مثل الطباعة الليثوغرافية أو الطباعة من سطح غائر.

obtenu par

```
\beginR \emph{Hrf mTb`I}\hfill\phantom{.}

\beginR qT`"t m`dnI"t I`lUhA wjh Hrf TbA`I bArz, tstkhdm fy
TbA`"t AlHrUf < AltybU > l-lTb`mnhA mbAshr"t. qd t'ukhdh
`nhA qUAlb m`dny"t 'aU mTATy"t 'aU blAstIk, Alkh, 'aU qd
tnql `nhA 'ashkAl 'akhrI l-lAstfAd"t mnhA fy `mlIlt mthl
AlTbA`"t Al-lythUghrAfy"h 'aU AlTbA`"t mn sTH ghA'yr.
\hfill\phantom{.}
```

Texte voyellisé :

*أَلْأَرْنَبُ وَٱلفِيلُ*

زَعَمُوا أَنَّ أَرْضاً مِنْ أَرْضِ الفِيَلَة، تَتَابَعَتْ عَلَيْهَا السُنُونَ وَأَجْدَبَت، فَقَلَّ الماءُ في تِلْكَ البِلَادِ وَغَارَتِ العُيُونُ، وَأَصَابَ الفِيَلَةَ عَطَشٌ شَدِيدٌ.

obtenu par

```
\beginR \emph{'aal<>'aar<>nabu wa"Alfiylu}\hfill\phantom{.}

\beginR za`amuwA 'aanna 'aar<>DAaN min<> 'aar<>Di Alfiyala"ti,
```

```
tataAba‘at<> ‘alay<>haA AlsunUna wa’aaj<>dabat, faqalla AlmA||u
fy til<>ka AlbilaAdi waghaArati Al‘uyUnu, wa’aaSaAba Alfiyala"ta
‘aTashuN shadiyduN.\hfill\phantom{.}\endR
```

### 3.5 Conventions spéciales pour l'ourdou

L'ourdou différencie les deux écritures de la lettre ha : ـہـ et ـھـ. Au lieu de chercher une astuce de transcription pour différencier ces deux lettres, ainsi que la lettre ha arabe habituelle, nous avons choisi de définir un « mode ourdou »[2] (cf. 4.1 pour savoir comment entrer en mode ourdou), dans lequel deux nouvelles conventions de saisie s'ajoutent :

1. la saisie `x` produit la véritable lettre ha ourdou, dont les quatre formes sont ہ ـہ ـہـ ہـ ;
2. la saisie `h` produit la deuxième partie de certains digraphes, par exemple `jh` produit جھ. Pour éviter des ligatures, on peut de nouveau utiliser le tiret : ainsi on différencie خ (`kh`) et کھ (`k-h`).

## 4 Contrôle du préprocesseur

Pour indiquer au préprocesseur le début et la fin du texte à processer, on peut procéder de deux manières : à l'aide de directives, ou à l'aide d'un caractère d'échappement.

### 4.1 Directives

Les directives sont toujours écrites en capitales et commencent par un "hash" `#` en début de ligne.

Pour entrer en mode arabe on utilise la directive

```
#ARABIC
```

Pour sortir du mode arabe on utilise

```
#LATIN
```

Pour passer en mode ourdou si on est déjà en mode arabe, et pour que la directive `#ARABIC` soit par la suite considérée comme un passage en mode ourdou, on utilisera

```
#URDU_MODE
```

n'importe où dans le document. De même, suite à

```
#ARABIC_MODE
```

passera en mode arabe pur (si on est déjà en mode arabe), et `#ARABIC` signifiera par la suite « mode arabe pur ».

[2] Des modes pashto et sindhi sont en préparation.

### 4.2 Caractère(s) d'échappement

Pour faciliter le passage en mode arabe, on peut utiliser un caractère d'échappement. Celui-ci peut être choisi par l'utilisateur, et modifié à tout instant. La directive qui permet de le choisir est

```
#DEFINE ARABIC *
```

où * est remplacé par le caractère d'échappement choisi. On a intérêt à choisir des caractères 8-bit, pour être sûr que ceux-ci n'apparaitront pas ailleurs dans le document. À noter également que le préprocesseur traite aussi les commentaires TeX, au même titre que le texte.

