نظام لاتخ ﴿ العربي ﴾

منة الباري ﷻ في الكتابة بالخط العربي

عربي و فارسی

Typesetting Arabic and Farsi with the Arabi package
The Users Guide

يوسف الجابري
© YOUSSEF JABRI

École nationale des sciences appliquées,
Box 696, Oujda, Morocco
yjabri@ensa.univ-oujda.ac.ma

Version 1.1, December 16, 2006

Contents

List of Figures

List of Tables

بسم الله الرحمن الرحيم

قل لن يصيبنا إلا ما
كتب الله لنا هو مولانا
وعلى الله فليتوكل المؤمنون

كان من دعاء مُحَمَّد رسول الله ﷺ :

اللهم أغنني بالعلم ، وزيني بالحلم ،
وأكرمني بالتقوى ، وجملني بالعافية .

أمنن على من شئت تكن أميره،
واستغن عمن شئت تكن نظيره،
واحتج إلى من شئت تكن أسيره!

الإمام علي بن أبي طالب
كرم الله وجهه

Dedicated to *All* the people who need it.

Abstract

The Arabi package provides the **Right-to-Left scripts** support for TEX without the need of any external preprocessor. The *Bi-directional capability* supposes that the user has a TEX engine that knows the four primitives `\beginR`, `\endR`, `\beginL` and `\endL`. That is the case of the TEX--XET and ε-TEX engines and all their extensions/variants like XETEX and `Alpha`[1]. Arabi is fully compliant with the BABEL system and hence a standard switching mechanism to chose the supported languages. We intend to add some important multilingual typesetting capabilities not yet supported by BABEL. It comes with many GNU Arabic and Farsi good quality fonts and may of course also use the Arabic fonts of commercial manufacturers. A *limited* support is already provided to use some widely used Arabic and Farsi fonts that come with such systems. The package is distributed under the LATEX Project Public License (LPPL), and has the LPPL maintenance status "author-maintained." It can be used *freely* (including commercially) to produce beautiful texts that mix Arabic, Farsi and Latin (or other type) characters.
You can simply type the text you want in 8-bit what you see is what you get text in many input encodings (actually, CP 1256, ISO 8859-6 and Unicode UTF-8 are supported) It can also typeset classical Arabic poetry, and has a limited, but still useful, capability of vocalizing. It is LATEX 2ε and BABEL compliant!
It has even an experimental module of transliteration!

[1] Arabi has not been tested with `Alpha` yet.

بسم الله الرحمن الرحيم

تقديم مقتضب لرزمة «العربي»

رزمة العربي نظام يتيح إمكانية استعمال الحروف العربية واللاتينية جنباً إلى جنب في مستند واحد باستعمال نظام « تيخ » TEX لتصفيف الحروف الذي ابتكره الأستاذ « دونالد كنوث » من جامعة ستانفورد من أجل تصفيفِ المستندات العلمية (كالتي تحتوي على الرياضيات والفيزياء والكيمياء، على سبيل المثال لا الحصر) ووضعه تحت تصرف المجتمع العلمي دون مقابل .

رزمة العربي مساهمة متواضعة من أجل إضافة إمكانية استعمال اللغتين (عربي و فارسي) مع نظام تيخ لاستغلال كل ما يحتويه هذا العالم الزاخر من إمكانيات في تصفيف النصوص، والتي جعلت منه اختيار المؤسسات والهيئات العالمية التي تقوم بنشر البحوث العلمية في العالم بأسره .

ومنذ البداية ، فهذا النظام يتميز بكونه محمولا ويتمتع بقدر كبير من المرونة ، لأنه قابل للاستعمال مع معظم ما تم إنجازه من إضافات وبرامج مساعدة وما أكثرها ! إضافة إلى أنه لا يحتاج إلى أي معالج خارجي لتحديد أشكال الحروف في الكلمة .

يقدمُ العربي حالياً مصحوباً بمجموعة زاخرة من الخطوط العربية والفارسية ، حرة الاستعمال كما يمكنه استعمال عدد من الخطوط التي تأتي مع نظام ويندوز مثلاً .

كما هو الحال بالنسبة لنظام تيخ ، فإن العربي مجاني ولا يكلف مُستَعمِله إلا عناء الاستعمال ،

والله المستعان

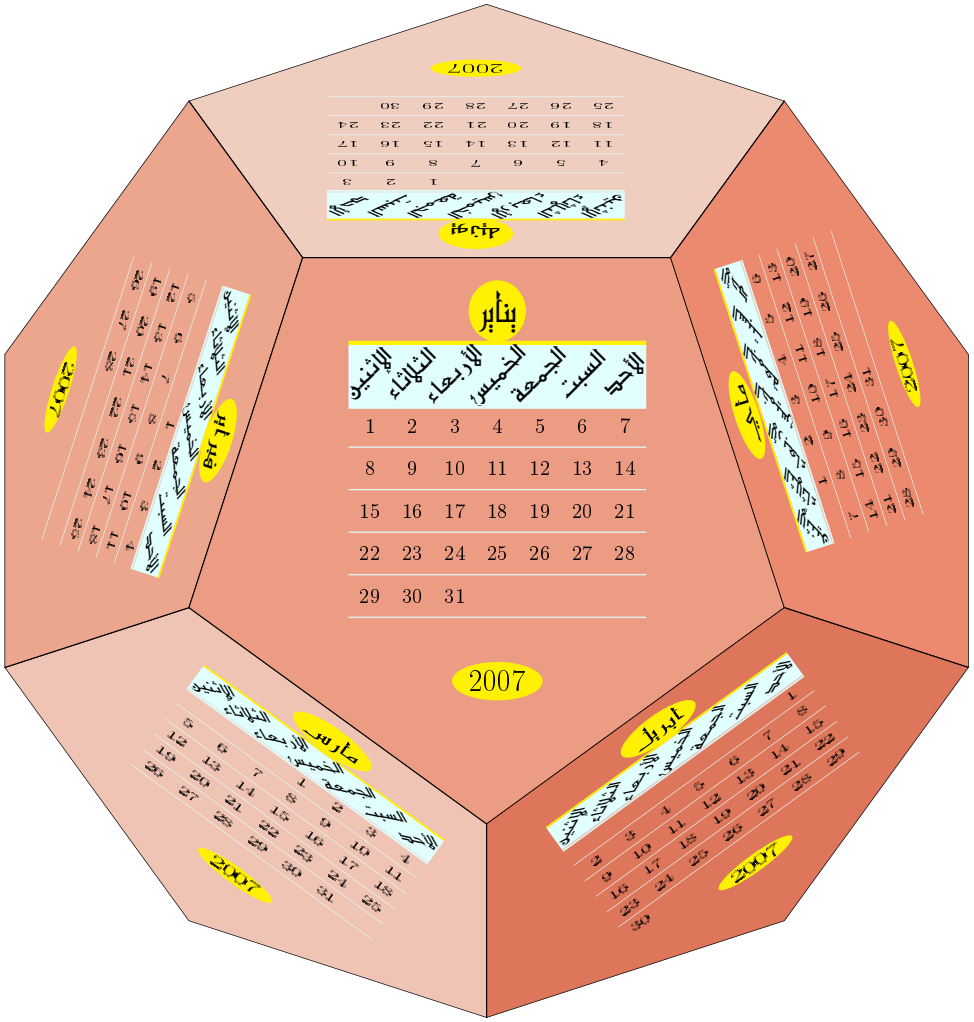


Figure 0.1: Arabi and the `calendrierfp` package

CHAPTER 1

Preface to Version 1.1
تقديم للإصدار ١.١

In this version of Arabi, we introduce some new features and corrections to version 1.0.

Actually, Arabi is a part of the MikTEX and BakomaTEX distributions, we could only test the version that comes with MikTEX which is free, BakomaTEX is a shareware.

Among the new things with this version we may mention among other things the possibility to search and copy PDF files created with this version of Arabi when you use the pdftex engine. This works quite well. This led us to rewrite the Arabi documentation to remove all direct use of PSTricks and replace it with pgf. We were nicely surprised that Arabi is *totally* compatible with pgf as we didn't get any problem. So, with this version you can search and copy Arabic and Farsi texts from the Arabi documentation. This works with *any* font that is used with Arabi. The CMap translations files we created to support the cmap package.

An other new feature, still experimental but which seems to work too but we did not test a lot since it's rather new is the possibility to get HTML files from Arabi input files using the TEX4ht program.

It's possible to get Arabic characters in PDF files bookmarks in a Latin document that uses Arabic (and Farsi of course). Just get a look at this document bookmarks! Hyperref is not yet supported, we get some problems when the main direction of the text is Right-to-Left. This has to do possibly with the way we handle \everypar, and the table of contents.

There are also some changes in the way to call Arabic, Farsi and especially Latin small texts insertions in a paragraph where the opposite direction dominates. For example, the command \L is already defined to get the Polish character Ł. So Polish people and those who write Polish would not be able to use the Arabi, as we were told in the TUG 2006 conference.

For a quick look at some of the changes and additions brought by this version, especially for users who read already the versions 1.0 documentation, you may skip reading the entire documentation and get a look at Appendix C entitles *Changes* on page 75. Nevertheless, for the details of the changes and the new features, it is still necessary to read this guide.

Happy Arabi TEXing!!

CHAPTER 2

Presentation
مقدمة

من يرحم يرحم ، ومن يصمت يسلم ، ومن يجهل يغلب ، ومن يعجل يخطئ ، ومن يحرص على الشر لا يسلم ، ومن لا يدع المراء يشتم ، ومن لا يكره الشتم يأثم ، ومن يكره الشر يعصم ، ومن يتبع وصية الله يحفظ ، ومن يحذر الله يأمن ، ومن يتولى الله يتمنع ، ومن لا يسأل الله يفتقر ، ومن لا يكن مع الله يخذل ، ومن يستعن بالله يظفر .

This document descries the Arabi System version 1.1, نظام لاتيخ العربي[1], an author-maintained LPPL package for LaTeX 2ε that provides full support of *Arabic* عربي and *Farsi* **فارسی** for Johannes Braams BABEL system. It typesets *Arabic text*[2] in addition to *all* what can TeX and LaTeX do.

> In the sequel the word Arabic may be used to refer to both Arabic and Farsi. The distinction would be made only if something specific to one the two "languages" should be mentioned.

Our system does not need/use any preprocessor, and is certainly compatible with most available packages because we tried to shorten TeX `coding` to deal with the specific stuff of the Arabic script as much as we could to avoid eventual conflicts. We tried to encapsulate all the system intelligence (we are essentially meaning *contextual analysis* to detect shapes of characters[3]) in the fonts. While we rely on the `\beginR` and `\endR` primitives for the bi-directional support, to write from Right-to-Left and Left-to-Right. Thus, at the end, the necessary TeX code needed is similar to that of any Latin system for TeX and is dealt with as such by the TeX engine.

