

The arabluatex package v1.4.1 – 2016/07/09

Robert Alessi
alessi@robertalessi.net

Contents

License and disclaimer	2	4.7 Digits	21
1 Introduction	2	4.7.1 Numerical figures	21
1.1 arabluatex is for Lua [®] TeX	4	4.7.2 The <i>abjad</i>	21
2 The basics of arabluatex	4	4.8 Additional characters	21
2.1 Activating arabluatex	4	4.9 Arabic emphasis	22
2.2 Options	5	5 Special applications	22
2.2.1 Classic contrasted with modern typesetting of Arabic	5	6 Transliteration	22
2.3 Typing Arabic	6	6.1 Additional note on <code>dmg</code> convention	24
2.3.1 Local options	7	6.2 Examples	25
3 Standard ArabTeX input	7	7 Buckwalter input scheme	26
3.1 Consonants	7	8 L[®]TeX Commands in Arabic environments	28
3.2 Vowels	8	8.1 <code>csquotes</code>	31
3.2.1 Long vowels	8	8.2 <code>reledmac</code>	31
3.2.2 Short vowels	9	9 Future work	31
4 arabluatex in action	10	10 Implementation	32
4.1 The vowels and diphthongs	10	References	37
4.2 Other orthographic signs	12	Change History	37
4.3 Special orthographies	16	Index	37
4.4 Quoting	18		
4.4.1 Quoting the <i>hamza</i>	19		
4.5 The “pipe” character (<code> </code>)	20		
4.6 Stretching characters: the <i>taṭwīl</i>	20		

List of Tables

1 ArabTeX consonants	8	4 “Quoted” <i>hamza</i>	19
2 ArabTeX long vowels	9	5 Additional Arabic codings	21
3 ArabTeX short vowels	9	6 Buckwalter scheme	26

Abstract

This package provides for Lua^AT_EX an ArabT_EX-like interface to generate Arabic writing from an ASCII transliteration. It is particularly well-suited for complex documents such as technical documents or critical editions where a lot of left-to-right commands intertwine with Arabic writing. `arabluatex` is able to process any ArabT_EX input notation. Its output can be set in the same modes of vocalization as ArabT_EX, or in different roman transliterations. It further allows many typographical refinements. It will eventually interact with some other packages yet to come to produce from `.tex` source files, in addition to printed books, TEI `xml` compliant critical editions and/or lexicons that can be searched, analyzed and correlated in various ways.

License and disclaimer

Copyright © 2016 Robert Alessi

Please send error reports and suggestions for improvements to Robert Alessi:

- email: <alessi@robertalessi.net>
- website: <http://www.robertalessi.net/arabluatex>
- development: <http://git.robertalessi.net/arabluatex>
- comments, feature requests, bug reports: <http://issues.robertalessi.net>

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 3 of the License, 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, see <<http://www.gnu.org/licenses/>>.

This release of `arabluatex` consists of the following source files:

- `arabluatex.ins`
- `arabluatex.dtx`
- `arabluatex.lua`
- `arabluatex_voc.lua`
- `arabluatex_fullvoc.lua`
- `arabluatex_novoc.lua`
- `arabluatex_trans.lua`

1 Introduction

In comparison to Prof. Lagally’s outstanding ArabT_EX,¹ ArabLuaT_EX is at present nothing more than a modest piece of software. Hopefully—if I may say so—it will eventually provide all of its valuable qualities to the Lua^AT_EX users.

¹See <http://ctan.org/pkg/arabtex>

arabtex dates back to 1992. As far as I know, it was then the first and only way to typeset Arabic texts with \TeX and \LaTeX . To achieve that, arabtex provided—and still does—an Arabic font in *Nashī* style and a macro package that defined its own input notation which was, as the author stated, “both machine, and human, readable, and suited for electronic transmission and e-mail communication”.² Even if the same can be said about Unicode, Arab \TeX ASCII input notation still surpasses Unicode input, in my opinion, when it comes to typesetting complex documents, such as scientific documents or critical editions where footnotes and other kind of annotations can be particularly abundant. It must also be said that most text editors have trouble in displaying Arabic script connected with preceding or following \LaTeX commands: it often happens that commands seem misplaced, not to mention punctuation marks, or opening or closing braces, brackets or parentheses that are unexpectedly displayed in the wrong direction. Of course, some text editors provide ways to get around such difficulties by inserting invisible Unicode characters, such as LEFT-TO-RIGHT or RIGHT-TO-LEFT MARKS (U+200E, U+200F), RTL/LTR “embed” characters (U+202B, U+202A) and RLO/LRO “bidi-override” characters (U+202E, U+202D).³ Nonetheless, it remains that inserting all the time these invisible characters in complex documents rapidly becomes confusing and cumbersome.

The great advantage of Arab \TeX notation is that it is immune from all these difficulties, let alone its being clear and straightforward. One also must remember that computers are designed to process code. Arab \TeX notation is a way of encoding Arabic language, just as \TeX “mathematics mode” is a way of processing code to display mathematics. As such, not only does it allow greater control over typographical features, but it also can be processed in several different ways: so without going into details, depending on one’s wishes, Arab \TeX input can be full vocalized Arabic (*scriptio plena*), vocalized Arabic or non-vocalized Arabic (*scriptio defectiva*); it further can be transliterated into whichever romanization standard the user may choose.

But there may be more to be said on that point, as encoding Arabic also naturally encourages the coder to vocalize the texts—without compelling him to do so, of course. Accurate coding may even have other virtuous effects. For instance, hyphens may be used for tying particles or prefixes to words, or to mark inflectional endings, and so forth. In other words, accurate coding produces accurate texts that can stand to close grammatical scrutiny and to complex textual searches as well.

Having that in mind, I started arabluatex. With the help of Lua, it will eventually interact with some other packages yet to come to produce from `.tex` source files, in addition to printed books, TEI xml compliant critical editions and/or lexicons that can be searched, analyzed and correlated in various ways.

²Lagally (2004, p. 2).

³Gáspár Sinai’s Yudit probably has the best Unicode support. See <http://www.yudit.org>.

1.1 arablualatex is for Lua⁴LaTeX

It goes without saying that `arablualatex` requires Lua⁴LaTeX. `TeX` and `LaTeX` have `arabxetex`, and `XYLaTeX` has `arabxetex`. Both of them are much more advanced than `arablualatex`, as they can process a number of different languages,⁴ whereas `arablualatex` can process only Arabic for the time being. More languages will be included in future releases of `arablualatex`.

In comparison to `arabxetex`, `arablualatex` works in a very different way. The former relies on the `TECKit` engine which converts Arab`TeX` input on the fly into Unicode Arabic script, whereas the latter passes Arab`TeX` input on to a set of Lua functions. At first, `LaTeX` commands are taken care of in different ways: some, as `\emph`, `\textbf` and the like are expected to have Arabic text as arguments, while others, as `\LR`, for “left-to-right text”, are not. Then, once what is Arabic is carefully separated from what is not, it is processed by other Lua functions which rely on different sets of correspondence tables to do the actual conversion in accordance with one’s wishes. Finally, Lua returns to `TeX` the converted strings—which may in turn contain some other Arab`TeX` input yet to be processed—for further processing.

2 The basics of arablualatex

2.1 Activating arablualatex

As usual put in your preamble:

```
\usepackage{arablualatex}
```

The only requirement of `arablualatex` is Lua⁴LaTeX; it will complain if you try to compile your document with another engine. That aside, `arablualatex` does not load packages such as `polyglossia` or `luabidi`. It can work with `polyglossia` though, but does not require it.

Font setup If you wish to use your own Arabic font, you can define it before loading `arablualatex`. Assuming that `fontspec` is loaded, put this in your preamble just above the line that loads `arablualatex`:

```
\newfontfamily\arabicfont{<fontname>}[Script=Arabic]
```

where `<fontname>` is the standard name of the Arabic font you wish to use.

By default, if no Arabic font is selected, `arablualatex` will issue a warning message and attempt to load the Amiri font⁵ like so:—

```
\newfontfamily\arabicfont{Amiri}[Script=Arabic]
```

REM. By default Amiri places the *kasra* in combination with the *tašdid* below the consonant, like so: `ك`. That is correct, as at least in the oldest manuscripts `ك` may stand for `ك` as well as `ك`. See Wright (1896, i.14.C–D). The placement of the *kasra* above the consonant may be obtained by

⁴To date, both packages support Arabic, Maghribi, Urdu, Pashto, Sindhi, Kashmiri, Uighuric and Old Malay; in addition to these, `arabxetex` also has a Hebrew mode, including Judeo-Arabic and Yiddish.

⁵Hosny (2015).

selecting the `ss05` feature of the Amiri font, like so:—⁶

```
\newfontfamily\arabicfont{Amiri}[Script=Arabic,RawFeature={+ss05}]
```

Other Arabic fonts may behave differently.

2.2 Options

`arabluatex` may be loaded with four mutually exclusive global options, each of which may be overridden at any point of the document (see below section 2.3.1 on page 7):

`voc`

default

In this mode, which is the one selected by default, every short vowel written generates its corresponding diacritical mark: *damma* (◌ُ), *fatha* (◌َ) and *kasra* (◌ِ). If a vowel is followed by N, viz. $\langle uN, aN, iN \rangle$, then the corresponding *tanwīn* (◌ِ◌◌, ◌◌◌◌, ◌◌◌◌ or ◌◌◌◌) is generated. Finally, $\langle u, a, i \rangle$ at the commencement of a word indicate a “connective *ʿalif*” (*ʿalifu ʿl-waṣli*), but `voc` mode does not show the *waṣla* above the *ʿalif*; instead, the accompanying vowel may be expressed at the beginning of a sentence (◌◌◌◌).

`fullvoc`

In addition to what the `voc` mode does, `fullvoc` expresses the *sukūn* and the *waṣla*.

`novoc`

None of the diacritics is showed in `novoc` mode, unless otherwise specified (see “quoting” technique below section 4.4 on page 18).

`trans`

This mode transliterates the ArabTeX input into one of the accepted standards. At present, two standards are supported (see below section 6 on page 22 for more details):

dmg *Deutsche Morgenländische Gesellschaft*, which is selected by default;

loc *Library of Congress*.

More standards will be included in future releases of `arabluatex`.

2.2.1 Classic contrasted with modern typesetting of Arabic

By default, `arabluatex` typesets Arabic in a classic, traditional style the most prominent features of which are the following:

- ‘Classic’ *madda*: when *ʿalif* and *hamza* accompanied by a simple vowel or *tanwīn* is preceded by an *ʿalif* of prolongation (◌◌◌◌), then a mere *hamza* is written on the line, and a *madda* is placed over the *ʿalif*, like so:—

`samA'uN` سَمَاءُ *samāʿun*, `JA'a` جَاءَ *ǧāʿa*, `yataSA'alUna` يَتَسَاءَلُونَ *yatasāʿalūna*⁷
(see on page 13 for further details).

- The euphonic *tašdīd* is generated (see on page 14).
- Assimilation rules laid on item b on page 14 are applied.
- In `fullvoc` mode, the *sukūn* is expressed.

⁶See the documentation of `amiri`, Hosny (2015, p. 5).

⁷Note that in old mss. such forms as سَمَاءُ, جَاءُ are also found; see Wright (1896, i. 24 D).

`\SetArbEasy` Such refinements may be discarded by the command `\SetArbEasy`, either globally in the preamble or locally at any point of the document. Default ‘classic’ rules may be set back at any point of the document with the command `\SetArbDflt`. Examples follow:—

(a) `\SetArbDflt`:

- i. voc وَمَاتَ اسْتِسْقَاءٌ قَبْلَ أَنْ يُتِمَّ كِتَابَهُ فِي نُجُومِ السَّمَاءِ
- ii. fullvoc وَمَاتَ اسْتِسْقَاءٌ قَبْلَ أَنْ يُتِمَّ كِتَابَهُ فِي نُجُومِ السَّمَاءِ
- iii. trans *wa-māta 'stisqā^{an} qabla 'ay yutimma kitāba-hu fī nuǧūmⁱ 's-samā^{ai}*

(b) `\SetArbEasy`:

- i. voc وَمَاتَ اسْتِسْقَاءٌ قَبْلَ أَنْ يُتِمَّ كِتَابَهُ فِي نُجُومِ السَّمَاءِ
- ii. fullvoc وَمَاتَ اسْتِسْقَاءٌ قَبْلَ أَنْ يُتِمَّ كِتَابَهُ فِي نُجُومِ السَّمَاءِ
- iii. trans *wa-māta 'stisqā^{an} qabla 'an yutimma kitāba-hu fī nuǧūmⁱ 's-samā^{ai}*

Please note that this document is typeset with `\SetArbDflt` throughout.

2.3 Typing Arabic

`\arb` Once `arabluatex` is loaded, a command `\arb{<Arabic text>}` is available for inserting Arabic text in paragraphs, like so:—

```
1 From \textcite[i. 1 A]{Wright}:--- Arabic, like Hebrew and
2 Syriac, is written and read from right to left. The letters
3 of the alphabet (\arb{hurUf-u 'l-hijA'-i}, \arb{hurUf-u
4 'l-tahajjI}, \arb{al-.hurUf-u 'l-hijA'iyyaT-u}, or
5 \arb{hurUf-u 'l-mu`jam-i}) are twenty-eight in number and
6 are all consonants, though three of them are also used as
7 vowels (see §3).
```

From Wright (1896, i. 1 A):— Arabic, like Hebrew and Syriac, is written and read from right to left. The letters of the alphabet (حُرُوفُ الْحِجَاءِ, حُرُوفُ الْحُرُوفِ الْحِجَاءِ, or حُرُوفُ الْمُعْجَمِ) are twenty-eight in number and are all consonants, though three of them are also used as vowels (see § 3).