### 4.3 Traitement spécial des *keshideh*

Les *keshideh* sont les liaisons entre les caractères. À défaut de coupure des mots, ce sont ces traits qui permettent une justification du texte en évitant les blancs excessifs. Les *keshideh* sont introduits automatiquement par ArabicTeX, mais il arrive que l'utilisateur veuille les éviter. Pour cela on dispose de la directive

```
#ARABIC_NO_KESHIDEH
```

On peut aussi définir un autre caractère d'échappement, qui servira à délimiter le texte arabe sans *keshideh*, cela se fait à l'aide de la directive

```
#DEFINE ARABIC_NO_KESHIDEH *
```

### 4.4 Exemple

Choisissons • comme caractère d'échappement (avec *keshideh*) et ¶ comme tel sans *keshideh*. Alors

```
Après •ghyr•, •kll•... le nom se
met au génitif dont la marque au pluriel est ¶yn¶.
```

donnera

> Après غـــيـــر, كـــلّ... le nom se met au génitif dont la marque au pluriel est ين.

## 5 Polices

Les polices sont disponibles sont : `ar` (police arabe moderne), `arb` (police arabe grasse), `aro` (police arabe moderne en contours). La dernière est préconisée pour remplacer l'italique.

Exemples :

أَهـــلاً وَسَـــهـــلاً (police `ar`)

أَهـــلاً وَسَـــهـــلاً (police `aro`)

أهـــــلاً وَ سَـــــهـــــلاً (police `arb`)

Des polices PostScript commerciales ou du domaine public peuvent facilement être adaptées et utilisées avec ArabiTeX, en passant par des polices virtuelles.

## 6 Macros LaTeX nécessaires

Le point faible de TeX--XeT est l'utilisation de ses primitives : en effet, il faut (a) re-écrire la primitive `\beginR` en début de chaque nouveau paragraphe de texte arabe, (b) annuler le retrait de début de paragraphe, parce que TeX--XeT le place à gauche, quoique l'on fasse, et (c) ajouter un `\hfill\phantom{.}` à la fin de chaque paragraphe (utilisant des *keshideh*), sinon TeX essaie d'égaliser le blanc de fin de paragraphe avec les *keshideh* du texte.

Ces problèmes ne se présentent pas lorsqu'on n'écrit que des courts extraits arabes. Par contre ils sont omniprésents dans le cas d'un livre entièrement en arabe. On essaiera de résoudre ces problèmes dans Ω.

On peut donc définir des macros `\arab{}` et `\arabpar{}` comme suit :

```
\newcommand{\arab}[1]{{\beginR\fontfamily{ar}\selectfont#1\endR}}
\newcommand{\arabpar}[1]{{\beginR\fontfamily{ar}\selectfont#1%
\hfill\phantom{.}\endR}}
```

en sachant qu'il faut utiliser `\arabpar{}` pour chaque paragraphe de texte arabe...

Les chiffres sont gérés par ArabicTeX et les mathématiques par TeX--XeT, il n'y a donc pas d'autre précaution à prendre.

Dans le préambule d'un fichier comportant du texte arabe avec *keshideh* il faut définir la commande de *keshideh* `\kesh`, de la manière suivante :

```
\newcommand{\kesh}{\nobreak\xleaders\hbox{\char'367}\hfil\nobreak}
```

## 7 Statut et installation

La distribution ArabicTeX appartient entièrement au domaine public[3]. Elle peut être récupérée par ftp anonyme du serveur
`ftp.ens.fr:/pub/tex/yannis/arabictex`

### 7.1 Installation

On trouvera sur serveur des exécutables DOS et Macintosh (MPW, DirectTeX et autonome). Pour re-compiler ArabicTeX sur ces plate-formes, ou pour le compiler sous Unix, il faut :

1. Lancer Flex (version 2.4.7 ou ultérieure) sur `arabitex.lex`, avec la ligne de commande suivante :

[3] Pour être plus exact elle a le statut de "postcardware" : il est suggeré aux utilisateurs satisfaits d'envoyer une carte postale de l'endroit où ils habitent à l'auteur. Les utilisateurs insatisfaits ou neutres en sont dispensés.

```
flex -8 -t arabitex.lex > arabitex.c
```

2. Compiler le fichier `arabitex.c` ainsi produit avec votre compilateur C préféré (celui de l'auteur étant gcc). Sous UNIX la ligne de commande pourrait être :

```
gcc -o arabitex arabitex.c
```