An other important advantage of our system is that in fact it's also *compatible* with *all* ***other formats***, like plain TeX and ConTeXt for example. You may get a look at the two sample files provided, using plain and ConTeXt. This happens easily just because the whole contextual analysis is done in the fonts! But since, for the moment at least, we use the LaTeX format, we felt that we should prepare our system for use with LaTeX.

One nice development would be the writing of the necessary macros for its use with the ConTeXt format!

---

[1] The name TeX as all TeX users know come from $\tau\epsilon\chi$, the first three letters of the Greek word whose equivalent in English is Technology. The sound equivalent to the Greek χ is exactly the sound of the letter *ẖāʾ* ، خاء in Arabic.

[2] With opposition to the Arabic language, our system should be able to typeset any language that uses the *Arabic script* and some other scripts written from Right-to-Left. Support of the *Farsi* language is provided since version 1.0. Some other languages using the Arabic script should also be implemented.

[3] As explained farther, Arabic (and Farsi, of course) characters' shapes vary according to the context, in particular their forms vary according to their positions in a word. In general, in many earlier tentatives, an external preprocessor was used to do the contextual analysis and compute all the needed ligatures before feeding TeX with the result.

2.1. What you need to have to use Arabi

In order to use Arabi within TEX and/or LATEX,

1. You have to use a TEX program with Right-to-Left capabilities. That is an "engine" that knows the primitives `\beginL`, `\endL`, `\beginR` and `\endR`. In this category we do get programs like TEX--XET, ε-TEX (ε-LATEX).
   For example, we cite among the possible ones:
   - teTEX for UNIX/Linux comes with ε-TEX,
   - NTEX for UNIX with TEX--XET,
   - `Web2C` for Windows, this is even the default TEX program in MikTEX,
   - DirectTEX for Older Mac OS systems has TEX--XET built-in,
   - XETEX (Mac OS X, Linux and Windows), is an extended ε-TEX system. This one is particularly interesting, see below Chapter 8.6 in preparation for the details!
   - TEX Live CD-ROMs distributed by the TEX Users Group (TUG).
2. You should also have BABEL, this is certainly the case if your distribution is not too much old. Otherwise you will have to download it from some `CTAN`[4]. We think that we have for the moment a system running with enough Right-to-Left text formatting capabilities to be used quite efficiently. We did not want to loose time rewriting a code that exists and is used widely.

2.2. How this document is organized

In the next chapter, we recall succinctly some Arabic script specificities. We focus our attention especially on what may affect its typesetting with TEX. In Chapter 3, we describe the way our system should be used. The input description and basic commands to get the Arabic script, vowelized or not, are given with some examples. In Chapter 4, we describe the bi-directional sectioning commands of Arabi. We also show how appear the different lists environments and footnotes in an Arabic context. In Chapter 5, we describe some Arabic features available with this version 1.1 of Arabi, like the different fonts distributed with the system that can be used from now on. The *experimental* transliteration module and some important packages that cooperate fully with Arabi without any problem!

[4] At the time of writing this documentation, we use `babel.sty` version 3.8a of 2004/19/02.

هيا بنا !

CHAPTER 3

The Arabic Script
خصوصيات الخط العربي

قال حافظ إبراهيم:

رجعت لنفسي فاتهمت حصاتـــــي ونـاديت قومي فاحتسبت حياتـــــي

رموني بعقم في الشباب وليتنـــــي عقمت، فلم أجزع لقول عداتـــــي

أنا البحر في أحشائه الدر كامـــــن فهل سألوا الغواص عن صدفاتـــــي

The name of this package should not be misunderstood. It is not designed to support the only Arabic language, but all languages that use the Arabic script. Technically speaking, for BABEL, they will all be considered as *dialects* of Arabic.

3.1. The Arabic script

The Arabic script is one of the most used scripts on earth. This is the case also for Latin and Chinese. It dominates in the Arabic countries, but has a special place for all Muslims because it's the script used to write the Koran, the holy book of Muslims.

The Arabic script, like all other Semitic languages, is written from *Right-to-Left*.

It evolved from the *Nabataean Aramaic* script. It has been used since the 4th century AD, but the earliest document, an inscription in Arabic, Syriac and Greek, dates from 512 AD. During the 7th century, dots were added to existing letters that shared the same glyphs in order to avoid ambiguities. Further diacritics indicating short vowels were introduced later, but they are generally used only in some particular documents as we will see later.

Many *languages* are written with, or used to be written with the Arabic script. This includes: Afrikaans, Albanian, Arabic, Azeri, Baluchi, Berber, Bashkir, Belarussian, Bosnian, Chaghatai, Chechen, Comorian, Fulani, Hausa, Kashmiri, Kazakh, Kurdish, Kyrghyz, Malay, Mandinka, Morisco, Mozarabic, Nubian, Pashto, Persian/Farsi, Punjabi, Sanskrit, Sindhi, Somali, Songhay, Swahili, Tamazight, Tatar, Turkish, Turkmen, Urdu, Uyghur, Uzbek and Wolof.

One important specificity of the Arabic script is that *no hyphenation* is needed or allowed at all. In very old Arabic documents the words could be splitted after a non connecting character while characters that connect never get splitted. In modern Arabic script hyphenation is forbidden completely. That makes it more difficult to get justification when long words occur at the end of a line but the Arabic is cursive and has (in modern fonts to mimic the hand writing) a special character called *keshida* (*keshideh* or *tatweel* a Farsi word that means *stretch*) that may be used between joining characters to make the word become longer makes it easier hence justifies

the lines with keshidas and not spaces as in Latin texts. An example is the following same word مثال that may occupy longer مثال and longer مثـال and much more longer مثـــال spaces. So at the end, the good news are that ***no*** *hyphenation patterns* are needed for all languages that use the Arabic script.

3.2. The Arabic language

Arabic is a Semitic language with much more than 200 million speakers in Algeria, Bahrain, Chad, Djibouti, Egypt, Eritrea, Iran, Iraq, Jordan, Kuwait, Lebanon, Libya, Mali, Mauritania, Morocco, Niger, Oman, Palestine, Qatar, Saudi Arabia, Somalia, Sudan, Syria, Tunisia, United Arab Emirates, Uzbekistan and Yemen among other countries.

The Arabic script is particularly suited to the Arabic language. It is constituted in its basic form by 28 *consonants* (29 if we count the *hamza*), and 6 *vowels*:

- 3 short vowels (Vowel diacritics) *fatha*, *damma* and *kasra*  that are written like the accents in the Latin script. The first two vowels (from the right) appear above letters while the third one is put below letters.
- and three long vowels represented by the three Arabic consonants *alif*, *waw* and *yāa* 

The short vowels are written like accents in the Latin script, above or under consonants. They are in general used only in religious books or in specialized books that treat of Arabic grammar or eventually for beginners who are learning Arabic. Sometimes the diacritics are used for decorative purposes in book titles, letterheads, nameplates, etc.

Being an `abjad`, the expression "Arabic alphabet" is often used to mean Arabic consonants.

3.3. The Arabic alphabet

The main two problems[1] faced when trying to typeset the Arabic script with computers and in particular with TEX are:

1. Like all other Semitic languages, the Arabic script is written from *Right-to-Left*. This is solved in TEX by using an engine that is able to typeset from Right-to-Left and from Left-to-Right like ε-TEX or TEX--XHT.
2. In Arabic, letters forms (understand *glyphs*) depend on their position in the word and on the adjacent characters. We must be able to do a *contextual analysis* to determine the right shape of the character before calling the right glyph that represents it. This was done before either using a preprocessor written in some programming language like `C`, for example, or even using TEX itself as within the ArabTEX system of K. Lagally.

   For example, the 4 different forms (4 different glyphs for the same character) depending on its position in a word (*initial*, *medial*, *final* and *isolated*) for the following two

[1] They are inherent to the Arabic script!

characters look like:

isolated	initial	medial	final
ب	بـ	ـبـ	ـب
ج	جـ	ـجـ	ـج

Thus, a three letter word, for example, will start with a letter in initial form, followed by a letter in medial form and, finally, by a letter in final form like:

كتب

instead of

ﻛﺘﺐ

But the reality is even worse since even in the middle of a word, a character may have the final or the initial form like in:

عربي

because some characters do not connect to any character that comes after. They have only two forms (*isolated* (which is also used as *initial*) and *final* (also used as *middle*))! **The contextual rules of the Arabic script are independent of the language, font and style and have no exception!**

In addition to the *contextual ligatures*, just seen above, there are some *linguistic ligatures* that are *mandatory* in Arabic like the *lāmalif*:

we write لا and never ﻻ for ﻟﺎ

or the second part of the word *āllāh* الله

we write الله and never ﷲ for ﷲ

The word Allah الله for *God*.

There is also a huge number of optional aesthetic ligatures that increase the number of possible glyphs for Arabic (that contains less than 30 characters) to more than 1000 glyphs for the *naskhi* font for example. This is the standard font used since the beginning of the 20th century in published Arabic literature and in journals!

For the actual version of Arabi, we do not support aesthetic ligatures just because the fonts that can be used with TEX should not exceed 256 characters. Otherwise, even for this version, we would get all the possible glyphs that exist in the Postscript or True type fonts used. Nevertheless, in future versions, a limited number of such ligatures will be added for the fonts that contain such glyphs.

3.4. The Persian language(s) "Farsi"

Persian is an Indo-European language, spoken in Iran and several neighboring countries (Tajikistan, Afghanistan, Uzbekistan, and Bahrain). Prior to British colonization, Persian was also widely used as a second language in the Indian subcontinent; it took prominence as the language of culture and education in several Muslim courts in the subcontinent throughout the Middle Ages and became the official court language under the *Mughol* emperors. The three major variations of the Persian language are *Dari*, the official language of Afghanistan, *Farsi* the official language of Iran, and *Tajik* the official language of Tajikistan. In the Persian language itself, the name of the language is *Farsi*. It counts about 61–71 million native speakers, and around 110 million in total.

After the conversion to Islam, it took approximately one hundred fifty years to adopt the Arabic alphabet as a replacement for the older alphabet. Previously, two different alphabets were used for the Persian language: one was also called *Pahlavi* and was a modified version of the *Aramaic alphabet*, and the other was a native Iranian alphabet called *Dîndapirak*.

However, the script used actually is borrowed from Arabic. The adaptation propagated to Pakistan, Afghanistan, India, China, Malaysia, and Java. The Arabic alphabet was extended even more, each language adding letters and making modifications as needed: from 29 Arabic letters to more than 140 letters (141 in the Unicode Standard Version 3.0) in modern use.