`arab` Running paragraphs of Arabic text should rather be placed inside an *Arabic environment*

```
1 \begin{arab}
2 [...]
3 \end{arab}
```

like so:—

```

1 \begin{arab}
2   'at_A .sadIquN 'il_A ju.hA ya.tlubu min-hu .himAra-hu
3   li-yarkaba-hu fI safraTiN qa.sIraTiN fa-qAla la-hu:
4   \enquote{sawfa 'u`Idu-hu 'ilay-ka fI 'l-masA'-i
5   wa-'adfa`u la-ka 'ujraTaN.} fa-qAla ju.hA:
6   \enquote{'anA 'AsifuN jiddaN 'annI lA 'asta.tI`u 'an
7   'u.haqqiqa la-ka ra.gbabata-ka fa-'l-.himAr-u laysa hunA
8   'l-yawm-a.} wa-qabla 'an yutimma ju.hA kalAma-hu bada'a
9   'l-.himAr-u yanhaqu fI 'i.s.tabli-hi. fa-qAla la-hu
10  .sadIqu-hu: \enquote{'innI 'asma`u .himAra-ka yA ju.hA
11  yanhaqu.} fa-qAla la-hu ju.hA: \enquote{.garIbuN
12  'amru-ka yA .sadIqI 'a-tu.saddiqu 'l-.himAr-a
13  wa-tuka_d_diba-nI?}
14 \end{arab}

```

أَتَى صَدِيقٌ إِلَى جَحَّا يَطْلُبُ مِنْهُ حِمَارَهُ لِيَرْكَبَهُ فِي سَفَرَةٍ قَصِيرَةٍ فَقَالَ لَهُ: "سَوْفَ أُعِيدُهُ إِلَيْكَ فِي الْمَسَاءِ
وَأَدْفَعُ لَكَ أُجْرَةً." فَقَالَ جَحَّا: "أَنَا آسَفٌ جِدًّا أَنِّي لَا أَسْتَطِيعُ أَنْ أُحَقِّقَ لَكَ رَغْبَتَكَ فَالْحِمَارُ لَيْسَ هُنَا
الْيَوْمَ." وَقَبْلَ أَنْ يُتِمَّ جَحَّا كَلَامَهُ بَدَأَ الْحِمَارُ يَنْهَقُ فِي إِصْطَبَلِهِ. فَقَالَ لَهُ صَدِيقُهُ: "إِنِّي أَسْمَعُ حِمَارَكَ يَا جَحَّا
يَنْهَقُ." فَقَالَ لَهُ جَحَّا: "غَرِيبُ أَمْرِكَ يَا صَدِيقِي أَتَصَدِّقُ الْحِمَارَ وَتَكْذِبُنِي؟"

2.3.1 Local options

As seen above in section 2.2 on page 5, arabuatex may be loaded with four mutually exclusive global options: `voc` (which is the default option), `fullvoc`, `novoc` and `trans`. Whatever choice has been made globally, it may be overridden at any point of the document, as the `\arb` command may take any of the `voc`, `fullvoc`, `novoc` or `trans` modes as optional arguments, like so:—

<code>voc</code>	– <code>\arb[voc]{\langle Arabic text \rangle};</code>
<code>fullvoc</code>	– <code>\arb[fullvoc]{\langle Arabic text \rangle};</code>
<code>novoc</code>	– <code>\arb[novoc]{\langle Arabic text \rangle};</code>
<code>trans</code>	– <code>\arb[trans]{\langle Arabic text \rangle}.</code>

The same optional arguments may be passed to the environment `arab`: one may have `\begin{arab}[\langle mode \rangle] ... \end{arab}`, where `\langle mode \rangle` may be any of `voc`, `fullvoc`, `novoc` or `trans`.

3 Standard ArabTeX input

3.1 Consonants

Table 1 gives the ArabTeX equivalents for all of the Arabic consonants.

Letter	Transliteration ⁸		ArabT _E X notation
	dmg	loc	
ا ⁹	<i>a</i>	<i>a</i>	a
ب	<i>b</i>	<i>b</i>	b
ت	<i>t</i>	<i>t</i>	t
ث	<i>ṭ</i>	<i>th</i>	_t
ج	<i>ǧ</i>	<i>j</i>	^g or j
ح	<i>ḥ</i>	<i>ḥ</i>	.h
خ	<i>ḫ</i>	<i>kh</i>	_h or x
د	<i>d</i>	<i>d</i>	d
ذ	<i>ḍ</i>	<i>dh</i>	_d
ر	<i>r</i>	<i>r</i>	r
ز	<i>z</i>	<i>z</i>	z
س	<i>s</i>	<i>s</i>	s
ش	<i>š</i>	<i>sh</i>	^s
ص	<i>ṣ</i>	<i>ṣ</i>	.s
ض	<i>ḍ</i>	<i>ḍ</i>	.d
ط	<i>ṭ</i>	<i>ṭ</i>	.t
ظ	<i>ẓ</i>	<i>ẓ</i>	.z
ع	<i>c</i>	<i>‘</i>	`
غ	<i>ǧ</i>	<i>gh</i>	.g
ف	<i>f</i>	<i>f</i>	f
ق	<i>q</i>	<i>q</i>	q
ك	<i>k</i>	<i>k</i>	k
ل	<i>l</i>	<i>l</i>	l
م	<i>m</i>	<i>m</i>	m
ن	<i>n</i>	<i>n</i>	n
ه	<i>h</i>	<i>h</i>	h
و	<i>w</i>	<i>w</i>	w
ي	<i>y</i>	<i>y</i>	y
ة	<i>a</i>	<i>ah</i>	T

Table 1: Standard ArabT_EX (consonants)

3.2 Vowels

3.2.1 Long vowels

Table 2 gives the ArabT_EX equivalents for the Arabic long vowels.

⁸See below section 6 on page 22.

⁹For *ʾalif* as a consonant, see Wright (1896, i. 16 D).

Letter	Transliteration ¹⁰		ArabTeX notation
	dmg	loc	
ا	\bar{a}	\bar{a}	A
و	\bar{u}	\bar{u}	U
ي	\bar{i}	\bar{i}	I
آ ¹¹	\bar{a}	\acute{a}	_A or Y
أ	\bar{a}	\bar{a}	_a
و	\bar{u}	\bar{u}	_u
ي	\bar{i}	\bar{i}	_i

Table 2: Standard ArabTeX (long vowels)

REM. *a.* The long vowels \bar{a} , \bar{u} , \bar{i} , otherwise called *hurūf^u 'l-maddⁱ*, the letters of prolongation, involve the placing of the short vowels *a*, *u*, *i* before the letters ا, و, ي respectively. `arabluatex` does that automatically in case any from `voc`, `fullvoc` or `trans` modes is selected e.g. قَالَ *qāla*, قِيلَ *qīla*, يَقُولُ *yaqūlu*.

REM. *b.* Defective writings, such as ا, *al-ʿalif^u 'l-mahḍūfat^u*, or defective writings of \bar{u} and \bar{i} are encoded _a _u and _i respectively, e.g. _d_alika ذَلِكَ, al-mal_a'ikaT-u 'l-ra.hm_an-u المَلِكُ الرَّحْمَنُ, .hu_dayfaT-u bn-u 'l-yamAn_i اِبْنُ حُدَيْفَةَ for *Huḍayfat^u bn^u 'l-Yamānī*, etc.

3.2.2 Short vowels

Table 3 gives the ArabTeX equivalents for the Arabic short vowels.

Letter	Transliteration ¹²		ArabTeX notation
	dmg	loc	
ا	<i>a</i>	<i>a</i>	a
و	<i>u</i>	<i>u</i>	u
ي	<i>i</i>	<i>i</i>	i
ان	<i>an</i>	<i>an</i>	aN
un	<i>un</i>	<i>un</i>	uN
in	<i>in</i>	<i>in</i>	iN

Table 3: Standard ArabTeX (short vowels)

Whether Arabic texts be vocalized or not is essentially a matter of personal choice. So one may use `voc` mode and decide not to write vowels except at some particular places for disambiguation purposes, or use `novoc` mode, not write vowels—

¹⁰See below section 6 on page 22.

¹¹= *al-ʿalif^u 'l-maqṣūrat^u*.

¹²See below section 6 on page 22.

as `novoc` normally does not show them—except, again, where disambiguation is needed.¹³

However, it may be wise to always write the vowels, leaving to the various modes provided by `arabluatex` to take care of showing or not showing the vowels.

That said, there is no need to write the short vowels *fathā*, *ḍamma* or *kasra* except in the following cases:

- at the commencement of a word, to indicate that a connective *ʿalif* is needed, with the exception of the article (see below section 4.4 on page 18);
- when `arabluatex` needs to perform a contextual analysis to determine the carrier of the *hamza*;
- in the various transliteration modes, as vowels are always expressed in romanized Arabic.

4 arabluatex in action

4.1 The vowels and diphthongs

Short vowels As said above, they are written $\langle a, u, i \rangle$:

_halaqa (or xalaqa) خَلَقَ *halaqa*, ^samsuN شَمْسٌ *šams^{un}*, karImuN كَرِيمٌ *Karīm^{un}*.
 bi-hi بِه *bi-hi*, 'aqi.tuN أَقِطُ *ʿaqiṭ^{un}*.
 la-hu لَهُ *la-hu*, .hujjaTuN حُجَّةٌ *ḥuġġat^{un}*.

Long vowels They are written $\langle U, A, I \rangle$:

qAla قَالَ *qāla*, bI`a بَاعَ *bī`a*, .tUruN طُورٌ *tūr^{un}*, .tInuN طِينٌ *tīn^{un}*,
 murU'aTuN مُرُوَّةٌ *murūʿat^{un}*.

ʿalif maqṣūra It is written $\langle _A \rangle$ or $\langle Y \rangle$:

al-fat_A الْفَتَى *al-fatā*, al-maqh_A الْمَقْهَى *al-maqhā*, 'il_A إِلَى *ʿilā*.

ʿalif otiosum Said *ʿalif^u* *ʿl-wiqāyatⁱ*, “the guarding *ʿalif*”, after و at the end of a word, both when preceded by *ḍamma* and by *fathā* is written $\langle UA \rangle$ or $\langle aW, aWA \rangle$:

na.sarUA نَصَرُوا *naṣarū*, katabUA كَتَبُوا *katabū*, ya.gzUA يَغْزُوا *yaġzū*, ramaw رَمَوْا *ramaw*, banaWA بَنَوْا *banaw*.

¹³See below section 4.4 on page 18.

^o*alif maḥdūfa* and defective \bar{u} , \bar{i} They are written $\langle_a, _i_u\rangle$:

al-l_ah-u اللهُ $al-lāh^u$, 'il_ahNإله $ilāh^{un}$.
 al-ra.hm_an-uالرَّحْمَنُ $ar-raḥmān^u$, l_akinلَٰكِن $lākin$, h_ahunAهَٰهُنَا $hāhunā$,
 .hunayn-u bn-u 'is.h_aq-aحَنِينُ بْنُ إِسْحَاقَ $Hunayn^u$ bn^u $Ishāq^a$, rabb_i
 رَبِّ $rabbī$, al-`A.s_iالْعَاصِ $al-^c\bar{A}ṣī$.

Silent ي/و Some words ending with كَة are usually written حَة or نَة instead of كَة : see Wright (1896, i. 12 A). *arabluatex* preserves that particular writing; the same applies to words ending in يَة for كَة . Long vowels $\langle U, I \rangle$ shall receive no *sukūn* after a ^o*alif maḥdūfa* and are discarded in *trans* mode:

.hay_aUTuNحَيَوةٌ $hayāt^{un}$, .sal_aUTuNصَلَاةٌ $ṣalāt^{un}$, mi^sk_aUTuNمَشْكُوةٌ $miškāt^{un}$,
 tawr_aITuNتَوْرِيَّةٌ $tawrāt^{un}$.
 And so also: al-rib_aIT-uالرِّبِيَّةُ $ar-ribāt^u$.

^c*Amr^{un}*, and the silent و To that name a silent و is added to distinguish it from ^c*Umar^u*: see Wright (1896, i. 12 C). In no way this affects the sound of the *tanwīn*, so it has to be discarded in *trans* mode:

`amruNUعَمْرُو $^c\text{amr}^{un}$, `amraNUعَمْرُوا $^c\text{amr}^{an}$, `amriNUعَمْرُو $^c\text{amr}^{in}$.

When the *tanwīn* falls away (Wright 1896, i. 249 B): `amr-uU bn-u mu.hammadINعَمْرُو بْنُ مُحَمَّدٍ $^c\text{Amr}^u$ bn^u $Muḥammad^{in}$, mu.hammad-u bn-u `amr-iU bn-i_hAlidiNعَمْرُو بْنُ خَالِدٍ $Muḥammad^u$ bn^u $^c\text{Amr}^i$ bnⁱ $Hālid^{in}$.

And so also: al-rib_aUAالرِّبَا $ar-ribā$, ribaNUرِبَا rib^{an} .

tanwīn The marks of doubled short vowels, $\text{َ}, \text{ُ}, \text{ِ}$, are written $\langle uN, aN, iN \rangle$ respectively. *arabluatex* deals with special cases, such as َ taking an َ after all consonants except ة , and *tanwīn* preceding ى as in هَدَى , which is written $\langle aN_A \rangle$ or $\langle aNY \rangle$:

mAluNمَالٌ $māl^{un}$, bAbaNبَابًا $bāb^{an}$, madInaTaNمَدِينَةٌ $madīnat^{an}$, bintiNبِنْتٌ $bint^{in}$ maqhaN_Aمَقْهَى $maqha^n$, fataNYفَتَى $fata^n$.
arabluatex is aware of special orthographies: `say'uNشَيْءٌ $ṣay^{un}$,
 `say'aNشَيْئًا $ṣay^{an}$, `say'iNشَيْءٍ $ṣay^{in}$.

In some cases, it may be useful to mark the root form of defective words so as to produce a more accurate transliteration of ending *tanwīn*. As seen above, *tanwīn* preceding ى is written ⟨*aN_A*⟩ or ⟨*aNY*⟩. Such forms as قَاضٍ may likewise be written ⟨*iNI*⟩:—

al-qA.dI الْقَاضِي *al-qāḍī*, qA.diyaN قَاضِيًا *qāḍiyāⁿ*, qA.diNI قَاضٍ *qāḍiⁿ*.

4.2 Other orthographic signs

tāʾ marbūṭa It is written ⟨*T*⟩:

madInaTuN مَدِينَةٌ *madīnat^{un}*, madInaTaN مَدِينَةً *madīnat^{an}*, madInaTiN مَدِينَةٍ *madīnatⁱⁿ*.

hamza It is written ⟨*ʾ*⟩, its carrier being determined by contextual analysis. In case one wishes to bypass this mechanism, he can use the “quoting” feature that is described below in section 4.4 on page 18.

Initial *hamza*: 'asaduN أَسَدٌ *ʾasad^{un}*, 'u_htuN أُخْتُ *ʾuḥt^{un}*, 'iqlIduN إِقْلِيدٌ *ʾiqlīd^{un}*, 'anna أَنْ *ʾanna*, 'inna إِنْ *ʾinna*.

hamza followed by the long vowel و is encoded ' _U: ' _U1_A أُولَى *ʾulā*, ' _U1U أُولُو *ʾulū*, ' _U1A' ika أُولَٰئِكَ *ʾulāʾika*.

hamza followed by the long vowel ي is encoded ' _I: ' _ImAnuN إِيْمَانٌ *ʾīmān^{un}*.

Middle *hamza*: xA.ti'-Ina خَاطِئِينَ *ḫāṭiʿīna*, ru'UsuN رُؤُوسٌ *ruʾūs^{un}*, xA.tI'aTuN خَاطِئَةً *ḫāṭiʿat^{un}*, su'ila سُئِلَ *suʾila*, 'as'ilaTuN أَسْئَلُهُ *ʾasʾilat^{un}*, mas'alaTuN مَسْأَلَةٌ *masʾalat^{un}*, 'as'alu أَسْأَلُ *ʾasʾalu*, yatasA'alUna يَتَسَاءَلُونَ *yatasāʾalūna*, murU'aTuN مُرُوءَةٌ *murūʾat^{un}*, ta'xIruN تَأْخِرُ *taʾḫīr^{un}*, ta'axxara تَأَخَّرَ *taʾaḫḫara*, ji'tu-ka جِئْتُكَ *ǧiʾtu-ka*, qA'iluN قَائِلٌ *qāʾil^{un}*.

From Wright (1896, i. 14 B):— All consonants, whatsoever, not even *ʾalifhènzatum* excepted, admit of being doubled and take *tašdīd*. Hence we speak and write ra' 'AsuN رَأْسٌ *raʾās^{un}*, sa' 'AluN سَأَلَ *saʾāl^{un}*, na' 'AjuN نَأَجَّ *naʾāǧ^{un}*.

Final *hamza*: xA.ta'uN خَطَأٌ *ḫaṭaʾ^{un}*, xA.ta'aN خَطَأً *ḫaṭaʾ^{an}*, xA.ta'iN خَطِئًا *ḫaṭaʾⁱⁿ*, 'aqra'u أَقْرَأُ *ʾaqraʾu*, taqra'Ina تَقْرَأِينَ *taqraʾīna*, taqra'Una تَقْرَأُونَ *taqraʾūna*, yaqra'na يَقْرَأْنَ *yaqraʾna*, yaxba'Ani يَخْبَأَنَّ *yaxbaʾāni*, xaba'A خَبَأَ *ḫabaʾā*, xubi'a خُبِيَ *ḫubiʾa*, xubi'UA خُبُوا *ḫubiʾū*, jA'a

جَاءَ $g\bar{a}^a$, ridA'uN رَدَاءُ $rid\bar{a}^{un}$, ridA'aN رَدَاءُ $rid\bar{a}^{an}$, jI'a جِيءَ $g\bar{i}^a$,
radI'iN رَدِيءُ $rad\bar{i}^{in}$, sU'uN سُوءُ $s\bar{u}^{un}$, .daw'uN ذَوُّ daw^{un} , qay'iN قِيءُ qay^{in} .

ˆsay'uN شَيْءُ $\check{s}ay^{un}$, ˆsay'aN شَيْئًا $\check{s}ay^{an}$, ˆsay'iN شَيْءٍ $\check{s}ay^{in}$, al-ˆsay'-u
أَلشَّيْءُ $a\check{s}-\check{s}ay^{u}$, 'aˆsyA'-u أَشْيَاءُ $^a\check{s}y\bar{a}^{u}$, 'aˆsyA'-a أَشْيَاءُ $^a\check{s}y\bar{a}^{a}$, .zim'aN
ظِمًّا $z\bar{i}m^{an}$.

tahfif^u 'l-hamzatⁱ: if the *hamza* has *ğazma* and is preceded by *ˆalif hamzatum*, it must be changed into the letter of prolongation that is homogeneous with the preceding vowel; hence: 'a'mana أَمَنَ $^a\bar{m}ana$,
'u'minu أَوْمِنُ $^u\bar{m}inu$, 'i'manuN إِيمَانُ $^i\bar{m}\bar{a}n^{un}$. For other possible ways of encoding such sequences, see on page 12 (*hamza* followed by و and ي) and the *madda* on this page.

madda At the beginning of a syllable, *ˆalif* with *hamza* and *fatha* (أ) followed by *ˆalifu* 'l-maddi (*ˆalif* of prolongation) or *ˆalif* with *hamza* and *ğazma* (إ) are both represented in writing *ˆalif* with *madda*: آ (see Wright 1896, i. 25 A–B).

Hence one should keep to this distinction and encode 'a'kulu أَكُلُ $^a\bar{k}ulu$ and
'Akilun أَكِلُ $^a\bar{k}il^{un}$ respectively.

arbluatex otherwise determines *al-ˆalif^u 'l-mamdūdāt^u* by context analysis.

'is'AduN إِسَادُ $^is\bar{a}d^{un}$, 'AkilUna أَكِلُونُ $^a\bar{k}il\bar{u}na$, 'a'manna أَمَنَّا $^a\bar{m}ann\bar{a}$,
al-qur'An-u الْقُرْآنُ $al-qur\bar{a}n^u$.

jA'a جَاءَ $g\bar{a}^a$, yatasA'alUna يَتَسَاءَلُونَ $yatas\bar{a}^al\bar{u}na$, ridA'uN رَدَاءُ $rid\bar{a}^{un}$,
xaba'A خَبَأَ $h\bar{a}b\bar{a}^a$, yaxba'Ani يَخْبَأُ $yah\bar{b}a^ani$.

šadda *tašdīd* is either *necessary* or *euphonic*.

The necessary *tašdīd* always follows a vowel, whether short or long (see Wright 1896, i. 15 A–B). It is encoded in writing the consonant that carries it twice:

ˆallaqa عَاقَ $^a\bar{l}laqa$, mAdduN مَادُّ $m\bar{a}dd^{un}$, 'ammara أَمَّرَ $am\bar{m}ara$, murruN
مُرَّرَ $murr^{un}$.

The euphonic *tašdīd* always follows a vowelless consonant which is passed over in pronunciation and assimilated to a following consonant. It may be found (Wright 1896, i. 15 B–16 C):—

- (a) With the *solar* letters ت, ث, د, ذ, ر, ز, س, ش, ص, ض, ط, ظ, ل, ن, after the article *al*:—

Unlike *arabtex* and *arabxetex*, *arabluatex* *never* requires the *solar* letter to be written twice, as it automatically generates the euphonic *tašdīd* above the letter that carries it, whether the article be written in the assimilated form or not, e.g. *al-^ˀsams-u* الشَّمْسُ *aš-šams^u*, or *a^ˀ-s-^ˀsams-u* الشَّمْسُ *aš-šams^u*.

al-tamr-u التَّمْرُ *at-tamr^u*, *al-ra.hm_an-u* الرَّحْمَنُ *ar-raḥmān^u*, *al-.zulm-u* الظُّلْمُ *aẓ-ẓulm^u*, *al-lu.gaT-u* اللُّغَةُ *al-luġat^u*.

- (b) With the letters ر, ل, م, و, ي after ن with *ğazma*, and also after the *tanwīn*:—

Note the absence of *sukūn* above the passed over ن in the following examples, each of which is accompanied by a consistent transliteration: *min rabbi-hi* مِنْ رَبِّهِ *mir rabbi-hi*, *min layliN* مِنْ لَيْلٍ *mil laylⁱⁿ*, *'an yaqtula* أَنْ يَقْتُلَ *'ay yaqtula*.

With *tanwīn*: *kitAbuN* مِيقَاتٍ *kitāb^{um} mubīn^{un}*.

- (c) With the letter ت after the dentals ث, د, ذ, ض, ط, ظ in certain parts of the verb: this kind of assimilation, e.g. لَبِثْتُ for لَبِثْتُ *labittu*, will be discarded here, as it is largely condemned by the grammarians (see Wright 1896, i. 16 B–C).

The definite article and the *ʿalif* *ʿl-waṣṭi* At the beginning of a sentence, *ʿ* is never written, as اَلْحَمْدُ لِلّٰهِ; instead, to indicate that the *ʿalif* is a connective *ʿalif* (*ʿalif^u ʿl-waṣṭi*), the *hamza* is omitted and only its accompanying vowel is expressed:

al-.hamd-u li-l-l_ah-i اَلْحَمْدُ لِلّٰهِ *al-ḥamd^u li-llāhⁱ*.

As said above on page 5, *fullvoc* is the mode in which *arabluatex* expresses the *sukūn* and the *waṣṭa*. *arabluatex* will take care of doing that automatically provided that the vowel which is to be absorbed by the final vowel of the preceding word is properly encoded, like so:—

- (a) Definite article at the beginning of a sentence is encoded

al-, or *a<solar letter>-*

if one wishes to mark the assimilation—which is in no way required, as *arabluatex* will detect all cases of assimilation.

- (b) Definite article inside sentences is encoded

'l- or *'<solar letter>-*.

- (c) In all remaining cases of elision, the ^ʔ*alifu* 'l-*waṣli* is expressed by the vowel that accompanies the omitted *hamza*: ⟨u, a, i⟩.

Article: bAb-u 'l-madrasaT-i الْمَدْرَسَةُ بَابُ *bāb*^u 'l-madrasatⁱ, al-maqAlaT-u 'l-'_Ul_A الْمَقَالَةُ الْأُولَى *al-maqālat*^u 'l-^ʔ*ulā*, al-lu.gaT-u 'l-'arabiyyaT-u فِي صِنَاعَةِ اللُّغَةِ الْعَرَبِيَّةِ *al-luġat*^u 'l-^c*arabiyyat*^u, fI .sinA`aT-i 'l-.tibb-i صِنَاعَةُ الْطَبِّ *fī sinā`at*ⁱ 't-tibbⁱ, 'il_A 'l-intiqA.d-i إِلَى الْإِنْتِقَاضِ *ila* 'l-*intiqād*ⁱ, fI 'l-ibtidA-i فِي الْإِبْتِدَاءِ *fī* 'l-*ibtidā`*ⁱ, 'abU 'l-wazIr-i أَبُو الْوَزِيرِ *abu* 'l-*wazīr*ⁱ, fa-lammA ra'aW 'l-najm-a فَلَمَّا رَأَوْا النَّجْمَ *fa-lammā ra`awu* 'n-naġm^a.

Particles:—

- (a) *li*:- ^ʔ*alif*^u 'l-*waṣl*ⁱ is omitted in the article *أل* when it is preceded by the preposition *لِ*: *li*-l-rajul-i لِلرَّجُلِ *li-r-raġul*ⁱ.
If the first letter of the noun be *ل*, then the *ل* of the article also falls away, but arabluatex is aware of that: *li*-l-laylaT-i لِلَّيْلَةِ *li-l-laylat*ⁱ.
- (b) *la*:- the same applies to the affirmative particle *لَ*: *la*-l-.haqq-u لِلْحَقِّ *la-l-ḥaqq*^u.
- (c) With the other particles, ^ʔ*alif*^u 'l-*waṣl*ⁱ is expressed: fI 'l-madInaT-i فِي الْمَدِينَةِ *fī* 'l-*madīnat*ⁱ, wa-'l-rajul-u وَالرَّجُلُ *wa-'r-raġul*^u, bi-'l-qalam-i بِالْقَلَمِ *bi-'l-qalam*ⁱ, bi-'l-ru`b-i بِالرُّعْبِ *bi-'r-ru`b*ⁱ.

Perfect active, imperative, nomen actionis: qAla isma` قَالَ أَسْمَعُ *qāla* 'sma^c, qAla uqtul قَالَ أَقْتُلُ *qāla* 'qtul, huwa inhazama هُوَ أَنْهَزَمَ *huwa* 'nhazama, wa-ustu`mila وَأَسْتَعْمِلَ *wa-'stu`mila*, qad-i in.sarafa قَدْ أَنْصَرَفَ *qadi* 'nṣarafa, al-iqtidAr-u الْأَقْتِدَارُ *al-iqtidār*^u, 'il_A 'l-intiqA.d-i إِلَى الْإِنْتِقَاضِ *ila* 'l-*intiqād*ⁱ, lawi istaqbala لَوْ اسْتَقْبَلَ *lawi* 'staqbala.