Farsi is equivalent in many heads to a *Nastaliq font* خط نستعلیق, that was invented in 15th century, and few more characters than the usual Arabic alphabet. In fact Farsi's typography is much more. See the next section for some details. We give now an example of a Farsi text for illustration and testing only (This the first part of the *The Hunter-Student*, a story from کلیله و دمنه).

(It is a text we found on the net, from Persian studies web site of Texas University, the author does not speak or understand Farsi!).

شکارچی دانش‌آموز، از قصه‌های کلیله و دمنه

روزی بود و روزگاری بود. در زمان قدیم یک شکارچی بود که بعضی از روزها در بیابان کبک‌ها و کبوترهای صحرائی را شکار می‌کرد و بعض روزها در کنار دریا ماهی صید می‌کرد و با این کار زندگی خود و زن و بچه‌اش را روبه‌راه می‌کرد.

یک روز این آقای شکارچی در گوشه‌ای از بیابان کنار یک تپه قدری گندم و برنج و ارزن پاشیده بود و دام، یعنی تور مخصوص شکار را روی آن آماده کرده بود و خودش سر نخ آن را گرفته بود و در پشت تپه پنهان شده بود – به قول معروف در کمین نشسته بود – و منتظر بود که کبوترهائی که در آن نزدیکی می‌چریدند به دام او بیفتند.

پس از انتظار زیاد که سه تا از کبوترها به دام نزدیک شده بودند، ناگهان شکارچی از پشت سر خود صدای داد و فریاد دو نفر را شنید که داشتند نزدیک می‌شدند و با صدای بلند باهم گفتگو می‌کردند. شکارچی از ترس اینکه کبوترها رم کنند و به دام نیفتند فوری خود را به آن دو نفر رسانید و گفت: »آقایان، محض رضای خدا در اینجا داد و فریاد نکنید تا مرغ‌های من نترسند و فرار نکنند.«

آن دو نفر که »طلبه« بودند، یعنی دانش‌جویان مدرسه‌های قدیم مذهبی بودند، گفتند: »ما با کسی کاری نداریم، ما داریم در یک مسئله‌ای که در آن اختلاف داریم گفتگو و مباحثه می‌کنیم و اینجا هم بیابان خداست و بلند حرف زدن آزاد است، اینجا که بچه؟ کسی نخوابیده که بیدار شود یا آدام مریض بستری نیست که ناراحت بشود!«

شکارچی گفت: »آخر من اینجا دام گذاشته‌ام و می‌خواهم کبوتر بگیرم و این‌ها از سر و صدای شما می‌ترسند و فرار می‌کنند ولی اگر ساکت باشید ممکن است به دام بیفتند.«

طلبه‌ها جواب دادند: »تو می‌گوئی ما از کار خودمان دست بر داریم تا تو به کار خودت برسی؟ در این صورت اگر تو حاضر هستی دو تا کبوتر هم به ما بدهی ساکت می‌شویم وگرنه هر کسی باید به کار خودش برسد و اینجا جای درس خواندن ما است. ممکن است تو بروی بساط خود را جای دیگر پهن کنی.«

صیاد گفت: »آقایان عزیز، آخر من آدم کاسب هستم و چند نفر نان‌خور دارم و باید با فروش این مرغ‌ها زندگی کنم و از صبح تا حالا انتظار کشیده‌ام تا حالا که سه کبوتر آمده‌اند نزدیک تله می‌چرند و ممکن است به دام بیفتند و اگر دو تا را شما ببرید و یکی بماند برای من نان نمی‌شود.«

آن دو نفر جواب دادند: »تو هر روز این کار را می‌کنی و ما مدت‌هاست گوشت شکار نخورده‌ایم و چون گوشت کبوتر در مدرسه؟ ما خیلی تحفه است ما می‌خواهیم امروز به دوستان خود در مدرسه مهمانی بدهیم و کبوترهای امروز قسمت ما است.«

صیاد گفت: »آخر ای خوش‌انصاف‌ها، این مرغان که کبوتران مدرسه نیستند، مال بیابانند، این دام را هم که طلبه‌ها نساخته‌اند، زن من بافته است، این زمین هم که وقف مدرسه نیست و شما هیچ حقی به گردن من ندارید، پس چرا می‌خواهید مزاحم بشوید.«

اما هرچه شکارچی التماس کرد، به گوش آن‌ها نرفت که نرفت و گفتند یا باید قبول کنی که دو کبوتر هم به ما بدهی تا ساکت شویم یا ما هم به کار خودمان مشغول می‌شویم و اگر مرغ‌ها پریدند به ما مربوط نیست و تو هم حق نداری برای درس خواندن و مباحثه کردن ما مزاحم بشوی.« . . .

3.5. The Farsi Alphabet additions

The Persian alphabet shares with the Arabic alphabet most of its letters. There are **four** *extra letters* that are not used in traditional Arabic.

- U+067E Peh پ,
- U+0686 Tcheh چ,
- U+0698 Jeh ژ,
- and U+06AF Gaf گ

While two letters have **modified forms** in final and isolated position as you can see in the table 3.1 on page 19:

- `U+064A` Arabic Letter Kaf (Arabic Kaf) ك
- `U+06CC` Arabic Letter Keheh (Persian Kaf) ک
  Keheh is the *Sindhi* name.
- `U+064A` Arabic Letter Yeh (Arabic Yeh) ي
- `U+06CC` Arabic Letter Farsi Yeh (Persian Yeh) ی

Character Name	Isolated form	Final form	Medial form	Initial form
Arabic Kaf	ك	ك	كـ	كـ
Persian Kaf	ک	ک	کـ	کـ
Arabic Yeh	ي	ي	يـ	يـ
Persian Yeh	ی	ی	یـ	یـ

Table 3.1: The modified Farsi letters

We call them `\farsikaf` and `\kaf`. The `\y` is used for both forms of the letter Yeh, while the font encoding `LAE`, used with Arabic, points to the Arabic Yeh and the font encoding `LFE`, used with Farsi, points to the Farsi Yeh. This is just to allow the use of `CP 1256` that does not contain a Farsi Yeh. Nevertheless, since the user is supposed to work in 8-bit and rarely has to type letter names as control sequences, there should be no problem. Remark that the final and isolated form of the Farsi Yeh is identical to the Arabic *Alef Maqsura* ی but Farsi does not use the character Alef Maqsura. It has no *Taa Marbutah* ة neither!

The letter `U+06C0` Heh with Yeh Above, which is in fact represented as *Heh with hamza above*, that does not exist in Arabic is not yet implemented. It does not appear in `CP 1256` or in `ISO 8859-6` code pages neither!

Farsi uses also two other special characters Arabic does not use. First, `ZWNJ` (`U+200C` *Zero Width Non-Joiner*) which seems to be widely used in Farsi to prevent joining without adding a space. Think of the `\bibname` کتاب‌نامه. This can be obtained in Arabi also by typing a star `\ZWNJ` between the characters or you just type the `ZWNJ` on your Farsi keyboard.

The second one is `ZWJ` (`U+200D` *Zero Width Joiner*), to force a character to join when it normally would not seems to be used occasionally only. You can also do this in Arabi by typing `\noboundary` or you just type it on your Farsi keyboard. The examples in § 4.7 on the use `\noboundary` on page 27 may help you too.

For the numbers in Farsi, go to section § 4.6.

The *hamza* and its different forms. Although the form of the *hamza* can be *explicitly* known from the vowel of the character preceding the *hamza* carrier and the vowel of the hamza carrier itself.

We rely on the fact that the user should type the character he needs and the program has not to guess each form the *hamza* will have. This is the way Arabic is written on usual visual

systems and the case for existing 8-bit Arabic texts. All the possible forms ء, ئ, ؤ, أ and إ exist on the *keyboard*.

How Arabic and Farsi poetry is typeset. The (classical) poetry, in both Arabic and Farsi, is formatted in two "parallel" verses that begin and end at the same positions. When verses are too short, they are written closer to the (vertical) center of the page like in the example:

لا تفرحن بفأل ، إن سمعت به ؛

أبو العلاء المعري

لا تفرحـــن بفـــأل ، إن سمعـــت بـــه ولا تطيـــر ، إذا ما ناعـــب نعبـــا
فالخطـــب أفظـــع مـــن سراء تأملهـــا والأمر أيســـر مـــن أن تضمـــر الرعبـــا
إذا تفكرت فكـــرا ، لا يمازجـــه فســـاد عقل صحيح ، هان ما صعبـــا
فاللب إن صـــح أعطى النفس فترتهـــا حتى تمـــوت ، وسمى جدها لعبـــا
وما الغـــواني الغـــوادي ، في ملاعبهـــا إلا خيالات وقـــت ، أشبهـــت لعبـــا
زيـــادة الجسم عنت جسم حاملـــه إلى التـــراب ، وزادت حافرا تعبـــا

Our choice for the Farsi poetry, that will close this chapter, goes to one of the prestigious poets of all times. A few verses of *Hafez* حافظ inscribed on his tomb in Shiraz شیراز, the *Hafeziyye*:

غزال

حافظ

مژده وصل تو کو کز سر جـــان برخیزم طایر قدسم و از دام جهـــان برخیزم
به ولای تو که گر بنده خـــویشم خوانی از سر خواجگی کون و مکـــان برخیزم
یا رب از ابر هدایت برســـان بارانی پیش تر زان که چو گردی ز میـــان برخیزم
بر سر تربت من با می و مطـــرب بنشین تا به بویت ز لحد رقص کنـــان برخیزم
خیز و بالا بنما ای بت شیرین حرکـــات که چو حافظ ز سر جـــان و جهـــان برخیزم
گرچه پیرم تو شبی تنگ در آغوشم کش تا سحرگه ز کنـــار تو جـــوان برخیزم

Figure 3.1: NUTSHELL. Arabi and the parshape package I

CHAPTER 4

Use of the System
إستعمال النظام

تعلم فليس المرء يولـــــد عالمـــــاً وليس أخـــو علـــم كمن هو جاهـــل
وإن كبير القـــــوم لا علـــــم عنده صغير إذا ضمـــــت عليه المحافـــل

4.1. Input encodings supported by Arabi

Typesetting Arabic and Farsi texts with TEX implies the use of special *input* and *output encodings*, so we would need to use the packages `inputenc` and `fontenc`.

We use two special font encodings. For Arabic we use `LAE` that stands for `Local Arabic Encoding`, defined in the file `laeenc.def`. While for Farsi we use `LFE` that stands also for `Local Farsi Encoding`, defined in the file `lfeenc.def`.

> These two encodings *are not final*. Some character positions may/will change, and some still empty slots will be filled with new characters.

Concerning the input encoding, the user simply creates an ordinary LATEX file, in which he can use `8-bit` Arabic characters, typed visually on some system that supports the Arabic script.

For now, the system Arabi supports the following input code pages:

1. Arabic Windows `CP 1256` for both Arabic and Farsi,
2. `ISO 8859-6` for Arabic, a lot of Farsi characters are missing.
3. The multibyte encoding `UTF-8` (Unicode Transmission Format) `ISO 10646` for both Arabic and Farsi. This encoding is rather new i LATEX and is still experimental?? We could compile a Farsi document in Unicode when the whole document was in Unicode, and the same document complaints about some characters that are not set up for use with LATEX. We think that we covered all Farsi characters, but if you find any missing characters, we would like you to contact us to add them!
4. A partial support for a `7-bit` input encoding, similar to the ASCII input encoding of ArabTEX, will also be provided[1], this is not a particularly urgent problem since you can use ArabTEX with it's standard `ASCII` encoding within an Arabi LATEX document. The choice of ArabTEX is motivated by the fact, according to our experience with the Arabic TEX users community locally, the most Arabic users that use TEX to typeset their Arabic texts use actually ArabTEX. So, the potential users that may get interested in using Arabi that were using ArabTEX with ASCII input should be able to use their old documents at

[1] In fact, there is a partial support as for now, but we did not get enough time to complete it. It should be improved and supported in future versions but we make no promise.

the cost of minor changes to their files and habits. For those who were using ArabTEX with some `8-bit` encoding we hope that there should be no problem.

4.2. Declaring the right input encoding

You have to specify the default input encoding, with the help of the standard `inputenc` package, using **before** loading BABEL,

```
\usepackage[encoding name]{inputenc}
```

For example you say

```
\usepackage[cp1256]{inputenc}
```

for Windows Arabic `CP 1256`, or

```
\usepackage[8859-6]{inputenc}
```

for the standard `ISO 8859-6` encoding used by UNIX and Linux systems, or

```
\usepackage[utf8]{inputenc}
```

for the standard Unicode `UTF-8`, known also as `ISO 10646` encoding, supported actually by all modern systems. You may also combine many, the last one being the default.

If no input encoding has been chosen, Windows Arabic `CP 1256` is used as the default, for the moment.

The choice of an input encoding in the preamble is not a restriction since it can be changed also inside the document by the `\inputencoding` command, like

```
\inputencoding{cp1256}
```

for example. A possible use of this command within a document is when using text from several documents to build up a composite work. You may not get Farsi text in `CP 1256` format while your Arabic text is in `CP 1256` or `ISO 8859-6`, will may need this "feature."

> Older 7-bit codes, like the one used *jadis* by the DOS operating system, **are not supported**. They require a new corresponding font encoding, and in view of the fact that they are actually not used anymore as far as we know, this should be a waste of time and storage space.

Each encoding has an associated `.def` file. For example, the file "`8859-6.def`" defines the characters of the standard `ISO 8859-6` encoding.

4.3. Calling BABEL

Then, you have to call BABEL with the `arabic` or `farsi` options or both as in

```
\usepackage[farsi,arabic,french,english]{babel}
```

the last option being always the *default* for the document.

Important note.
Nevertheless, since the Arabic part of the code was developed first and the Farsi part was developed after, we advise the user who wants to mix the two languages to call the option `arabic` always after `farsi`. And if you want Farsi to be the main document language, just issue the command

```
\TOCLanguage{english}
```

and call the Farsi language whenever needed as shown below. This may save you a lot of trouble (concerning the choice of the fonts to be used by the system) until the system becomes more mature.

Actually, the Arabi font encodings (LaTeX Arabic encoding `LAE` and Farsi encoding `LFE`) should be used, with the aid of the standard `fontenc` package:

```
\usepackage[LAE,LFE]{fontenc}
```

This font encoding was designed by partially placing Arabic glyphs in their corresponding ASCII equivalent (if one character) from the ArabTeX 7-bit input encoding looking for further compatibility to allow inputting text in ASCII if wanted.

The way we adopted during the implementation of our system relies on the existing character set *on the keyboard*. That made that ء, ئ, ؤ, أ and إ can/should be typed explicitly by the user and the program has not to guess each form the hamza will have. This is the way Arabic is written on usual visual systems and the case for existing 8-bit Arabic texts. So, we ignore consciously the fact that the actual position of the hamza can be explicitly known from the vowel of the character preceding the hamza carrier and the vowel of the hamza carrier itself.

Then run your TeX file through ε-LaTeX as you usually do for any LaTeX file and the standard LaTeX. If you use the console, you should have to type something like

```
$prompt> elatex foo.tex
```

4.4. Preparing an Input file

Languages can be switched for example by the following standard macros used by the BABEL system to switch between different languages. there is nothing really special there. We just need some few macros to call either Arabic or Farsi little portions of text inside a dominating Latin text.

For Arabic:

```
\selectlanguage{arabic}
```

كتابة باللغة العربية
كتابة أخرى باللغة العربية أيضا
...

```
\begin{otherlanguage}{arabic}
```

كتابة باللغة العربية
كتابة أخرى باللغة العربية أيضا . . .

```
\end{otherlanguage}
```

For Farsi:

```
\selectlanguage{farsi}
```

بسمه تعالی
جمهوری اسلامی ایران . . .
...

```
\begin{otherlanguage}{farsi}
```

بسمه تعالی
جمهوری اسلامی ایران . . .

```
\end{otherlanguage}
```

Inside *Farsi text*:
بسمه تعالی
متن فارسی . . .
`\textLR{Latin text}`
متن فارسی . . .

Inside *Arabic text*:
كتابة بالعربية . . .
`\textLR{Latin text}`
كتابة أخرى باللغة العربية
أيضا . . .

Inside *Latin text*:
`Latin text`
`\AR{ كتابة بالعربية }`
`more Latin text`
`\FR{ متن فارسی }`
and more . . .

After these examples that show you how to use Arabi to select the Arabic and Farsi languages, some explanations. First, the command

`\selectlanguage{LanguageName}`

and the environment

`\begin{otherlanguage}{LanguageName}`

are a part of standard BABEL switching interface.

The other commands are used for language and direction switching inside a paragraph.

The command `\textLR{Latin text}` is used to type a Latin text inside Arabic or Farsi. The last Left-to-Right language will be used.

The commands `\FR{متن فارسی}` and `\textFR{متن فارسی}` are used to get a Farsi text in any context.

While the commands `\AR{كتابة بالعربية}` and `\textAR{كتابة بالعربية}` are used to get an Arabic text in any context.

You have also the possibility to use the shorthand command `\textRL{xxxx}` to type either Arabic or Farsi, the language that was *loaded last*! Nevertheless, this is deprecated!

To force insertion of Arabic or Farsi in English text, you should use `\ARmbox` for Arabic `mbox`, `\FRmbox` for Farsi `\mbox` and `\LRmbox` for English (in fact Latin) `\mbox` macros respectively. This is sometimes necessary, like inside mathematical formulae for example.

4.5. The document

For a short example that shows how Arabi is used, get a look at the *sample input text* in Figure 4.1 on page 26 and how its *output* looks like in Figures 4.2 on page 27. There is nothing specific to Arabic in the example, we use it for Farsi the same way!

4.6. Numbers within Arabic and Farsi texts

The numbers are written normally in a Latin context. But in a Right-to-Left context, that is Arabic or Farsi for us, and since the Arabic text should be reflected, the numbers have to be protected in some way that prevents such reflection and choose the right font to be used (numbers in Arabic are written in two different forms according to the Arabic country where they are used).

Arabi input

ما تكتبونه مع العربي

```
\documentclass{article}

\usepackage[ cp1256 ]{inputenc}

\usepackage[ LAE ]{fontenc}

\usepackage[ arabic ,english]{babel}

\begin{document}

\selectlanguage{arabic}

بسم الله الرحمن الرحيم .
\\
الفصل السادس عشر في الاستخارة


في صحيح البخاري عن جابر قال كان رسول الله \salat يعلمنا الاستخارة في الامر كما يعلمنا السورة من
القران اذا هم احدكم بالامر فليركع ركعتين من غير الفريضة ثم ليقل اللهم اني استخيرك بعلمك واستقدرك
بقدرتك واسالك من فضلك العظيم فانك تقدر ولا اقدر وتعلم ولا اعلم وانت علام الغيوب اللهم ان كنت تعلم
ان هذا الامر ويسمى حاجته خير لي في ديني ومعاشي وعاقبة امري فاقدره لي ويسره لي ثم بارك لي فيه
وان كنت تعلم ان هذا الامر شر لي في ديني ومعاشي وعاقبة امري فاصرفه عني واصرفني عنه واقدر لي
الخير حيث كان ثم ارضني به وفي مسند الامام احمد من حديث سعد بن ابي وقاص عن النبي ص انه قال من
سعادة ابن ادم استخارة الله ومن سعادة ابن ادم رضاه بما قضى الله ومن شقوة ابن ادم تركه استخارة الله
ومن شقوة ابن ادم سخطه بما قضى الله وقد قال سبحانه وتعالى
                                                    [\textmash{
وشاورهم في الامر فاذا عزمت فتوكل على الله
                                                    }]
وقال قتاده ما تشاور قوم يبتغون وجه الله الا هدوا الى ارشد امرهم
\textLR{This is a simple example of Arabic text you may want to type}
نم والحمد لله رب العالمين.

\end{document}
```

Figure 4.1: Sample Arabi input

Numbers should be used with the control sequence `\I{Arabo-Indic number}`, as in `\I{0123456789}` to be typeset using *Arabo-Indic digits* (U+0660 – U+0669) like ٠١٢٣٤٥٦٧٨٩. The same control sequence `\I` works to get *Extended Arabo-Indic digits* (U+06F0 – U+06F9) ۰۱۲۳۴۵۶۷۸۹ if you are in a Farsi text. The choice of one of the two forms is done automatically according to the contest. That is if you are in a Farsi or an Arabic text.

Figure 4.2: Sample Arabi output

If you are in a Latin text and would like to get Arabo-Indic digits —as in this user guide— you still may use `\I` but if you want to get Extended Arabo-Indic number you have to use the control sequence `\EI{Extended Arabo-Indic number}`.

If you prefer/need to get *Arabic digits*, use only `\textLR{123}`, to get 123. You should use one of these commands according to what you want to get at the end.

4.7. Breaking ligatures

All the typesetting in Arabi is done via TEX's *ligature* mechanism. When you type Arabic (or Farsi, the word Arabic is used here to describe the script, not the language) text, each character corresponds for Arabi to the medial form, always —even if not in the middle of a word— and that is the ligature mechanisms that decides of the form the character should use!