Other cases: 'awi ismu-hu أَوْ اسْمُهُ *'awi* 'smu-hu, zayduN ibn-u `amriNU زَيْدُ بْنُ عَمْرٍو *Zayd*^{uni} 'bn^u ^c*Amr*ⁱⁿ,¹⁴ `umar-u ibn-u 'l-_ha.t.tAb-i عُمَرُ بْنُ عَبْدِ اللَّهِ *'umar*^u 'bn^u 'l-*Ḥaṭṭāb*ⁱ,¹⁵ imru'-u 'l-qays-i إِمْرُؤُ الْقَيْسِ *Imru*^u 'l-*Qays*ⁱ, la-aymun-u 'l-l_ah-i لَا يُؤْمِنُ اللَّهُ *la-'ymun*^u 'l-*lāh*ⁱ.

^ʔ*alif*^u 'l-*waṣl*ⁱ preceded by a long vowel The long vowel preceding the connective ^ʔ*alif* is shortened in pronunciation (Wright 1896, i. 21 B–D). This does

¹⁴ “*Zayd* is the son of ^c*Amr*”: the second noun is not in apposition to the first, but forms part of the predicate. Hence زَيْدُ بْنُ عَمْرٍو and not زَيْدُ بْنُ عَمْرٍو, “*Zayd*, son of ‘*Amr*’.”

¹⁵ “^c*Umar* is the son of *al-Ḥaṭṭāb*” (see footnote 14).

not appear in the Arabic script, but arabuatex takes it into account in some transliteration standards:—

fI 'l-nAs-i فِي النَّاسِ fi 'n-nāsⁱ, 'abU 'l-wazIr-i أَبُو الْوَزِيرِ ^{abu} 'l-wazīrⁱ,
fI 'l-ibtidA'-i فِي الْإِبْتِدَاءِ fi 'l-ibtidā^{ai}, _dU 'l-i`lAl-i ذُو الْأَعْلَالِ du
'l-i^clālⁱ, maqh_A 'l-'amIr-i مَقْهَى الْأَمِيرِ maqha 'l-^aamīrⁱ.

^aalif^u 'l-waṣlⁱ preceded by a diphthong The diphthong is resolved into two simple vowels (Wright 1896, i. 21 D–22 A) viz. *ay* → *āi* and *aw* → *āū*. arabuatex detects the cases in which this rule applies:—

fI `aynay 'l-malik-i فِي عَيْنِي الْمَلِكِ ^{aynayi} 'l-malikⁱ, ix[^]say 'l-qawm-a
إِخْشَى الْقَوْمِ iḥṣayi 'l-qawm^a, mu.s.tafaw 'l-l_ah-i مُصْطَفَوْا اللَّهَ muṣṭafawu
'l-lāhⁱ.
ramaW 'l-.hijAraT-a رَمَوْا الْحِجَارَةَ ramawu 'l-ḥiğārat^a, fa-lammAra'aW 'l-najm-a
فَلَمَّا رَأَوْا النَّجْمَ fa-lammā ra^aawu 'n-nağm^a.

^aalif^u 'l-waṣlⁱ preceded by a consonant with *sukūn* The vowel which the consonant takes is either its original vowel, or that which belongs to the connective ^aalif or the *kasra*; in most of the cases (Wright 1896, i. 22 A–C), it is encoded explicitly, like so:—

'antumU 'l-kA_dib-Una أَنْتُمْ الْكَادِبُونَ ^{antumU} 'l-kādib^{ūna}, ra'aytumU
'l-rajul-a رَأَيْتُمُ الرَّجُلَ ra^aaytumU 'r-rağul^a, mani 'l-ka_d_dAb-u مَنِ
الْكَذَّابِ mani 'l-kaddāb^u, qatalati 'l-rUm-u قَتَلَتِ الرُّومُ qatalati 'r-Rūm^u.

However, the Arabic script does not shows the *kasra* which is taken by the nouns having *tanwīn* although it is explicit in pronunciation and must appear in some transliteration standards. arabuatex takes care of that automatically:—

mu.hammaduN 'l-nabI مُحَمَّدٌ النَّبِيُّ Muḥammad^{uni} 'n-nabī.

4.3 Special orthographies

The name of God The name of God, اللَّهُ, is compounded of the article اَلْ, and إِلَهِ (noted إِلَهِ with the defective ^aalif) so that it becomes اَلْإِلَهِ; then the *hamza* is suppressed, its vowel being transferred to the ل before it, so that there remains اَللَّهُ (I refer to Lane, *Lexicon*, I. 83 col. 1). Finally, the first ل is made quiescent and incorporated into the other, hence the *tašdīd* above it. As arabuatex never requires a solar letter to be written twice (see above, on page 14), the name of God is therefore encoded al-l_ah-u or 'l-l_ah-u:—

al-l_ah-u اللهُ *al-lāh^u*, yA|¹⁶ al-l_ah-u اللهُ يَا *al-lāh^u*, 'a-fa|¹⁷-al-l_ah-i
 la-ta.g`alanna أَفَاللهُ لَتَعْلَنَّ *ʔa-fa-al-lāhⁱ la-taḡ^calanna*, bi-'l-l_ah-i
 بِاللهِ *bi-'l-lāhⁱ*, wa-'l-l_ah-i وَاللهِ *wa-'l-lāhⁱ*, bi-sm-i 'l-l_ah-i بِسْمِ
 اللهُ *bi-smⁱ 'l-lāhⁱ*, al-.hamd-u li-l-l_ah-i اللَّهُ أَحْمَدُ *al-ḥamd^u li-llāhⁱ*,
 li-l-l_ah-i 'l-qA'il-u لِلْقَائِلِ اللهُ *li-llāhⁱ 'l-qā^ʔil^u*.

The conjunctive الَّذِي Although it is compounded of the article اَلْ, the demonstra-
 tive letter ل and the demonstrative pronoun ذَا, both masculine and feminine forms
 that are written defectively are encoded alla_dI and allatI respectively. Forms
 starting with the connective *ʔalif* are encoded 'lla_dI and 'llatI:—

'a_hAfu mina 'l-malik-i 'lla_dI ya.zlimu 'l-nAs-a أَخَافُ مِنَ الْمَلِكِ
ʔaḥāfu mina 'l-malikⁱ 'lladī yazlimu 'n-nās^a, `udtu
 'l-~say_h-a 'lla_dI huwa marI.duN عُدْتُ الشَّيْخَ الَّذِي هُوَ مَرِيضٌ *ʔudtu*
'š-šayḥ^a 'lladī huwa marīd^{un}, mA 'anA bi-'lla_dI qA'iluN la-ka
 ~say'aN أَنَا بِالَّذِي قَائِلٌ لَكَ شَيْئًا *mā ʔanā bi-'lladī qā^ʔil^u la-ka šay^{an}*.
 'ari-nA 'lla_dayni 'a.dallA-nA mina 'l-jinn-i wa-'l-'ins-i أَرِنَا
 الَّذِينَ أَضَلَّانَا مِنَ الْجِنِّ وَالْإِنْسِ *ʔari-nā 'lladayni ʔadallā-nā mina 'l-ḡinnⁱ*
wa-'l-ʔinsⁱ.

The other forms are encoded regularly as al-l or 'l-l:—

fa-'innA na_dkuru 'l-.sawt-ayni 'l-la_dayni rawaynA-humA `an
 ja.h.zaT-a فَانَا نَذْكُرُ الصَّوْتَيْنِ الَّذِينَ رَوَيْنَاهُمَا عَنْ جَهْظَةٍ *fa-ʔinnā nadkuru 'š-šawt^{ayni}*
'l-ladayni rawaynā-humā ʔan Ġaḥẓat^a.

And also: al-la_dAni الَّذِينَ *al-ladāni*, al-la_dayni الَّذِينَ *al-ladayni*,
 al-latAni اللَّاتَانِ *al-latāni*, al-latayni اللَّاتَيْنِ *al-latayni*, al-lAtI اللَّاتِي *al-lātī*,
 al-lA'|Ati¹⁸ اللَّائِيَاتِ *al-lāʔāti*, al-lA'I اللَّائِي *al-lāʔī*, and so forth.

¹⁶Note the “pipe” character ‘|’ here after yA and below after fa before footnote mark 17: it is needed
 by the dmḡ transliteration mode as in this mode any vowel at the commencement of a word preceded by
 a word that ends with a vowel, either short or long, is absorbed by this vowel viz. *ʔala 't-ṭarīqⁱ*. See
 section 4.5 on page 20 on the “pipe” and section 6 on page 22 on dmḡ mode.

¹⁷See footnote 16.

¹⁸Note here the “pipe” character ‘|’: as already stated on page 13, the sequence 'A usually encodes
ʔalif with *hamza* followed by *ʔalif* of prolongation, which is represented in writing *ʔalif* with *madda*: ٲ.
 The “pipe” character prevents this rule from being applied. See section 4.5 on page 20.

4.4 Quoting

It is here referred to “quoting” after the package `arabtex`.¹⁹ The “quoting” mechanism of `arabluatex` is designed to be very similar in effect to the one of `arabtex`.

To start with an example, suppose one types the following in `novoc` mode: عَلَّمَ علم الهيئة; is it عَلَّمَ, *he was taught the science of astronomy*, or عَلَّمَ, *he taught the science of astronomy*? In order to disambiguate this clause, it may be sensible to put a *damma* above the first عَلَّمَ علم الهيئة, which is achieved by “quoting” the vowel *u*, like so: `ullima, or, with no other vowel than the required *u*: `ullm.

This is how the “quoting” mechanism works: metaphorically speaking, it acts as a *toggle switch*. If something, in a given mode, is supposed to be visible, “quoting” hides it; conversely, if it is supposed not to, it makes it visible.

As shown above, “quoting” means inserting one straight double quote (") *before* the letter that is to be acted upon. Its effects depend on the mode which is currently selected, either `novoc`, `voc` or `fullvoc`:—

novoc In this mode, “quoting” essentially means make visible something that ought not to be so.

(a) Quoting a vowel, either short or long, makes the *damma*, *fatha* or *kasra* appear above the appropriate consonant:—

`ullima `ilm-a 'l-hay'aT-i علم الهيئة ^uullima ⁱilm^a 'l-hay^{at}i,
ya.gz"UA يغزوا *yağzū*.

(b) The same applies when “quoting” the *tanwīn*:—

wa-'innA sawfa tudriku-nA 'l-manAyA muqadd"araT"aN وإنا سوف
تدركنا المنيا مقدراً ^{an}, wa-^{an}innā sawfa tudriku-na 'l-manāyā muqaddarat^{an}.

(c) If no vowel follows the straight double quote, then a *sukūn* is put above the preceding consonant:—

qAla isma`" قال اسمع ^cqāla 'sma^c, jA'at" hinduN جَاءَتْ هِنْدُ ^{at}gā^{at}
Hind^{um}, `sabIhuN bi-man q"u.ti`at" qadamA-hu شبيه بمن قُطِعَتْ ^{um}qadamā-hu.
قدماه *šabīh^{um} bi-man quṭi^cat qadamā-hu*.

(d) At the commencement of a word, the straight double quote is interpreted as *ʿalif*^u 'l-waṣl':—

wa-"ust"u`mila وأستعمل ^uwa-'stu^umila, huwa "inhazama هو أَنهزم ^uhuwa
'nhazama, al-"intiqA.du الانتقاض ^ual-intiqādu.

voc In accordance with the general rule, in this mode, “quoting” makes the vowels and the *tanwīn* disappear, should this feature be required for some reason:—

(a) Short and long vowels:—

¹⁹See Lagally (2004, p. 22)

q"Ala q"A'iluN قَالَ قَاتِلُ *qāla qā'il^{un}*, ibn-u 'abI 'u.saybi`aT-"a
 اِبْنُ أَبِي أُصَيْبَةَ *Ibn^u 'Abi 'Uṣaybi'at^a*.

(b) *tanwīn*:—

madInaT"aN مَدِينَةٌ *madīnat^{an}*, bAb"aN أَبَا *bāb^{an}*, hud"aN_A هُدًى *hudaⁿ*,
 ˆsay'"iN شَيْءٌ *šayⁱⁿ*.

One may more usefully “quote” the initial vowels to write the *waṣla* above the *ʔalif* or insert a straight double quote after a consonant not followed by a vowel to make the *sukūn* appear:—

(a) *ʔalif^u 'l-waṣlⁱ*:—

fI "istisqA'-a فِي اسْتِسْقَاءٍ *fi 'stisqā^{aa}*, wa-"istisqA'-u وَاسْتِسْقَاءٌ *wa-'stisqā^{au}*,
 qAla "uhrub fa-lan tuqtala قَالَ أَهْرُبُ فَلَنْ تُقْتَلَ *qāla 'hrub fa-lan tuqtala*.

(b) *sukūn*:—

qAla "uqtul" fa-lan tuqtala قَالَ أَقْتُلُ فَلَنْ تُقْتَلَ *qāla 'qtul fa-lan tuqtala*, mA jA'at" mini imra'aTiN مَا جَاءَتْ مِنْ امْرَأَةٍ *mā ḡā'at mini 'mra^{atⁱⁿ}*, kam" qad" ma.dat" min" laylaTiN كَمْ قَدْ مَضَتْ مِنْ لَيْلَةٍ *kam qad maḡat min laylatⁱⁿ*.

fullvoc In this mode, “quoting” may be used to take away any short vowel (or *tanwīn*, as seen above) or any *sukūn*:—

al-jamr-u 'l-.sayfiyy-u 'lla_dI kAna bi-q"rAn" |nUn-a الْجَمْرُ الصَّفِيُّ *al-ḡamr^u 'ṣ-sayfiyy^u 'lladī kāna bi-Qrānnūn^a*.

4.4.1 Quoting the *hamza*

As said above in section 4.2 on page 12, the *hamza* is always written ⟨ ' ⟩, its carrier being determined by contextual analysis. “Quoting” that straight single quote character like so: ⟨ " ' ⟩ allows to determine the carrier of the *hamza* freely, without any consideration for the context. Table 4 gives the equivalents for all the possible carriers the *hamza* may take:

Letter	Transliteration ²⁰		ArabTeX notation
	dmg	loc	
ء	ʔ	'	" '
أ	ʔā	'ā	A " '
إ	ʔ	'	a " '

²⁰See below section 6 on page 22.

Letter	Transliteration		ArabTeX notation
	dmg	loc	
أ	ʾ	'	u"'
و	ʾ	'	w"'
إ	ʾ	'	i"'
ئ	ʾ	'	y"'

Table 4: “Quoted” *hamza*

As one can see from table 4 on page 19, the carrier of the *hamza* is inferred from the letter that precedes the straight double quote ⟨"⟩. Of course, any “quoted” *hamza* may take a short vowel, which is to be written *after* the ArabTeX equivalent for the *hamza* itself, namely ⟨'⟩. For example, أَعْدَاؤُكُمْ is encoded ⟨w" 'a⟩, while أَعْدَاءُكُمْ is encoded ⟨w" ' '⟩. In the latter example, the second straight double quote encodes the *sukūn* in voc mode in accordance with the rule laid above on pages 18–19.

'a`dA'ukum أَعْدَاؤُكُمْ $\text{ʾa}^c d\bar{a}^{\text{v}} ukum$, 'a`dA|" 'ukum أَعْدَاءُكُمْ $\text{ʾa}^c d\bar{a}^{\text{v}} ukum$,
'a`dA'ikum أَعْدَائِكُمْ $\text{ʾa}^c d\bar{a}^{\text{v}} ikum$, 'a`dA|" 'ikum أَعْدَاءُكُمْ $\text{ʾa}^c d\bar{a}^{\text{v}} ikum$.

4.5 The “pipe” character (|)

In the terminology of ArabTeX, the “pipe” character ‘|’ is referred to as the “invisible consonant”. Hence, as already seen above in section 4.4.1 on page 19, its usage to encode the *hamza* alone, with no carrier: |" ' ء.

Aside from that usage, the “pipe” character is used to prevent almost any of the contextual analysis rules that are described above from being applied. Two examples have already been given to demonstrate how that particular mechanism works in footnote 16 on page 17 and in footnote 18 on page 17. One more example follows:—

bi-qraN|nUn-a بِقْرَانُونْ $bi\text{-}Qrānnūn^a$, “in Crannon” (Thessaly, Greece).²¹

As one can see, the “pipe” character between the two ⟨n⟩ prevents the necessary *tašdād* rule (page 13) from being applied.

4.6 Stretching characters: the *taṭwīl*

A double hyphen ⟨--⟩ stretches the ligature in which one letter is bound to another. Although it is always better to rely on automatic stretching, this technique may be used to a modest extent, especially to increase legibility of letters and diacritics which stand one above the other:—

.hunayn-u bn-u 'is.h--_aq-a $\text{حُنَيْنُ بْنُ إِسْحَاقَ}$ $Hunayn^u bn^u ʾIshāq^a$

²¹See more context on page 19.

4.7 Digits

4.7.1 Numerical figures

The *Indian numbers*, *ar-raḡam*^u *ʿl-hindiyy*^u, are ten in number, and they are compounded in exactly the same way as our numerals:—

1874 ١٨٧٤, 123-456,789 ١٢٣-٤٥٦,٧٨٩, fI sanaT-i 1024 ١٠٢٤ فِي سَنَةِ

4.7.2 The abjad

The numbers may also be expressed with letters from right to left arranged in accordance with the order of the Hebrew and Aramaic alphabets (see Wright 1896, i. 28 B–C). The *ʿabġad* numbers are usually distinguished from the surrounding words by a stroke placed over them.

`\abjad` *ʿabġad* numbers are inserted with the command `\abjad{⟨number⟩}` in any of the `voc`, `fullvoc` and `novoc` modes, where `⟨number⟩` may be any number between 1 and 1999, like so:—

`\abjad{45}` kitAbu-hu fI 'l-`AdAt-i مَ كَآبُهُ فِي الْعَادَاتِ 45 kitābu-hu fi ʿl-*cādāt*.

REM. *a*. As can be seen in the above given example, `arabluatex` expresses the *ʿabġad* numbers in Roman numerals if it finds the command `\abjad` in any of the transliteration modes.

REM. *b*. `\abjad` may also be found outside Arabic environments. In that case, `arabluatex` does not print the stroke as a distinctive mark over the number for it is not surrounded by other Arabic words. In case one nonetheless wishes to print the stroke, he can use the `\aemph` command that is described below in section 4.9 on the following page:—

The `\arb[trans]{ʿabjad}` number for 1874 is `\abjad{1874}` The *ʿabġad* number for 1874 is غَضْعَد.

The `\arb[trans]{ʿabjad}` number for 1874 is `\aemph{\abjad{1874}}` The *ʿabġad* number for 1874 is غَضْعَد.

4.8 Additional characters

In the manuscripts, the unpointed letters, *al-ḡurūfu* *ʿl-muḡmalatu*, are sometimes further distinguished from the pointed by various contrivances, as explained in Wright (1896, i. 4 B–C). One may find these letters written in a smaller size below the line, or with a dot or another mark below. As representing all the possible contrivances leads to much complexity and also needs to be agreed among scholars, new ways of encoding them will be proposed and gradually included as `arabluatex` will mature.

For the time being, the following is included:—

Letter	Transliteration ²²	ArabT _E X notation
	dmg loc	
ب	b	.b

²²See below section 6 on the next page.

Letter	Transliteration		ArabT _E X notation
	dmg	loc	
د	<i>d</i>	<i>d</i>	<code>ˆd</code>
ف	<i>f</i>	<i>f</i>	<code>.f</code>
ق	<i>q</i>	<i>q</i>	<code>.q</code>
ك	<i>k</i>	<i>k</i>	<code>.k</code>
ن	<i>n</i>	<i>n</i>	<code>.n</code>
⸔	<i>(</i>	<i>(</i>	<code>((</code>
⸕	<i>)</i>	<i>)</i>	<code>))</code>

Table 5: Additional Arabic codings

'afAman.tUs Gal.(M) .fmn.n.ts (sic) Gal.(E1), أفامنطوس Gal.(M) فمنطس (sic) Gal.(E1), *ʿafāmanṭūs* Gal.(M) *fmnṇṭs* (sic) Gal.(E1).

4.9 Arabic emphasis

As already seen in section 4.7.2 on page 21, the *ʿabǧad* numbers are distinguished from the surrounding words by a stroke placed over them. This technique is used to distinguish further words that are proper names or book titles.

`\aemph` One may use the command `\aemph{<Arabic text>}` to use the same technique to emphasize words, like so:—

`\abjad{45}: kitAbu-hu \aemph{fi 'l-`AdAt-i}` 45: مَهْ: كِتَابُهُ فِي الْعَادَاتِ
kitābu-hu fi 'l-^cĀdātⁱ.

5 Special applications

Linguistics The same horizontal stroke as the *taṭwīl* (see section 4.6 on page 20) may be encoded ``; `<BB>` will receive the *tašdīd*. This is useful to make linguistic annotations and comments on vowels:—

Bu Ba Bi BuN BaN BiN ^ˆu a i ^{un an in}, BBu BBa BBi ^ˆu a i, B--aN
^ˆ -^{an}, B" ^ˆ.

6 Transliteration

It may be more appropriate to speak of “romanization” than “transliteration” of Arabic. As seen above in section 2.2 on pages 5–7, the “transliteration mode” may be selected globally or locally.

This mode transliterates the ArabT_EX input into one of the accepted standards. As said above on page 5, two standards are supported at present:

dmg *Deutsche Morgenländische Gesellschaft*, which was adopted by the International Convention of Orientalist Scholars in Rome in 1935.²³ **dmg** transliteration convention is selected by default;

loc *Library of Congress*: this standard is part of a large set of standards for romanization of non-roman scripts adopted by the American Library Association and the Library of Congress.²⁴

More standards will be included in future releases of `arabluatex`.

`\SetTranslitConvention` **Convention** The transliteration mode, which is set to **dmg** by default, may be changed at any point of the document by the command `\SetTranslitConvention{<mode>}`, where `<mode>` may be either **dmg** or **loc**. This command is also accepted in the preamble should one wish to set the transliteration mode globally, eg.:—

```
1 \usepackage{arabluatex}
2 \SetTranslitConvention{loc}
```

`\SetTranslitStyle` **Style** Any transliterated Arabic text is printed in italics by default. This also can be changed either globally in the preamble or locally at any point of the document by the command `\SetTranslitStyle{<style>}`, where `<style>` may be any font shape selection command, eg. `\upshape`, `\itshape`, `\slshape`, and so forth.

New feature
v1.4

`\SetTranslitFont` **Font** `\SetTranslitFont{}` allows any specific font to be selected for rendering transliterated text with the font-selecting commands of the `fontspec` or `luaotfload` package. Of course, this font must have been defined properly. To take one example, here is how the *Gentium Plus* font may be used for rendering transliterated text:—

```
1 \newfontfamily\translitfont{Gentium Plus}[Ligatures=TeX]
2 \SetTranslitFont{\translitfont}
```

`\cap` **Proper names** Proper names or book titles that must have their first letters uppercased may be passed as arguments to the command `\cap{<word>}`. `\cap` is a clever command, for it will give the definite article *al-* in lower case in all positions. Moreover, if the initial letter, apart from the article, cannot be uppercased, viz. ^ʔ or ^ع, the letter next to it will be uppercased:—

`\cap{.hunayn-u}` `bn-u` `\cap{'is.h_aq-a}` حُنَيْنُ بْنُ إِسْحَقَ *Hunayn^u bn^u*
^ʔ*Ishāq^a*, `\cap{u_tm_an-u}` عُثْمَانُ *Uṭmān^u*, `.daraba` `\cap{zayd-u}`
`bn-u` `\cap{h_alidiN}` `\cap{sa`d-a}` `bn-a` `\cap{awf-i}` `bn-i` `\cap{`abd-i}`

²³See Brockelmann et al. (1935).

²⁴See <http://www.loc.gov/catdir/cpsd/roman.html> for the source document concerning Arabic language.

`\cap{'l-l_ah-i}` عَبْدُ اللَّهِ *daraba Zayd^u bn^u*
Ḥālidⁱⁿ Sa^{cd}a bn^a cAwf^f bnⁱ cAbdⁱ 'l-Lāhⁱ.

However, `\cap` must be used cautiously in some very particular cases, for the closing brace of its argument may prevent a rule from being applied. To take an example, as seen above on page 16, the transliteration of مُحَمَّدٌ النَّبِيُّ must be *Muḥammad^{uni} 'n-nabī*, as nouns having the *tanwīn* take a *kasra* in pronunciation before *ʿalifu 'l-waṣli*. In that case, encoding مُحَمَّدٌ like so: `\cap{mu.hamaduN}` is wrong, because the closing brace would prevent `arabluatex` from detecting the sequence $\langle -uN \rangle$ immediately followed by $\langle 'l \rangle$. Fortunately, this can be circumvented in a straightforward way by inserting only part of the noun in the argument of `\cap` vz. up to the first letter that is to be uppercased, like so: `\cap{m}u.hamaduN`.

Hyphenation In case transliterated Arabic words break the T_EX hyphenation algorithm, one may use the command `\-` to insert discretionary hyphens. This command will be discarded in all of the Arabic modes of `arabluatex`, but will be processed by any of the transliteration modes:—

`\cap{'abU} \cap{bakriN} \cap{mu\-.ham\-.madu} bnu \cap{za\-.ka`
`\-riy\-.yA'a} \cap{'l-rAzI}` أَبُو بَكْرٍ مُحَمَّدُ بْنُ زَكْرِيَّا الرَّازِي *Abū Bakrⁱⁿ Mu-*
ḥammad^u bn^u Zakariyyā^{ra} 'r-Rāzī.

6.1 Additional note on dmḡ convention

According to Brockelmann et al. (1935, p. 6), Arabic *ʿicrāb* may be rendered into **dmḡ** in three different ways:

- (a) In full: *cAmrun*;
- (b) As superscript text: *cAmr^{un}*;
- (c) Discarded: *cAmr*.

`\arbup` By default, `arabluatex` applies rule **b**. Once delimited by a set of Lua functions, *ʿicrāb* is passed as an argument on to a `\arbup` command which is set to `\textsuperscript`.

`\NoArbUp` `\NoArbUp` may be used either in the preamble or at any point of the document in case one wishes to apply rule **a**. The default rule **b** can be set back with `\ArbUpDflt` at any point of the document.

`\SetArbUp` Finally, `\SetArbUp{<formatting directives>}` may be used to customize the way *ʿicrāb* is displayed. To take one example, here is how Arabic *ʿicrāb* may be rendered as subscript text:—

```
1 \SetArbUp{\textsubscript{#1}}
2 Arabic |dmḡ| transliteration for \arb{ra'aytu ḡAmi`aN
3 muhaddamaTaN mi'_danatu-hu}: \arb[trans]{ra'aytu
4 ḡAmi`aN muhaddamaTaN mi'_danatu-hu.}
```

New feature
v1.3

Arabic dm̄g transliteration for رَأَيْتُ جَامِعاً مُدَمِّمَةً مِثْلَهُ: *raʔaytu ġāmiʕ^c_{am} muhaddamat_{am} miʔdanatu-hu*.