For example, when you type the word كتب Arabi asks for the medial form of the characters: ـكـ then ـتـ and ـبـ and that is TEX's ligature mechanism which is responsible for converting this to كـ then ـتـ and finally بـ and ε-TEX `\beginR` and `\endR` primitives will reflect them to get بـتـكـ which is what you want most of the time.

However, in the sentence سنة ١٤٢٦ هـ الموافق إ ٢٠٠٥ م to specify a *Higri date* for example, you have to break this ligature. One way to do that consists in using explicitly the word `\noboundary` before of after the character, depending on the form you want to get as a result. For example in the above example we typed the letter *hā'* ه followed immediately (no space follows) by the control sequence `\noboundary`.

Note that you cannot get this initial form in visual systems like Word processors unless you add an explicit *tatweel* (to lengthen space between words) after the letters *hā'* and *lām* like:

ـهـ and ـلـ instead of ه and ل

or with an other font whose keshida is longer

ـهـ and ـلـ instead of ه and ل

where the difference is more visible!

4.8. Turning vowels ON/OFF

The Arabi system allows *vowelization* if entered by the user as any Arabic (or Farsi) character, like in ضَرَبَ زَيْدٌ عَمْرًا. if the font used contains such vowels!

Fluent readers of Arabic rarely resort to vowels, so the above sentence would be typed and rendered ضرب زيد عمرا.

Vowels are used usually in books for children, to explicit the pronunciations of foreign words, or in some grammar or religious books. Nevertheless, even if you don't want to typeset the vowels, you may still include them in the input, this will render a more accurate transliteration if you are interested in. Once this is done you have the choice between getting the vowels in the output or no. This is defined in the mini-package `arabnovowel.sty`.

> For the user, it suffices to issue the command `\Novocalize` to turn them OFF in the output and to issue the command `\Vocalize` to turn them ON again. By default, they are ON.

Beware that if the vowels are turned OFF, they will not appear in the Latin transliteration of Arabic. You have to turn them ON again before generating the transliteration.

The effects of `\Vocalize` and `\Novocalize` are best shown in the example of Figure 4.3 on page 29.

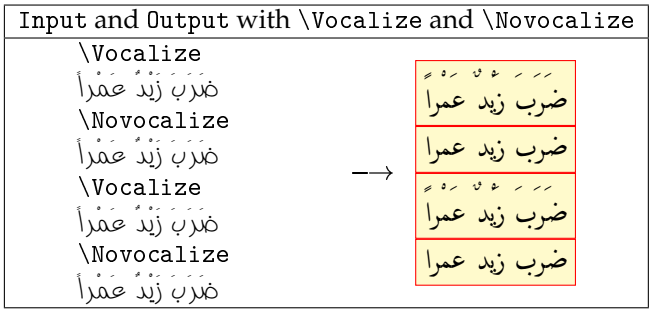


Figure 4.3: Activating, Deactivating and Reactivation vowelization again

4.9. Forcing the final form of a vowelized connecting character

This section does not apply for XeTeX users who use AAT Open type fonts!

When typing Arabic text, that is *not vocalized*, there is no problem. You can type it as you would like to get it in the output using the 8-bit (Arabic) characters on your keyboard. It's full WYSIWYG!

Concerning *vowelized words*, if the character to vowelize is not the final one in the word, there too, there is no particular convention to care about. Even if your last character should be vowelized and it's non connecting, there is no problem neither. But if it's a vowelized connecting character, it will show with a medial form *if the vowel is not a tanween*, instead of a final one or in an initial form instead of an isolated one. This is a problem we got, due to a limitation in the way ligatures are handled by TeX. So, the solution, there is one hopefully, is to type after the last connecting character and before the vowel it's carrying a star character "*" as it is unlikely that you would want a star between a character and its vowel anyway. The result would be the correct vowelized character in the output!

Some examples are given in Figure 4.4 on page 29.

What you type	And what you get
لغة الضاد	لغة الضاد
لُغَةٌ -- ضَادٌ	لُغَةٌ — ضَادٌ
لُغَةُ الضادِ	لُغَةُ الضادِ
عمل عامل	عمل عامل
عَمَلَ عَامِلٌ	عَمَلَ عَامِلٌ
عَمَلَ* عَامِلٌ	عَمَلَ عَامِلٌ

Figure 4.4: Typing final vowelized connecting characters

CHAPTER 5

Bi-directional Support
كيف تختار الاتجاهين

قال أبو نواس:

خذ العلـــــــوم ولاتعبأ بناقلـــــــها واقصُد بذلك وجه الخالـــق البـــاري

إن العلـــوم كأشجـــار لها ثمـــر فـــاجنِ الثمـــار وخل العـــود للنار

This chapter describes the Arabi Bi-directional capabilities. Unless, something specific to Arabic script texts is not working with the combination of the standard (and some less standard) LaTeX classes and styles by using the current macros from `arabic.ldf`, `farsi.ldf` or `arabicore.sty`, the decision is taken to use actually as it as is. It was tested with standard LaTeX 2_ε classes. The syntax is the standard one as you will see.

5.1. Global formatting parameters

Some formatting parameters may be seen as global in a document (some times in a chapter or a part) like the table of contents, the list of figures, the list of tables, and the page numbering styles.

That means for example in the case of page numbers that we should not mix Arabic digits page numbers with Indo-Arabic digits! This holds true more generally with the `headings` as all. They should depend on a global prevailing direction for the whole document. We have either an Arabic document that contains Latin (or other Left-to-Right scripts) texts or the converse. So, we should ensure that the proper disposition for the heading is taken.

For the table of contents, the list of figures and the list of tables. They should appear in a uniform way even if they contain Arabic (Right-to-Left) and Latin (Left-to-Right) entries. If no thing is done due to the BABEL mechanism that inserts a language switching command in the table of contents whenever such a command appeared in the test, the tables should appear in mixed directions if you have mixed Latin and Arabic sectioning commands

Our vision is that the global direction of the document should dominate. For example in the case of the table of contents, consider Arabic or Farsi and English [sub]sections or chapters that appear in the text of the document in two different contexts (Arabic and Farsi from Right-to-Left and Latin from Left-to-Right) with different formatting! In the table of contents, if the prevailing language (the table of contents (TOC) language) is Arabic or Farsi, all the entries, Arabic or Farsi and Latin, should appear from Right-to-Left with sections and page numbers in the right format, while the text of the heading will of course still remain typeset with Latin characters if it was not Arabic or Farsi! And conversely for an English table of contents or any Latin (Left-to-Right) language.

Other formatting features may be seen as semi-global or just local(?!) like the position of the `\footnoterule` for example whose position should depend on the *page* language context as it happened while preparing this user guide (where English dominates) while in some pages where only Arabic footnotes had to be issued and it would have been certainly inappropriate to get a `\footnoterule` on the left of the page! The same holds for the list of figures and the list of tables.

Other (semi-)global parameters for a document may also exist. They are not all dealt with in this version. We are still discovering them as we use the system more and more. The users interaction is very important to cover them all in future versions!

5.2. Sectioning commands

Usual LaTeX control sequences for sections headings

```
\subsubsection,
\subsection,
\section,
\chapter,
\part,
[\caption]
```

get typeset with the *default direction* in effect. That is if you have *selected* the Arabic language, any sectioning command that will be issued will be typeset from Right-to-Left with the appropriate choice of font, size of characters, etc. And if you have *selected* a language that uses some Left-to-Right script, the sectioning commands will be typeset from Left-to-Right with the usual formatting. Sections, subsections, chapters & co. all share the same counters that will be incremented whenever such a command is issued independently of the Language context where it was typed.

Proper formatting of the table of contents according to the *TOC language* is done automatically by Arabi and you have not to worry about it.

In the case where the you want a Left-to-Right language chapter and you are in a Latin context (you have typed `\selectlanguage{english}` for example), a new chapter command like

```
\chapter{How to use this package}
```

is issued. While if in an Arabic or Farsi context, we should declare a chapter as in

```
\chapter{كيفية استعمال هذه الرزمة }
```

or

```
\chapter{بسم تعالى}
```

The *table of contents*, *list of figures* and *list of tables* should be typeset, as usual, with the commands `\tableofcontents`, `\listoffigures` and `\listoftables` respectively. Their respective captions will be typeset in the main direction (and language), what we may call the *TOC language*, in effect where the above commands are placed. You have to issue a complementary command in your preamble somewhere after loading Arabi with the command

```
\TOCLanguage{<TOC Language>}
```

In the place <TOC Language> you should specify some BABEL valid language of course like `\TOCLanguage{arabic}`. If such a command was not given in the preamble, the system will suppose that the <TOC Language> is the main BABEL language `\bbl@main@language`, that is the last language specified with BABEL.

What happens actually is that, an Arabic section entry, for example, will appear from Right-to-Left with Indo-Arabic digits for the numbers while a Left-to-Right Latin section entry will appear from Left-to-Right with Arabic digits for the numbers, etc. This is normal because the package BABEL issues language switching control sequences like `\selectlanguage{english}` in the three tables auxiliary files whenever such a command appears in the TEX document!

Other global parameter besides *tables*, that should use the main language context include the *page numbering* scheme and the *headings*.

An other element of style that seems semi-global and merits more attention and reflexion is "footnotes". In a page where only Right-to-Left text (for example Arabic) appears, all footnotes and the footnote rule should appear on the right and similarly for a page where only Left-to-Right text (for example English) appears, all footnotes and the footnote rule should appear on the left, without any consideration for the global direction of text. The problem is when both Right-to-Left and Left-to-Right footnotes should appear on the same page or when a page with an Arabic paragraph or two appear on the top and the rest of the text is in Latin on the bottom and a Right-to-Left footnote should be issued, how should we proceed?

5.3. Displayed Material. Lists environments

Concerning displayed material, we shall begin by showing first some examples of how it looks like to use the different list environments.

The `itemize` environment. The syntax to use an `itemize` environment is the same as with standard Latin texts you are used to in LATEX except that you have to issue a `\selectlanguage{arabic}` or `\selectlanguage{farsi}` command to switch BABEL to Arabic or Farsi. The four levels of an Arabic `itemize` environment, for example, looks like:

بسم الله الرحمن الرحيم .