As shown in the above example, #1 is the token that is replaced with the actual *tanwīn* in the formatting directives of the `\SetArbUp` command.

ʔi^crāb boundaries Every declinable noun (*mu^crab*) may be declined either with or without *tanwīn*, viz. *munṣarif^{un}* or *ġayr^u munṣarifⁿ*. The former is automatically parsed by *arabluatex*, whereas the latter has to be delimited with an hyphen, like so:—

munṣarif: mu`allimuN مُعَلِّمٌ *mu^callim^{un}*, kA'inuN كَائِنٌ *kāʔin^{un}*, kA'inAtuN كَائِنَاتٌ *kāʔin^{ātun}*, \cap{`amraNU} عَمْرَوٌ *Amr^{an}*, fataN_A فَتًى *fataⁿ*, qA.diNI قَاضٍ *qāḍiⁿ*.

ġayr munṣarif: al-mu`allim-u الْمُعَلِّمُ *al-mu^callim^u*, kitAb-Ani كِتَابَانِ *kitāb^{āni}*, ra`sa'-Ani رِشَانِ *raša^{āni}*, sAriq-Una سَارِقُونَ *sāriq^{ūna}*, qA.d-Una قَاضُونَ *qāḍi^{ūna}*, al-.zulm-Atu الظُّلُمَاتُ *az-zulm^{ātu}*.

REM. a. As the *tanwīn* is passed over in pronunciation when it is followed by the letters ر, ل, م, ي (see item **b** on page 14), it may be desirable to further distinguish it by putting it above the line, but not to do the same for *ġayr munṣarif* terminations. This can be achieved by simply omitting the hyphen before any *ġayr munṣarif* termination:—

kAna .ganiyyaN l_akinna-hu labisa ġubbaTaN mumazzaqaN 'aydu-hA كَانَ غَنِيًّا لِكِنَّهُ لَبَسَ جُبَةً مُمَزَّقًا *kāna ġaniyy^{al} lākinna-hu labisa ġubbat^{am} mumazzaq^{an} ʔaydu-hā*.

REM. b. Although the hyphen before the *tanwīn* is optional as *arabluatex* always parses nouns with such termination, it may also be used to mark better the inflectional endings:—

mana`a 'l-nAs-a kAffaT-aN min mu_hA.tabati-hi 'a.had-uN bi-sayyidi-nA مَنَّ النَّاسُ كَافَّةً مِّنْ مُحَمَّدٍ *mana^a 'n-nās^a kāffat^{am} mim muḥāṭabati-hi ʔaḥad^{un} bi-sayyidi-nā*.

6.2 Examples

Here follows in transliteration the story of Ġuḥā and his donkey (جُحَا وَحَمَارُهُ). See the code on page 7:—

‘dm̄g’ standard: ʔatā ṣadīq^{un} ʔilā Ġuḥā yaṭlubu min-hu ḥimāra-hu li-yarkaba-hu fī safratⁱⁿ qaṣīratⁱⁿ fa-qāla la-hu: “sawfa ʔuʔidu-hu ʔilay-ka fī l-masāʔi wa-ʔadfaʕu la-ka ʔuġrat^{an}.” fa-qāla Ġuḥā: “ʔanā ʔāsif^{un} ġidd^{an} ʔannī lā ʔastaṭīʕu ʔan ʔuḥaqqiqa la-ka raġbata-ka fa-l-ḥimār^u laysa huna l-yawm^a.” wa-qabla ʔay yutimma Ġuḥā kalāma-hu badaʔa l-ḥimār^u yanhaqu fī ʔiṣṭabli-hi. fa-qāla la-hu ṣadīqu-hu: “ʔinnī ʔasmaʕu ḥimāra-ka yā Ġuḥā yanhaqu.” fa-qāla la-hu Ġuḥā: “ġarīb^{un} ʔamru-ka yā ṣadīqī ʔa-tuṣaddīqu l-ḥimār^a wa-tukaddība-nī?”

‘loc’ standard: atā ṣadīqun ilā Juḥā yaṭlubu min-hu ḥimāra-hu li-yarkaba-hu fī safratin qaṣīratin fa-qāla la-hu: “sawfa uʔidu-hu ilay-ka fī al-masāʔi wa-adfaʕu

`\SetInputScheme`

la-ka ujratān.” fa-qāla Juḥā: “anā āsifun jiddan annī lā astatī‘u an uḥaqqiqa la-ka raghbata-ka fa-al-ḥimāru laysa hunā al-yawma.” wa-qabla an yutimma Juḥā kalāma-hu bada’a al-ḥimāru yanhaqu fī iṣṭabli-hi. fa-qāla la-hu ṣadīqu-hu: “innī asma‘u ḥimāra-ka yā Juḥā yanhaqu.” fa-qāla la-hu Juḥā: “gharībun amru-ka yā ṣadīqī a-tuṣaddiqu al-ḥimāra wa-tukadhdhiba-nī?”

7 Buckwalter input scheme

Even though `arabluatex` is primarily designed to process the `ArabTeX` notation, it can also process the Buckwalter input scheme to a large extent.²⁵ The Buckwalter scheme is actually processed in two steps, as it is first converted into `ArabTeX`. Then, once this is accomplished, the `ArabTeX` scheme is processed through the above described functions. In this way, the Buckwalter input scheme can make the most of the `arabluatex` special features that are presented in section 2.2 on page 5.

The input scheme, which is set to `arabtex` by default, may be changed at any point of the document by the command `\SetInputScheme{<scheme>}`, where `<scheme>` may be either `arabtex` or `buckwalter`. This command is also accepted in the preamble should one wish to set the input scheme globally, like so:—

```
1 \usepackage{arabluatex}
2 \SetInputScheme{buckwalter}
```

‘base’, ‘xml’ and ‘safe’ schemes `arabluatex` can use any of the so-called Buckwalter ‘base’, ‘xml’ or ‘safe’ schemes as they are described in Habash (2010, pp. 25–26).²⁶ However, the following limitation apply to the ‘base’ and ‘xml’ schemes: the braces { and }, which are used to encode | and ؤ, must be replaced with square brackets viz. [and] respectively.

It is therefore recommended to use the Buckwalter ‘safe’ scheme.

Table 6 gives the Buckwalter equivalents that are currently used by `arabluatex`. The additional characters that are defined in table 5 on page 21 are also available.

Letter	Transliteration ²⁷		Buckwalter notation	
	dmg	loc	base/xml	safe
ا	<i>a</i>	<i>a</i>	A	A
ب	<i>b</i>	<i>b</i>	b	b
ت	<i>t</i>	<i>t</i>	t	t
ث	<i>ṭ</i>	<i>th</i>	v	v
ج	<i>ǧ</i>	<i>j</i>	j	j
ح	<i>ḥ</i>	<i>h</i>	H	H

²⁵See <http://www.qamus.org/transliteration.htm>

²⁶I am grateful to Graeme Andrews who suggested that the ‘safe’ scheme be included in `arabluatex`.

²⁷See section 6 on page 22.

Letter	Transliteration		Buckwalter notation	
	dmg	loc	base/xml	safe
خ	<i>ḫ</i>	<i>kh</i>	x	x
د	<i>d</i>	<i>d</i>	d	d
ذ	<i><u>d</u></i>	<i>dh</i>	*	V
ر	<i>r</i>	<i>r</i>	r	r
ز	<i>z</i>	<i>z</i>	z	z
س	<i>s</i>	<i>s</i>	s	s
ش	<i>š</i>	<i>sh</i>	\$	c
ص	<i>ṣ</i>	<i>ṣ</i>	S	S
ض	<i><u>ḍ</u></i>	<i><u>ḍ</u></i>	D	D
ط	<i><u>t</u></i>	<i><u>t</u></i>	T	T
ظ	<i><u>ẓ</u></i>	<i><u>ẓ</u></i>	Z	Z
ع	<i>c</i>	<i>‘</i>	E	E
غ	<i>ġ</i>	<i>gh</i>	g	g
ف	<i>f</i>	<i>f</i>	f	f
ق	<i>q</i>	<i>q</i>	q	q
ك	<i>k</i>	<i>k</i>	k	k
ل	<i>l</i>	<i>l</i>	l	l
م	<i>m</i>	<i>m</i>	m	m
ن	<i>n</i>	<i>n</i>	n	n
ه	<i>h</i>	<i>h</i>	h	h
و	<i>w</i>	<i>w</i>	w	w
ي	<i>y</i>	<i>y</i>	y	y
ى	<i><u>ā</u></i>	<i>á</i>	Y	Y
ة	<i>a</i>	<i>ah</i>	p	p
ء	◌◌	◌’	◌’	C
آ	◌◌ <i><u>ā</u></i>	◌’ <i><u>ā</u></i>		M
أ	◌◌	◌’	>	O
ؤ	◌◌	◌’	&	W
إ	◌◌	◌’	<	I
ئ	◌◌	◌’]	Q
ـ	—	—	~	~
أ	◌’	◌’	[L
ا	<i>a</i>	<i>a</i>	a	a
و	<i>u</i>	<i>u</i>	u	u
ي	<i>i</i>	<i>i</i>	i	i
ان	<i>an</i>	<i>an</i>	F	F
ون	<i>un</i>	<i>un</i>	N	N
ين	<i>in</i>	<i>in</i>	K	K
و	—	—	o	o

Letter	Transliteration		Buckwalter notation	
	dmg	loc	base/xml	safe
اَ	<i>ā</i>	<i>ā</i>	`	e
- (<i>taṭwīl</i>)	—	—	-	-

Table 6: Buckwalter scheme

Transliteration The Buckwalter notation can also be transliterated into any accepted romanization standard of Arabic. See above section 6 on page 22 for more information. However, it should be pointed out again that only accurate coding produces accurate transliteration. It is therefore at the very least highly advisable to use the hyphen for tying the definite article and the inseparable particles (viz. prepositions, adverbs and conjunctions) to words, like so:—

Al-EaAlamu الْعَالَمُ *al-^cālam^u*, Al-camsu الشَّمْسُ *aš-šams^u*, bi-SinaAEapi
Al-T~ib~i, بِصَنَاعَةِ الطِّبِّ *bi-ṣinā^catⁱ ṭ-ṭibbⁱ*.
wa-Al-l~ehi وَاللَّهِ *wa-^l-lāhⁱ*, Al-Hamdu li-l~ehi الْحَمْدُ لِلَّهِ *al-ḥamd^u li-llāhⁱ*.

Similary, it is not advisable to use | and [(‘base’ and ‘xml’ schemes) or M and L (‘safe’ scheme) to encode the ^ʔ*alif^u* ^l*-mamdūdātⁱ* and the ^ʔ*alif^u* ^l*-waṣṭⁱ* for such signs are supposed to be generated by `arabluatex` internal functions. Besides, as they do not *per se* convey any morphological information on what they are derived from, they cannot be transliterated accurately. To take one example, <ilY Al-LntiqaADi gives إِلَى الْأَنْتِقَاضِ as expected, but only <ilY Al-intiqaADi can be transliterated as ^ʔ*ilā* ^l*-intiḳāḏi* with the correct vowel ⟨i⟩ in place of the ^ʔ*alif^u* ^l*-waṣṭⁱ*.

8 L^AT_EX Commands in Arabic environments

General principle L^AT_EX commands are accepted in Arabic environments. The general principle which applies is that single-argument commands (`\command{⟨arg⟩}`) such as `\emph{⟨text⟩}`, `\textbf{⟨text⟩}` and the like, are assumed to have Arabic text as their arguments:—

`\abjad{45} kitAbu-hu \emph{fI 'l-\cap{`AdAt-i}}` مَكَّابُهُ فِي الْعَادَاتِ 45
kitābu-hu fi ^l-^cĀdātⁱ.²⁸

The same applies to footnotes:—

²⁸This is odd in Arabic script, but using such features as `\emph` or `\textbf` is a matter of personal taste.

```

1 \renewcommand{\footnoterule}%
2   {\hfill\noindent\rule[1mm]{.4\textwidth}{.15mm}}
3 \begin{arab}
4 'inna 'abI kAna mina 'l-muqAtilaT-i\footnote{al-muqAtilaT-i:
5 al-muqAtil-Ina.}, wa-kAnat 'ummI min `u.zamA'-i buyUt-i
6 'l-zamAzimaT-i\footnote{al-zamAzimaT-u: .tA'ifaT-u mina
7 'l-furs-i.}.
8 \end{arab}

```

إِنَّ أُنِي كَانَ مِنَ الْمُقَاتِلَةِ^a, وَكَانَتْ أُمِّي مِنْ عُظَمَاءِ بُيُوتِ الزَّمَازِمَةِ^b.

^aالمُقَاتِلَةُ: الْمُقَاتِلِينَ.
^bالزَّمَازِمَةُ: طَائِفَةٌ مِنَ الْفُرْسِ.

Some commands, however, do not expect running text in their arguments, or one may wish to insert English text eg. in footnotes or in marginal notes. `arabluatex` provides a set of commands to handle such cases.

`\LR{<arg>}` is designed to typeset its argument from left to right. It may be used in an Arabic environment, either `\arb{<Arabic text>}` or `\begin{arab} <Arabic text> \end{arab}`, for short insertions of left-to-right text, or to insert any `LATEX` command that would otherwise be rejected by `arabluatex`, such as commands the argument of which is expected to be a dimension or a unit of measurement.