إذا تعرضنا لتراث العرب العلمي في الرياضيات فقد نقرأ عما يلي :

- مآثر العرب في الحساب
  - نظام الترقيم وأنواع الأرقام
  - فكرة الصفر ومزايا النظام العشري
  - فضل العرب في اختراع الكسر العشري
    - أبواب الحساب
      - طرق الجمع والضرب وفوائدها للمبتدئين
      - بحوث النسبة
      - طريقة الخطأين
      - طريقة الكفات
    - نظريات الأعداد المتحابة ونظرية ابن قرة

* المتواليات
  - استخراج المجهولات
  - طريقة العمل بالعكس
- مآثر العرب في الجبر
- مآثر العرب في الهندسة
- مآثر العرب في المثلثات

The four levels of a Farsi `itemize` environment are the same. So, there is no need to rewrite them all. we provide a little humoristic example in Farsi entitled *Credit is Dead*, whose translation (always from the Persian studies site at the university of Texas) is:

```
He is the daily provider

    Credit is dead

    For this reason we announce Due to the tragic accident of delinquency

    Credit has passed away

    Due to not having:

      • The Treasure of Croesus
      • The Patience of Job
      • The Longevity of Noah
      • The Ability to withstand your being distant

    Please excuse us from giving credit or handouts

    In friendship Alike.     In business Strangers.
```

هو الرزاق

نسیه مرد

به این وسیله به اطلاع می‌رسانیم در اثر حادثه دلخراش بد حسابی

نسیه درگذشت

به علت نداشتن

- گنج قارون
- صبر ایوب
- عمر نوح
- طاقت دوری شما

از دادن نسیه و وجه دستی معذوریم

در دوستی یگانه در معامله بیگانه

The `enumerate` **environment.** The syntax to use an `enumerate` environment is also the same as with standard Latin texts you are used to in LaTeX. And there too, you have to use a `\selectlanguage{arabic}` or `\selectlanguage{farsi}`. The four levels of an Arabic `enumerate` environment, for example, looks like:

بسم الله الرحمن الرحيم .

إذا تعرضنا لتراث العرب العلمي في الرياضيات فقد نقرأ عما يلي :

(١) . مآثر العرب في الحساب
ا. نظام الترقيم وأنواع الأرقام
ب. فكرة الصفر ومزايا النظام العشري
ج. فضل العرب في اختراع الكسر العشري
١- أبواب الحساب
ا- طرق الجمع والضرب وفوائدها للمبتدئين
ب- بحوث النسبة
ج- طريقة الخطأين
د- طريقة الكفات
٢- نظريات الأعداد المتحابة ونظرية ابن قرة
٣- المتواليات
د. استخراج المجهولات
ه. طريقة العمل بالعكس
(٢) . مآثر العرب في الجبر
(٣) . مآثر العرب في الهندسة
(٤) . مآثر العرب في المثلثات

While the four levels of a Farsi `enumerate` environment looks like this. We use the same Arabic text as the above example of enumeration, supposing it's Farsi.

بسم الله الرحمن الرحيم .

إذا تعرضنا لتراث العرب العلمی فی الریاضیات فقد نقرأ عما یلی :

(١). مآثر العرب فی الحساب
ا. نظام الترقیم وأنواع الأرقام
ب. فکره الصفر ومزایا النظام العشری
پ. فضل العرب فی اختراع الکسر العشری
١- أبواب الحساب
ا- طرق الجمع والضرب وفوائدها للمبتدئین
ب- بحوث النسبه
پ- طریقه الخطأین

ت- طریقه الکفات
۲- نظریات الأعداد المتحابه ونظریه ابن قره
۳- المتواليات
ت. استخراج المجهولات
ث. طریقه العمل بالعکس
(۲). مآثر العرب فی الجبر
(۳). مآثر العرب فی الهندسه
(۴). مآثر العرب فی المثلثات

You must have noticed the, for the moment, we use the same form for the counters of the second and fourth levels of the enumerate environment. This should be fixed in the next version according to what is done in the standard Arabic literature. The same are used for Farsi with the exception that extended Arabo-Indic digits are used instead of Arabo-Indic digits.

Description environment. Concerning the `description` environment, the syntax is also standard. All you have to do is to precede it with a `\selectlanguage{arabic}` or `\selectlanguage{farsi}`. It looks like:

نظام الترقيم وأنواع الأرقام : نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم

فكرة الصفر ومزايا النظام العشري : فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري فكرة الصفر ومزايا النظام العشري

فضل العرب في اختراع الكسر العشري : فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري فضل العرب في اختراع الكسر العشري

If we use the same Arabic text as the above example of description, supposing it's Farsi.

قسمت اول : نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم

وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم وأنواع الأرقام نظام الترقيم

قسمت دوم : فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری فکره الصفر ومزایا النظام العشری

قسمت سوم : فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری فضل العرب فی اختراع الکسر العشری

5.4. More Displayed Material

Quotations. Concerning *quotations* there is nothing special neither, except that it is typeset from Right-to-Left of course. Of course, you type it like you do for Latin text with LaTeX. Here is a little example.

. . . الحمد لله المنفرد باسمه الاسمى ، المختص بالعز الأحمى ، الذي ليس دونه منتهى ولاوراءه مرمى ، وسع كل شئ رحمة وعلما ، واسبغ على اوليائه نعما عما ، تبارك اسمه ، وتمت كلمه ، وعمت نعمه ، وجمت حكمه ، وجرى بما كان ويكون قلمه ، واشهد ان سيدنا محمدا النبي الامي العربي الهاشمي القرشي عبده ورسوله، ارسله بدين الهدى ليظهره على سائر الاديان ، وايده بالقرآن المعجزة الخالدة المستمرة على تعاقب الازمان ، وضمن حفظه من تطرق التغيير اليه والحدثان ، صلى الله وسلم عليه صلاة تكون له رضا ، ولحقة اداء ، وكما هو اهله دائما وابدا . . .

And as we did above, we use a Farsi text about *"Hurricane Rita" which is nearing the coasts of the southern states of the US, is again plunging the residents of the southern states of the US like Louisiana, Texas and Florida into horror.*

. . . طوفان « ریتا » که به سواحل ایالت‌های جنوبی آمریکا نزدیک می‌شود، بار دیگر ساکنان ایالت‌های جنوبی آمریکا نظیر « لوئیزیانا، تگزاس و فلوریدا » را در وحشت فرو برد.
شبکه تلویزیونی سی‌ان‌ان اعلام کرد: پس از آنکه طوفان مرگبار « کاترینا » چندی پیش ایالت‌های جنوبی آمریکا نظیر لوئیزیانا و شهرهای واقع در این ایالت چون نیواورلئان و «

باتون روژ» را درنوردید و آنها را ویران کرد، اکنون طوفان خطرناک و قوی دیگری ساکنان این منطقه را در وحشت فرو برده است. . . .

5.5. Footnotes

Footnotes in Arabic and Farsi are also supported. The footnote text and footnote rule appear on the right side of the page as they should, like in the example below.

One problem with mixing Arabic Right-to-Left scripts with Latin ones is that the `footnote` commands issued in a mixed-script environments have to share the same separator line which depends on the current language when the page is output. So you have to choose a direction for your footnotes when you are typing. For example when you are in an *Arabic* page try to issue a footnote and inside use the control sequence `\textLR` to get Latin text instead of switching the language back and forth.

بسم الله الرحمن الرحيم

عونك اللهم يا لطيف

هذا النظام يتميز بكونه محمولا ويتمتع بقدر كبير من المرونة، لأنه قابل للاستعمال مع معظم ما تم إنجازه من إضافات وبرامج مساعدة في عالم تيخ[١] . إضافة إلى أنه لا يحتاج إلى أي معالج خارجي لتحديد أشكال الحروف في الكلمة .

. . . قلب الخليل أ، ب، ت، ث، فوضعها على قدر مخرجها من الحلق وهذا تأليفه:
ع، ح، ه، خ، غ، ق، ك، ج، ش، ض، ص، س، ز، ط، د، ت، ظ، ث، ذ، ر، ل، ن، ف، ب، م، و، أ، ي ـ همزة.[٢] (من مقدمة كتاب العين للخليل بن أحمد البصري ـ رحمة الله) .

And some Farsi footnotes too:

بسم الله الرحمن الرحيم

عونك اللهم يا لطيف

هذا النظام يتميز بكونه محمولا ويتمتع بقدر كبير من المرونة، لأنه قابل للاستعمال مع معظم ما تم إنجازه من إضافات وبرامج مساعده في عالم تيخ[٣] . إضافه إلى أنه لا يحتاج إلى أي معالج خارجي لتحديد أشكال الحروف في الكلمه .

---

[١]هذه إحالة بالعربية . يقول الإمام ابن تيمية في كتابه (اقتضاء الصراط المستقيم) : (واعلم أن نفس اللغة العربية من الدين، ومعرفتها فرض واجب في فهم الكتاب والسنة، وما لا يتم الواجب إلا به فهو واجب) .

[٢]وهذه إحالة أخرى بالعربية . تختص قواعد النحو بتحديد وظيفة كل كلمة داخل الجملة، وضبط أواخر الكلمات، وكيفية إعرابها، أي أن قواعد النحو تنظر للكلمة العربية من حيث أنها معربة (أي يتغير شكل آخرها بتغير موقعها من الجملة أو من حيث أنها مبنية (أي لا يتغير شكل آخرها بتغير موقعها) وهذا تفريق بينه وبين علم الصرف، الذي يختص ببنية الكلمة العربية وما يطرأ عليها من تغير بالزيادة أو النقص) (فهو علم يختص بنظام أبنية الكلم ولا دخل له في شكل آخرها

[٣]*هذه إحاله بالعربيه . يقول الإمام ابن تيميه في كتابه (اقتضاء الصراط المستقيم) : (واعلم أن نفس اللغه العربيه من الدين، ومعرفتها فرض واجب في فهم الكتاب والسنه، وما لا يتم الواجب إلا به فهو واجب).*

... قلب الخليل أ، ب، ت، ث، فوضعها على قدر مخرجها من الحلق وهذا تأليفه:
ع، ح، ه، خ، غ، ق، ك، ج، ش، ض، ص، س، ز، ط، د، ت، ظ، ث، ذ، ر، ل، ن، ف، ب، م، و، أ، ى ـ
همزه.[٤](من مقدمه كتاب العين للخليل بن أحمد البصرى ـ رحمه الله).

(٤)وهذه إحاله أخرى بالعربيه . تختص قواعد النحو بتحديد وظيفه كل كلمه داخل الجمله، وضبط أواخر الكلمات، وكيفيه إعرابها، أى أن قواعد النحو تنظر للكلمه العربيه من حيث أنها معربه (أى يتغير شكل آخرها بتغير موقعها من الجمله أو من حيث أنها مبنيه (أى لا يتغير شكل آخرها بتغير موقعها) وهذا تفريق بينه وبين علم الصرف، الذى يختص ببنيه الكلمه العربيه وما يطرأ عليها من تغير بالزياده أو النقص) (فهو علم يختص بنظام أبنيه الكلم ولا دخل له فى شكل آخرها

CHAPTER 6

Arabi Localisation
تمایز واختلاف

قال الطغرائي في لاميته المعروفة بلامية العجم:

حب السلامـــــة يثني هم صاحبـــــه ... عن المعـــالي ويغري المرء بالكســـل

فإن جنحت إليـــــه فاتخذ نفقـــــا ... في الأرض أو سلمـــا في الجو واعتـــزل

لو كـــان في شرف المأوى بلـــوغ منى ... لم تبرح الشمس يومـــا دارة الحمـــل

وشأن صدقـــك عند النـــاس كذبهم ... وهل يطـــــابق معوج بمعتـــــدل

ملك القنـــاعة لا يخشى عليـــه ولا ... يحتـــاج فيه إلى الأنصـــار والخـــول

ترجو البقـــاء بدار لا ثبات لهـــا ... فهل سمعت بظـــل غير منتقـــل

In this chapter, we will see some of the main issues that should be addressed in a BABEL language package, both for Arabic and Farsi.

6.1. Arabic and Farsi captions

The *Arabic and Farsi captions* used actually are summarized in the Table 6.1 on page 40). Those that are not defined yet are replaced with stars.

6.2. Arabic and Farsi dates

The names of the months used in each language are summarized in Table 6.2 on page 41. The Arabic form is just an Arabic form of the Western month used in the middle east. That is the *Arabic date* is just, the standard one expressed in Arabic:

Today is December 16, 2006

Today in Arabic is ١٦ كانون الأول ٢٠٠٦

The *Farsi date*, that uses *Jalali calendar*, is different. The Jalali calendar is the official one used in Iran, it's a solar calendar with its own leap years!

The code used is the one developed by the FarsiTEX فارسی‌تک team, we just rewrote the names of the months to be usable with Arabi correctly:

Command	English	بالعربية	فارسی
\prefacename	Preface	مدخل	***
\refname	References	المراجع	مرجع
\abstractname	Abstract	ملخص	چکیده
\bibname	Bibliography	المصادر	کتاب‌نامه
\chaptername	Chapter	الباب	فصل
\appendixname	Appendix	الملحق	پیوست
\contentsname	Contents	الفهرس	فهرست مندرجات
\listfigurename	List of Figures	قائمة الأشكال	لیست اشکال
\listtablename	List of Tables	قائمة الجداول	لیست جداول
\indexname	Index	الفهرس	فهرست الفبایی
\figurename	Figure	شكل	شکل
\tablename	Table	جدول	جدول
\partname	Part	القسم	بخش
\enclname	encl	المرفقات	***
\ccname	cc	نسخة موجهة إلى	***
\headtoname	To	إلى	***
\pagename	Page	صفحة	***
\seename	see	راجع	***
\alsoname	see also	راجع أيضا	***
\proofname	Proof	برهان	***
\glossaryname	Glossary	قاموس	***

Table 6.1: Captions in Arabic and Farsi

Today is December 16, 2006
Today in Farsi is ۲۵ آذر ۱۳۸۵

You can access the Arabic and Farsi dates even if not in an Arabic or Farsi document using the control sequences `\atoday` and `\ftoday` respectively.

You can also access the names of the Arabic and Farsi months using the control sequences `\arabimonth{month_number}` and `\farsimonth{month_number}` where `month_number` is of course a number between 1 and 12.

Notice that the current number order in the Jalali Calendar is not at the same order in the Western usual Gregorian calendar. Just compare the name of the month given by the Farsi date command seen above and

`\farsimonth{\month}` that gives اسفند

Month	Arabic name in the Middle East	Arabic name in North Africa	Jalali Calendar Local Official in Iran
1	كانون الثاني	يناير	فروردين
2	شباط	فبراير	اردیبهشت
3	آذار	مارس	خرداد
4	نيسان	ابريل	تیر
5	أيار	ماي	مرداد
6	حزيران	يونيه	شهریور
7	تموز	يوليوز	مهر
8	آب	غشت	آبان
9	أيلول	شتنبر	آذر
10	تشرين الأول	اكتوبر	دی
11	تشرين الثاني	نونبر	بهمن
12	كانون الأول	دسمبر	اسفند

Table 6.2: Arabic and Farsi month names

6.3. The Arabi `abjad` numeration system

The old `abjad` numerals used in medieval Arabic texts known in Arabic as حساب الجمل is also supported. It is widely used even these days to *enumerate items* in Arabic texts, so we need it to "arabize" the `enumerate` environment (You may see Table 6.3). The Farsi documents, we could see seem to use just a Farsi form of the `\alph` counter, that is the Farsi Alphabet in order. You can use it to write numbers in `abjad` notation using the control sequence `\abjadnumeral{number}`. The result is *different* according to the context: Arabic or Farsi!

> The control sequence `\abjadnumeral` is equivalent in the Farsi case to `\Fabjadnumeral` and in all other cases to `\Aabjadnumeral`. And the two are completely different!

As it may be wanted, the `abjad` notation can also be used with the control sequence `\pagenumbering` to get page numbers in `abjad` notation, like `\pagenumbering{abjad}`.

The Arabic case. In the Arabic language case, you can use the control sequence `\abjadnumeral{number}` to write numbers between 1 and 1999. You may use also the control sequence `\Aabjadnumeral{number}` that would produce Arabic abjad notation in *all cases*. The version with an additional letter capital `A`, is just in case you have some doubts.

You may use it like in `\Aabjadnumeral{1970}` to produce غظع.

If you use a number that is greater or equal to 2000, the control sequence `\Aabjadnumeral` will return the argument written in Arabo-Indic, as with `\abjadnumeral{2568}` for example, to get ٢٥٦٨.

The coding of the macros that produce abjad numerals profited a lot from similar one from the Greek package that supports BABEL. Many thanks to whom developed it!

abjad numeral	Value	abjad numeral	Value	abjad numeral	Value	abjad numeral	Value
ا	1	ي	10	ق	100	غ	1000
ب	2	ك	20	ر	200	—	—
ج	3	ل	30	ش	300	غظغ	1970
د	4	م	40	ت	400		
ه	5	ن	50	ث	500		
و	6	س	60	خ	600		
ز	7	ع	70	ذ	700		
ح	8	ف	80	ض	800		
ط	9	ص	90	ظ	900		

Table 6.3: Arabic abjad numerals

The Farsi case. The Farsi documents, we could see seem to use just a Farsi form of the \alph counter, that is the *Farsi Alphabet* in order. The additional characters are added where it seems appropriate, nevertheless, the two letters *wāw* و and *hāʾ* ه (If we assimilate the Farsi one to its Arabic counter part) have a reversed order comparing to the Arabic alphabet order!

You can use it to write numbers between 1 and 32 using the control sequence, if you are in a Farsi context, \abjadnumeral{number} like in \abjadnumeral{17} to produce ص. Otherwise you will have to use the control sequence \Fabjadnumeral, this will work in *all cases* too.

If you use a number that is greater or equal to 33, the control sequence \abjadnumeral will return the argument written in Extended Arabo-Indic, as with \abjadnumeral{36} for example, to get ۳۶.

abjad numeral	Value	abjad numeral	Value	abjad numeral	Value	abjad numeral	Value
ا	1	د	10	ط	19	م	28
ب	2	ذ	11	ظ	20	ن	29
پ	3	ر	12	ع	21	و	30
ت	4	ز	13	غ	22	ه	31
ث	5	ژ	14	ف	23	ی	32
ج	6	س	15	ق	24	—	—
چ	7	ش	16	ک	25	۳۶	36
ح	8	ص	17	گ	26		
خ	9	ض	18	ل	27		

Table 6.4: Farsi abjad numerals

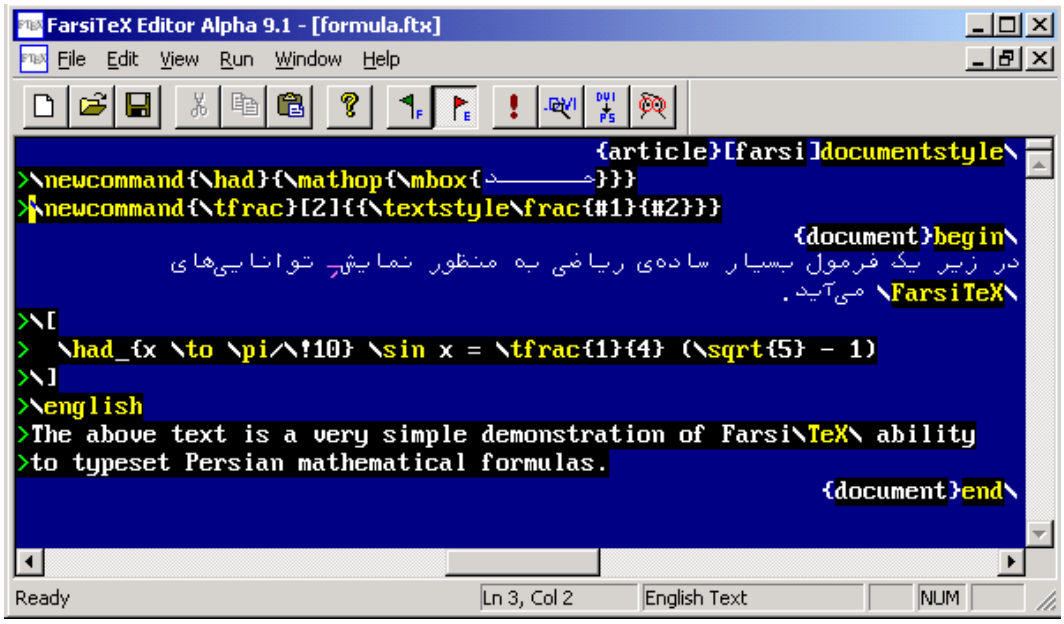


Figure 6.1: How Mathematics are written in the Farsi editor

6.4. Farsi mathematics

Farsi mathematics are in fact Left-to-Right mathematics with Farsi digits and operators names in Farsi like in:

$$\lim_{n\to\infty} \frac{۱}{n} = ۰$$

or

$$\lim_{x\to \pi/۱۰} \sin x = \tfrac{۱}{۴}(\sqrt{۵} - ۱)$$

To write formulae like these with Arabi, one had to use Farsi boxes to protect Farsi operator names and digits. For operator names this is the usual way of FarsiTEX to whom Farsi users are used to, We suppose, so it will just seem usual. But with FarsiTEX, digits in mathematical mode were automatically typeset in Farsi.

We add with this version 1.1 of Arabi this functionality, i.e., when the users switch to the Farsi language mode, he/she will gets automatically Farsi digits in mathematical mode, and the normal behavior, i.e. Arabic digits will be used when he/she exits from Farsi (see the following two figures from the paper [1] by Esfahbod and Pournader).

For example, if you type `$x^{24}$` or `$y_{56}$` inside a Farsi context you will get $x^{۲۴}$ or $y_{۵۶}$, and if the control sequence `\had` stands for the limit `\lim` defined by

```
\def\had{\mathop{\FRmbox{حـــــد}}}
```

We should get in displayed equations things like these.