`\RL{<arg>}` does the same as `\LR{<arg>}`, but typesets its argument from right to left. Even in an Arabic environment, this command may be useful.

For example, to distinguish words with a different color, one may proceed like so:—

```

1 \begin{arab}
2 _tumma "intalaqa _dU 'l-qarn-ayni 'il_A 'ummaT-iN 'u_hr_A fI
3 \LR{\textcolor{red}{\arb[fullvoc]{((ma.tli`-i 'l-^sams-i))}}}
4 wa-lA binA'-a la-hum yu'amminu-hum mina 'l-^sams-i.
5 \end{arab}

```

ثُمَّ اتَّLَقَ ذُو الْقَرْنَيْنِ إِلَى أُمَّةٍ أُخْرَى فِي ﴿مَطْلَعِ الشَّمْسِ﴾ وَلَا بِنَاءَ لَهُمْ يَوْمَهُم مِنَ الشَّمْسِ.

`\LRfootnote{<text>}` and `\RLfootnote{<text>}` typeset left-to-right and right-to-left footnotes respectively in Arabic environments. Unlike `\footnote{<text>}`, the arguments of both `\LRfootnote` and `\RLfootnote` are not expected to be Arabic text. For example, `\LRfootnote` may be used to insert English footnotes in running Arabic text:—

```

1 \arb[fullvoc]{\cap{z}ayd-uN\LRfootnote{%
2 \enquote{\arb[trans]{\cap{z}ayd} is the son of

```

```

3 \arb[trans]{\cap{`a}mr}}: the second
4 noun is not in apposition to the first, but forms
5 part of the predicate\ldots} "ibn-u \cap{`a}mr-iNU}

```

زيد^a ابن عمرو

^a “*zayd* is the son of ‘*amr*”: the second noun is not in apposition to the first, but forms part of the predicate...

When footnotes are typeset from right to left, it may happen that the numbers of the footnotes that are at the bottom of the page be typeset in the wrong direction. For example, instead of an expected number 18, one may get 81. `arabluatex` is not responsible for that, but should it happen, it may be necessary to redefine in the preamble the `LATEX` macro `\thefootnote` like so:—

```
\renewcommand*{\thefootnote}{\textsuperscript{\LR{\arabic{footnote}}}}
```

`\FixArbFtnmk`

Another solution is to put in the preamble, below the line that loads `arabluatex`, the command `\FixArbFtnmk`. However, for more control over the layout of footnotes marks, it is advisable to use the package `scrextend`.²⁹

`\LRmarginpar`

The command `\LRmarginpar` does for marginal notes the same as `\LRfootnote` does for footnotes. Of course, it is supposed to be used in Arabic environments. Note that `\marginpar` also works in Arabic environments, but it acts as any other single-argument command inserted in Arabic environments. The general principle laid on page 28 applies.

`\setRL`

`\setLR`

`\setRL` and `\setLR` may be used to change the direction of paragraphs, either from left to right or from right to left. As an example, an easy way to typeset a right-to-left sectional title follows:—

```

1 \setRL
2 \section*{\arb{barzawayhi li-buzurjumihra bn-i 'l-buxtikAni}}
3 \setLR
4 \begin{arab}
5 qAla barzawayhi bn-u 'azhar-a, ra's-u 'a.tibbA'-i fAris-a...
6 \end{arab}

```

بَرْزَوِيهِ لِبُزْرَجُمِهْرَ بْنِ الْبُخْتِكَانِ
قَالَ بَرْزَوِيهِ بْنُ أَزْهَرَ، رَأْسُ أَطِبَّاءِ فَارِسَ...

²⁹See <http://ctan.org/pkg/koma-script>; read the documentation of KOMA-script for details about the `\deffootnotemark` and `\deffootnote` commands.

8.1 csquotes

The recommended way of inserting quotation marks in running Arabic text is to use `csquotes`. With the help of the `\DeclareQuoteStyle` command, one can define an Arabic style, like so:—

```
1 \usepackage{csquotes}
2 \DeclareQuoteStyle{arabic}
3 {\rmfamily\textquotedblright}{\rmfamily\textquotedblleft}
4 {\rmfamily\textquoteright}{\rmfamily\textquoteleft}
```

Then, use this newly defined style with `\setquotestyle`, like so:—

```
1 \setquotestyle{arabic}
2 \begin{arab}
3   fa-qAla la-hu ju.hA: \enquote{\garIb-uN 'amru-ka yA .sadIqI
4     'a-tu.saddiqu 'l-.himAr-a wa-tuka_d_diba-nI?}
5 \end{arab}
6 \setquotestyle{english}
```

فَقَالَ لَهُ جُحَا: "غَرِيبُ أَمْرِكَ يَا صَدِيقِي أَتَصَدِّقُ الْحِمَارَ وَتَكْفِرُ بِي؟"

REM. Do not forget to set back the quoting style to its initial state once the Arabic environment is closed. See the last line in the code above.

8.2 reledmac

The two-arguments command `\edtext{<lemma>}{<commands>}` is supported inside `\begin{arab} ... \end{arab}`. As an example, one may get `arabluatex` and `reledmac` to work together like so:—

```
1 \beginnumbering
2 \pstart
3 \begin{arab}
4   wa-ya.sIru ta.hta 'l-jild-i
5   \edtext{\arb{.sadId-uN}}{\Afootnote{M: \arb{.sadId-aN} E1}}
6 \end{arab}
7 \pend
8 \endnumbering
```

9 Future work

A short, uncommented, list of what is planned in the versions of `arabluatex` to come follows:

- (a) Short-term:
 - i. Support for typesetting Arabic poetry.
 - ii. The *Qurʾān*: support for typesetting the *Qurʾān*.
 - iii. TEI xml support: arabluatex will interoperate with TEI xml through new global and local options that will output Arabic in a TEI xml compliant file in addition to the usual PDF output: see on page 3.
- (b) Medium-term:
 - i. More languages: the list of supported languages will eventually be the same as arabtex: see footnote 4 on page 4.
 - ii. Formulate propositions for extending the ArabT_EX notation and the transliteration tables. Include them in arabluatex. See section 4.8 on page 21.

10 Implementation

The most important part of arabluatex relies on Lua functions and tables. Read the .lua files that accompany arabluatex for more information.

```

1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{arabluatex}%
3 [2016/07/09 v1.4.1 ArabTeX-like interface for LuaLaTeX]
4 \RequirePackage{ifluatex}

```

arabluatex requires LuaL^AT_EX of course. Issue a warning if the document is processed with another engine.

```

5 \ifluatex\else
6 \PackageError{arabluatex}{lualatex needed}{%
7   Package `arabluatex' needs LuaTeX.\MessageBreak
8   So you should use `lualatex' to process your document.\MessageBreak
9   See documentation of `arabluatex' for further information.}%
10 \expandafter\expandafter\expandafter\csname endinput\endcsname
11 \fi

```

Declare the global options, and define them:

```

12 \DeclareOption{voc}{\def\al@mode{voc}}
13 \DeclareOption{fullvoc}{\def\al@mode{fullvoc}}
14 \DeclareOption{novoc}{\def\al@mode{novoc}}
15 \DeclareOption{trans}{\def\al@mode{trans}}
16 \ExecuteOptions{voc}
17 \ProcessOptions\relax
18 \def\al@mode@voc{voc}
19 \def\al@mode@fullvoc{fullvoc}
20 \def\al@mode@novoc{novoc}
21 \def\al@mode@trans{trans}

```

The following line will be used in the next release of arabluatex:

```

22 % \newif\ifal@mode@defined

```

Packages that are required by arabluatex:

```

23 \RequirePackage{fontspec}
24 \RequirePackage{amsmath}

```



```

25 \RequirePackage{etoolbox}
26 \RequirePackage{luacode}
27 \RequirePackage{xparse}
28 \RequirePackage{environ}

Here begins the real work: load arabluatex.lua:
29 \luadirect{dofile(kpse.find_file("arabluatex.lua"))}

This is needed by the current versions of polyglossia and luabidi. luabidi provides a
\Footnote command. Use it as well if it is loaded.
30 \luadirect{tex.enableprimitives("luatex",tex.extraprimitives("omega"))}

Font setup. If no Arabic font is selected, issue a warning message and attempt to
load the Amiri font which is included in TEXlive:
31 \AtBeginDocument{\ifdefined\arabicfont\relax\else
32 \PackageWarning{arabluatex}{\string\arabicfont\ is not defined.^^JI
33 will try to load Amiri}%
34 \newfontfamily\arabicfont[Script=Arabic]{Amiri}\fi}%

\setRL This neutralizes what is defined by the same command in luabidi:
35 \AtBeginDocument{\def\setRL{\pardir TRT\textdir TRT}}

\setLR The same applies to \setLR:
36 \AtBeginDocument{\def\setLR{\pardir TLT\textdir TLT}}

\LR This command typesets its argument from left to right. As \LR may be already
defined, we need to redefine for it to suit our purpose:
37 \AtBeginDocument{\ifdef{\LR}}%
38 {\RenewDocumentCommand{\LR}{m}{\bgroup\textdir TLT\rmfamily#1\egroup}}
39 {\NewDocumentCommand{\LR}{m}{\bgroup\textdir TLT\rmfamily#1\egroup}}

\RL This one typesets its argument from right to left. Same remark as above regarding
the need of redefinition.
40 \AtBeginDocument{\ifdef{\RL}}%
41 {\RenewDocumentCommand{\RL}{m}{\bgroup\textdir TRT\rmfamily#1\egroup}}
42 {\NewDocumentCommand{\RL}{m}{\bgroup\textdir TRT#1\rmfamily\egroup}}

\emph Arabic emphasis. Needs to be redefined as well.
43 \AtBeginDocument{\ifdef{\emph}}%
44 {\RenewDocumentCommand{\emph}{m}{\overline{\text{#1}}}}
45 {\NewDocumentCommand{\emph}{m}{\overline{\text{#1}}}}

\SetInputScheme arabluatex is designed for processing ArabTEX input notation. \SetInputScheme
may be used in the preamble or at any point of the document should the user wish
to use a different notation such as the ‘Buckwalter scheme’.
46 \def\al@input@scheme{arabtex}
47 \NewDocumentCommand{\SetInputScheme}{m}{\def\al@input@scheme{#1}}

```

`\SetArbEasy` By default, `arabluatex` applies complex rules to generate euphonic *tašdīd*, *ʿalif mamdūda* and *sukūn* depending on the modes which are selected, either `voc`, `fullvoc` or `trans`. Such refinements can be discarded with `\SetArbEasy`, either globally in the preamble or at any point of the document. Default complex rules can be set back at any point of the document with `\SetArbDflt`.

```

48 \def\al@arb@rules{dflt}
49 \NewDocumentCommand{\SetArbEasy}{}{\def\al@arb@rules{easy}}
50 \NewDocumentCommand{\SetArbDflt}{}{\def\al@arb@rules{dflt}}

```

`\SetTranslitFont` By default, the font that is used for transliterated text is the main font of the document. Any other font may also be selected with the font-selecting commands of the `fontspec` package.

```

51 \def\al@trans@font{\rmfamily}%
52 \NewDocumentCommand{\SetTranslitFont}{m}{\def\al@trans@font{#1}}

```

`\SetTranslitStyle` By default any transliterated Arabic text is printed in italics. This can be changed either globally in the preamble or at any point of the document:

```

53 \def\al@trans@style{itshape}%
54 \NewDocumentCommand{\SetTranslitStyle}{m}{\def\al@trans@style{#1}}

```

`\SetTranslitConvention` `\SetTranslitConvention{<convention>}` may be used to change the transliteration convention, which is `dmg` by default:

```

55 \def\al@trans@convention{dmg}
56 \NewDocumentCommand{\SetTranslitConvention}{m}{\def\al@trans@convention{#1}}

```

`\arbup` By default, `\arbup` is set to `\textsuperscript`. This is how the *tanwīn* that takes place at the end of a word should be displayed in `dmg` mode. `\NoArbUp` may be used either in the preamble or at any point of the document in case one wishes to have the *tanwīn* on the line. The default rule can be set back with `\ArbUpDflt` at any point of the document. Finally `\SetArbUp` may be used to customize the way *tanwīn* is displayed: this command takes the formatting directives as argument, like so: `\SetArbUp{<code>}`.

```

57 \NewDocumentCommand{\al@arbup@dflt}{m}{\textsuperscript{#1}}%
58 \NewDocumentCommand{\al@arbup}{m}{\al@arbup@dflt{#1}}
59 \NewDocumentCommand{\arbup}{m}{\al@arbup{#1}}
60 \NewDocumentCommand{\ArbUpDflt}{}{\let\al@arbup=\al@arbup@dflt}
61 \NewDocumentCommand{\NoArbUp}{}{\RenewDocumentCommand{\al@arbup}{m}{##1}}
62 \NewDocumentCommand{\SetArbUp}{m}{\RenewDocumentCommand{\al@arbup}{m}{#1}}

```

`\cap` Proper Arabic names or book titles should be passed to the command `\cap` so that they have their first letters uppercased. `\cap` is actually coded in Lua.

```

63 \DeclareDocumentCommand{\cap}{m}%
64 {\luairect{tex.sprint(cap(\luastringN{#1}))}}

```

`\txarb` `\txarb` sets the direction to right-to-left and selects the Arabic font. As it is supposed to be used internally by several Lua functions, this command is not documented, but available to the user should he wish to insert `utf8` Arabic text in his document.

`\txtrans` `\txtrans` is used internally by several Lua functions to insert transliterated Arabic text.

```

65 \DeclareDocumentCommand{\txarb}{+m}{\bgroup\textdir
66   TRT\arabicfont#1\egroup}
67 \DeclareDocumentCommand{\txtrans}{+m}{\bgroup\textdir
68   TLT\al@trans@font#1\egroup}

```

`\arb` The `\arb` command detects which Arabic mode is to be used, either globally if no option is set, or locally, then passes its argument to the appropriate Lua function.

```

69 \DeclareDocumentCommand{\arb}{0{\al@mode} +m}%
70 {\edef\@tempa{#1}%
71   \ifx\@tempa\al@mode@voc%
72     \bgroup\textdir TRT\arabicfont%
73     \luadirect{tex.sprint(processvoc(\luastringN{#2},
74       \luastring0{\al@arb@rules}, \luastring0{\al@input@scheme}))}\egroup%
75   \else%
76     \ifx\@tempa\al@mode@fullvoc%
77       \bgroup\textdir TRT\arabicfont%
78       \luadirect{tex.sprint(processfullvoc(\luastringN{#2},
79         \luastring0{\al@arb@rules}, \luastring0{\al@input@scheme}))}\egroup%
80     \else%
81       \ifx\@tempa\al@mode@novoc%
82         \bgroup\textdir TRT\arabicfont%
83         \luadirect{tex.sprint(processnovoc(\luastringN{#2},
84           \luastring0{\al@arb@rules}, \luastring0{\al@input@scheme}))}\egroup%
85       \else%
86         \ifx\@tempa\al@mode@trans%
87           \bgroup\textdir TLT\al@trans@style%
88           \luadirect{tex.sprint(processstrans(\luastringN{#2},
89             \luastring0{\al@trans@convention},
90             \luastring0{\al@arb@rules},
91             \luastring0{\al@input@scheme}))}\egroup%
92         \else%
93           \fi\fi\fi\fi}

```

`arab` The `arab` environment does for paragraphs the same as `\arb` does for short insertions of Arabic text.

```

94 \NewEnviron{arab}[1][\al@mode]%
95 {\par\edef\@tempa{#1}%
96   \ifx\@tempa\al@mode@voc%
97     \bgroup\pardir TRT\textdir TRT\arabicfont%
98     \luadirect{tex.sprint(processvoc(\luastring0{\BODY},
99       \luastring0{\al@arb@rules}, \luastring0{\al@input@scheme}))}\egroup%
100   \else%
101     \ifx\@tempa\al@mode@fullvoc%
102       \bgroup\pardir TRT\textdir TRT\arabicfont%
103       \luadirect{tex.sprint(processfullvoc(\luastring0{\BODY},
104         \luastring0{\al@arb@rules}, \luastring0{\al@input@scheme}))}\egroup%
105     \else%

```

```

106 \ifx\@tempa\al@mode@novoc%
107 \bgroup\pdir TRT\textdir TRT\arabicfont%
108 \luadirect{tex.sprint(processnovoc(\luastring0{\BODY},
109 \luastring0{\al@arb@rules}, \luastring0{\al@input@scheme}}))}\egroup%
110 \else \ifx\@tempa\al@mode@trans%
111 \bgroup\pdir TLT\textdir TLT\al@trans@style%
112 \luadirect{tex.sprint(processtrans(\luastring0{\BODY},
113 \luastring0{\al@trans@convention},
114 \luastring0{\al@arb@rules},
115 \luastring0{\al@input@scheme}}))}\egroup%
116 \else \fi\fi\fi\fi}\par]

```

`\abjad` `\abjad{⟨number⟩}` expresses its argument in Arabic letters in accordance with the *ʿabǧād* arrangement of the alphabet. `⟨number⟩` must be between 1 and 1999. It is now coded in Lua so that polyglossia is no longer needed. See `arabluatex.lua` for more information.

```

117 \AtBeginDocument{%
118 \ifdefined\abjad%
119 \RenewDocumentCommand{\abjad}{m}%
120 {\luadirect{tex.sprint(abjadify(#1))}}%
121 \else%
122 \NewDocumentCommand{\abjad}{m}%
123 {\luadirect{tex.sprint(abjadify(#1))}}
124 \fi}

```

`\LRmarginpar` `\LRmarginpar` is supposed to be inserted in an Arabic environment. It typesets his argument in a marginal note from left to right.

```

125 \DeclareDocumentCommand{\LRmarginpar}{m}{\marginpar{\textdir TLT #1}}

```

`\LRfootnote` `\LRfootnote` and `\RLfootnote` are supposed to be used in Arabic environments for insertions of non Arabic text. `\LRfootnote` typesets its argument left-to-right...

`\RLfootnote` while `\RLfootnote` typesets its argument left-to-right.

```

126 \DeclareDocumentCommand{\LRfootnote}{m}{\bgroup\pdir
127 TLT\LR{\footnote{#1}}\egroup}
128 \DeclareDocumentCommand{\RLfootnote}{m}{\bgroup\pdir
129 TRT\LR{\footnote{#1}}\egroup}

```

`\FixArbFtnmk` In the preamble, just below `\usepackage{arabluatex}`, `\FixArbFtnmk` may be of some help in case the footnote numbers at the bottom of the page are printed in the wrong direction. This quick fix uses and loads `scrextend` if it is not already loaded.

```

130 \NewDocumentCommand{\FixArbFtnmk}{}{%
131 \@ifpackageloaded{scrextend}%
132 {\AtBeginDocument{\deffootnote{2em}{1.6em}{\LR{\thefootnotemark}.\enskip}}}%
133 {\RequirePackage{scrextend}}
134 \AtBeginDocument{\deffootnote{2em}{1.6em}{\LR{\thefootnotemark}.\enskip}}}

```

That is it. Say goodbye before leaving.

```

135 \endinput

```

References

- Brockelmann, Carl et al. (1935). “Die Transliteration der arabischen Schrift in ihrer Anwendung auf die Hauptliteratursprachen der islamischen Welt”. In: *Denkschrift dem 19. internationalen Orientalistenkongreß in Rom*. In collaboration with Ph. S. van Ronkel and Otto Spies. Deutschen Morgenländischen Gesellschaft. Leipzig: Deutschen Morgenländischen Gesellschaft, in Kommission bei F. A. Brockhaus. URL: http://www.naher-osten.uni-muenchen.de/studium_lehre/werkzeugkasten/dmgtransliteration.pdf.
- Habash, Nizar Y. (2010). *Introduction to Arabic Natural Language Processing*. Synthesis Lectures on Human Language Technologies 10. Toronto: Morgan & Claypool Publishers.
- Hosny, Khaled (2015). *Amiri*. URL: <http://www.amirifont.org/>.
- Lagally, Klaus (2004). *ArabTeX. Typesetting Arabic and Hebrew*. User Manual Version 4.00. Version 4.00. URL: <http://mirrors.ctan.org/language/arabic/arabtex/doc/html/arabtex.htm>.
- Lane, Edward William (1863–1893). *An Arabic-English lexicon*. 8 vols. London – Edinburgh: Williams and Norgate.
- Wright, W. LL.D (1896). *A Grammar of the Arabic Language*. Rev. by W. Robertson Smith and M. J. de Goeje. With a forew. by Pierre Cachia. 3rd ed. 2 vols. Beirut: Librairie du Liban.

Change History

v1.0	General: Initial release	1	v1.3	<code>\arbut</code> : $\text{ʿi}^c\text{rāb}$ is now written as superscript text in <code>dmg</code> mode by default.	34
v1.0.1	General: Minor update of the documentation	1	v1.4	<code>\SetInputScheme</code> : <code>\SetInputScheme</code> may be used to process other input schemes such as ‘Buckwalter’	33
v1.1	<code>\abjad</code> : New and more flexible <code>\abjad</code> command.	36		<code>\SetTranslitFont</code> : For selecting a specific font for transliterated texts	34
v1.2	<code>\SetArbEasy</code> : New <code>\SetArbEasy/\SetArbDflt</code> for ‘modern’ or ‘classic’ Arabic styles.	34			

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols	<i>Arabic-English Lexicon,</i>	G
<code>\@ifpackageloaded</code> . 131	<i>An</i> 16	<i>Grammar of the Arabic</i>
<code>\@tempa</code> 70, 71,	<code>\arabicfont</code> 31,	<i>Language, A</i> . .
76, 81, 86, 95,	32, 34, 66, 72,	. 4–6, 8, 11–16, 21
96, 101, 106, 110	77, 82, 97, 102, 107	
	arabluatex (package) .	H
<code>_</code> 32	. . . 2–7, 9–11,	Habash, Nizar Y. 26
	13–16, 18,	Hosny, Khaled 4, 5
	21, 23–26, 28–34	
A	arabtex (package) . . .	I
<code>\abjad</code> 21, 117	. . . 3, 4, 14, 18, 32	<code>\ifal@mode@defined</code> . 22
<code>\aemph</code> 22, 43	arabulatex (package) . . 14	<code>\ifluatex</code> 5
<code>\al@arb@rules</code>	arabxetex (package) . 4, 14	<i>Introduction to Arabic</i>
. . . 48, 49, 50,	<code>\arb</code> 6, 69	<i>Natural Lan-</i>
74, 79, 84, 90,	<code>\arbup</code> 24, 57	<i>guage Process-</i>
99, 104, 109, 114	<code>\ArbUpDflt</code> 24, 57	<i>ing</i> 26
<code>\al@arbup</code>		<code>\itshape</code> 53
. 58, 59, 60, 61, 62	B	K
<code>\al@arbup@dflt</code>	<code>\BODY</code> . 98, 103, 108, 112	KOMA-script (package) 30
. 57, 58, 60	Brockelmann, Carl . 23, 24	L
<code>\al@input@scheme</code>	C	Lagally, Klaus 3, 18
. 46, 47,	<code>\cap</code> 23, 63	Lane, Edward William . 16
74, 79, 84, 91,	csquotes (package) . . . 31	<code>\LR</code> 29, 37,
99, 104, 109, 115		127, 129, 132, 134
<code>\al@mode</code> 12,	D	<code>\LRfootnote</code> . . . 29, 126
13, 14, 15, 69, 94	<code>\DeclareOption</code>	<code>\LRmarginpar</code> . . . 30, 125
<code>\al@mode@fullvoc</code> 12, 13, 14, 15	luabidi (package) . . . 4, 33
. 19, 76, 101	<code>\deffootnote</code> . . 132, 134	<code>\luadirect</code> 29,
<code>\al@mode@novoc</code>	E	30, 64, 73, 78,
. 20, 81, 106	environments:	83, 88, 98, 103,
<code>\al@mode@trans</code>	arab 6, 94	108, 112, 120, 123
. 21, 86, 110	<code>\ExecuteOptions</code> 16	luaotfload (package) . . . 23
<code>\al@mode@voc</code> . 18, 71, 96	F	<code>\luastringN</code>
<code>\al@trans@convention</code>	<code>\FixArbFtnmk</code> . . . 30, 130	. 64, 73, 78, 83, 88
. . . 55, 56, 89, 113	fontspec (package) . . .	<code>\luastringO</code> 74,
<code>\al@trans@font</code> 4, 23, 34	79, 84, 89, 90,
. 51, 52, 68	<code>\footnote</code> 127, 129	91, 98, 99, 103,
<code>\al@trans@style</code>	fullvoc (option) 5, 7	104, 108, 109,
. 53, 54, 87, 111		112, 113, 114, 115
<i>Amiri</i> 4, 5		M
amiri (package) 5		<code>\marginpar</code> 125
arab (environment) . 6, 94		
<i>ArabTeX</i> 3, 18		

N		R		T	
<code>\NewEnviron</code>	94	<code>reledmac (package)</code>	31	<code>\text</code>	44, 45
<code>\newfontfamily</code>	34	<code>\RequirePackage</code>	...	<code>\textdir</code>	35,
<code>\NoArbUp</code>	24, 57	4, 23, 24,		36, 38, 39, 41,
<code>novoc (option)</code>	5, 7		25, 26, 27, 28, 133		42, 65, 67, 72,
O		<code>\RL</code>	29, 40		77, 82, 87, 97,
options:		<code>\RLfootnote</code>	29, 126		102, 107, 111, 125
<code>fullvoc</code>	5, 7	<code>\rmfamily</code>	<code>\textsuperscript</code>	57
<code>novoc</code>	5, 7	.	38, 39, 41, 42, 51	<code>\thefootnotemark</code>	..
<code>trans</code>	5, 7	S		132, 134
<code>voc</code>	5, 7	<code>scrextend (package)</code>	30, 36	<code>trans (option)</code>	5, 7
<code>\overline</code>	44, 45	<code>\SetArbDflt</code>	6, 48	<i>Transliteration der ara-</i>	
P		<code>\SetArbEasy</code>	6, 48	<i>bischen Schrift,</i>	
<code>\par</code>	95, 116	<code>\SetArbUp</code>	24, 57	<i>Die</i>	23, 24
<code>\pardir</code>	<code>\SetInputScheme</code>	26, 46	<code>\txarb</code>	65
	35, 36, 97, 102,	<code>\setLR</code>	30, 36	<code>\txtrans</code>	65
	107, 111, 126, 128	<code>\setRL</code>	30, 35	V	
<code>polyglossia (package)</code>	..	<code>\SetTranslitConvention</code>	<code>voc (option)</code>	5, 7
.....	4, 33, 36	23, 55	W	
<code>\ProcessOptions</code>	17	<code>\SetTranslitFont</code>	23, 51	<code>Wright, W. LL.D</code>	...
<code>\ProvidesPackage</code>	2	<code>\SetTranslitStyle</code>	23, 53	.	4–6, 8, 11–16, 21