This is the normal behavior in a Farsi context

```
\[
```

در زیر یک فرمول بسیار ساده‌ی ریاضی به منظور نمایشِ توانایی‌های فارسی‌تک می آید.

$$\lim_{x\to\pi/۱۰} \sin x = \frac{۱}{۴}(\sqrt{۵}-۱)$$

The above text is a very simple demonstration of FarsiTEX ability to typeset Persian mathematical formulas.

Figure 6.2: And how they should appear once typeset

```
\had_{x\to\pi/\!10}\sin x = \frac{1}{4}(\sqrt{5}-1)
\]
```

gives

$$\lim_{x\to\pi/۱۰} \sin x = \frac{۱}{۴}(\sqrt{۵}-۱)$$

If for some reason you want Arabic digits inside mathematical mode in a Farsi context, just type

```
\arabicmathdigits
\[
\had_{x\to\pi/\!10}\sin x = \frac{1}{4}(\sqrt{5}-1)
\]
```

and you will get

$$\lim_{x\to\pi/10} \sin x = \frac{1}{4}(\sqrt{5}-1)$$

And you can reverse again to Farsi digits by typing

```
\farsimathdigits
\[
\had_{x\to\pi/\!10}\sin x = \frac{1}{4}(\sqrt{5}-1)
\]
```

to get

$$\lim_{x\to\pi/۱۰} \sin x = \frac{۱}{۴}(\sqrt{۵}-۱)$$

The user has control on these two possibilities, and can choose the one that fits him/her. We have two control sequences that can be used anywhere in the text. The control sequence `\farsimathdigits` switches to Farsi digits in Mathematical mode, while `\arabicmathdigits` restores the normal behavior.

We could easily add the possibility to allow typing Arabic and Farsi letters inside mathematical formulae, but Arabic and Farsi words shall not be displayed in Right-to-Left and hence

will appear incorrect because ε-TEX does not allow `\BeginL`, `\BeginR`, `\endL` and `\endR` in mathematical mode!

At this stage, we do not want to introduce in Arabi a reversing macro *à la* Knuth and McKay. Using a Farsi `mbox` for that purpose in not really a problem since it will be done just once when defining the operator.

6.5. Transliteration signs and standards

Arabi comes with a *very experimental* module that produces a *transliteration* of `Arabic texts`. No counter part has been done for Farsi yet! We hope that it will be useful to people who study Arabic.

When texts are in general not fully vowelized, the transliteration cannot be expected to be correct. Moreover, when writing using some 8-bit input encoding (`CP 1256`, `ISO 8859-6` or `UTF-8`) there is absolutely no way to distinguish between long vowels ا و ي and the letters *alif*, *yaa* and *waw*. Neither, it is possible to write correctly the *hamza* when on *ʾalif*, *wāw*, or *yāʾ* (See also page 24).

In the case we develop enough the ASCII input encoding of Arabi, this module might be fully functional!

To use it, just load the package `translit` it as any other package, and type Arabic text in 8 bits in a Latin context, that is you will have to write using Arabic characters without issuing a command that switches to the Arabic language to get it to work. And **you have not** to type any particular command!

1	*ʾabw āl‹lāʾ ālm‹ry*	أبو العلاء المعري	1
2	*wlā taṭayr, ʾiḏā mā nā‹ibuN n‹bā*	ولا تطير. إذا ما ناعب نعبا	2
3	*matnuN mubārakuN*	متن مبارك	3
4	*ḥǧ mbrwr*	حج مبرور	4

Table 6.5: A little example of transliteration

6.6. Special characters

We define now some special characters that may be rendered using Arabi.

1. Arabi defines the control sequence `\allah` that works both in Farsi and Arabic modes to print the symbol used to specify the name of GOD
   āllāh الله إسم الجلالة.
   For the moment, we did not provide the Arabic fonts with this symbol yet, so if the font (in general, and this applies for now for ALL Arabic fonts) has no symbol ﷲ, it just prints the word الله using the font `nazli`.

2. Arabi defines the control sequence `\rial` in Farsi mode to print the Iranian currency `rial` symbol ﷼. If the font has no symbol `rial`, it just prints the symbol `rial` from the font `nazli` which has one for sure!

3. Arabi defines also the control sequence `\Decimal` in both Arabic and Farsi modes to print the *Arabic decimal separator* ٫. There too, if the font has no symbol Arabic decimal separator, it just prints the Arabic decimal separator symbol from the font `nazli` which has.

   For the moment, no macro has yet been implemented to manage automatically this symbol. This will be done in a near future. If you need it, you should type it by hand.

CHAPTER 7

Arabi Features
خصوصيات العربي

7.1. The fonts

With Arabi, we can use, in addition to (normal TEX) fonts in METAFONT format, Arabic fonts in True Type and Postscript format too.

We made the choice to choose only *good quality fonts* that may be *freely* available to the users. An exception is made for Microsoft Windows Arabic fonts that come with the system[1], since we suppose that many users of the Arabic script may have them and would certainly want to use them. Some were made by well known fonts editors and are really of a good quality. Moreover, these fonts contain in general enough characters to write the two scripts that interest us here!

Concerning the names of the fonts to use, we follow the standard naming scheme used by LATEX. You can think of `\textmybf` and the old `\bf` changed now to `\bfseries`.

That means that we use for each existing font `myfont`, we provide two commands. The first one has the form `\textmyfont` and takes as an argument the text that that follows like:

```
\textmyfont{The text to be written using the font myfont}
```

The second one has the form `\myfont` and takes no argument! It will apply to all the text that that follows:

```
\myfont The text to be written using the font myfont...
```

In the documentation below, we will mention only the `\textmyfont` form. The other can be deduced from the explained scheme.

7.2. Available Arabic fonts

The others used, *actually*, come from the GNU Arabic True type fonts distributed by `Arabeyes` with and without their Arabic version of Linux. They can be downloaded freely on the net from the `Arabeyes project` home page. We also may use the *Omega project* font for writing Arabic. The font is, unfortunately, subdivided in three parts, and the first one that contains Arabic language specific glyphs does not contain the parentheses, the exclamation sign, the dollar sign, the guillemots and the quotes. But it contains all the needed vowels, you may use it if you will not use these signs. The fact that the glyphs needed for Farsi are in two different physical fonts mean than no ligatures can be used unless we make *virtual fonts*.

[1] The `default` font for the moment is `Traditional Arabic`, from the Arabic fonts that come with Windows!

In a next version of Arabi we will try to add support for Farsi language also using this font and manage to get all the needed glyphs!

Unfortunately, Microsoft fonts will not be included with the Arabi distribution, you are supposed to have them on your system or download them *freely* from Microsoft Homepage. But of course, TeX fonts metrics and Font descriptions files are available with the distribution ready to be used! If you want to use them in a Type 1 format with `dvips`, you should convert them yourself, as the author did, using one of the many available tools! And of course make the appropriate changes in the `psfonts.map`

LaTeX arranges files in three families:

1. The Roman ones used for the main text,
2. the `Type Writer Type` family that simulates text written on a typewriter machine (this is in general a `Monospaced` font) and a
3. Sans Serif family. And these families are subdivided in different shapes like **bold**, *italic* and SMALL CAPS.

We tried to make a choice as to which font use for each of these categories. There are two main constraints:

1. First, the classical Arabic typography ignores such subdivision. We have some classical fonts like *naskhi*, *thuluth*, *reqaa*, etc.
2. Second, the fonts actually available at the disposition of the author are either free or non free, while he can distribute only the free ones!

So, we made a little choice of what font should occupy what position in this scheme imposed by LaTeX. Nevertheless, you have always the possibility to rearrange the fonts according to your taste, needs and what fonts you have! An other remark is that we did not make slanted ones from the ones we have already, this is possible and should be done in a next revision.

We provide the user who wants to typeset Arabic with a package `ARfonts` with three options that represent three categories in which we have already made such a choice, and which constitutes a *template* that can be easily used and adapted to your needs. Just create a file `myfonts.sty` copy one of these macros to it, modify it and load it whenever you want.

1. First, we have the option `free` that uses only Arabeyes free fonts, which logically should be the default to get the system running from the beginning!
2. Then, we have `mscore` for Arabic Microsoft core that uses only Microsoft fonts, which may be used if you like the fonts, but it's not really the best one. Courier in Arabic is not really a regal for the eyes, it has no kerning at all and the spacing seems correct. It is definitely not the Latin font you know!
3. Of course, the third choice, `mixed` that uses as its name means both Arabeyes free fonts and Arabic Microsoft ones. The font used as the Roman counter part with `free` is a little bit heavy comparing to a font like Traditional Arabic!

Free download of the Arabic font pack from Microsoft (`arafonts.exe`). Click the download button (marked تحميل) to the left of the filename

`http://office.microsoft.com/arabicregion/Downloads/2000/arafonts.aspx`

The GNU Arabeyes fonts are distributed in both True Type format and Type 1 Postscript (`.PFB`) format converted by the author!

	Roman (n) **b/bx, sl/it, sc**	`Typewriter Type` (n) **b/bx, sl/it, sc**	Sans Serif (n) **b/bx, sl/it, sc**
`free`	Almohanad	nice (N)	Sindibad (S)
	almateen, dimnah, nada	graph, N, N	S, S, S
`mscore`	Traditional Arabic (TA)	Arial (A)	Simplified Arabic(SA)
	TA bold, TA, Andalus	A bold, A, Andalus	SA bold, SA, SA
`mixed`	Traditional Arabic (TA)	nice (N)	Simplified Arabic(SA)
	TA bold, dimnah, nada	graph, N, N	SA bold, SA, SA

Table 7.1: The three font choices for Arabic

The following table (Table 7.2 on page 50) presents the fonts that are already available and may be used with no problem with the Arabi system to write Arabic.

7.3. Available Farsi fonts

The following table (Table 7.3 on page 51) presents the fonts that are already available and may be used with no problem with the Arabi system to write Farsi.

The same remark for Microsoft fonts, said in the former section, applies here too! The font *Farsi simple* in its two forms is also available with the Arabic fonts pack available freely on their site!

We use also *some* GNU fonts from the `FarsiWeb` project.

More fonts will be added in a next release!

As with Arabic, in Farsi too, we provide the user with with a package `FRfonts` with two options that represent two categories, and which constitutes a *template* that can be easily used to adapt it to your needs. Just create a file `myfonts.sty` copy one of these macros to it and modify it and load it whenever you want.

1. First, we have `free` that uses only FarsiWeb free fonts, which logically should be the *default* to get the system running from the beginning.
2. Then, we have `\mscore` for Arabic Microsoft core that uses only Microsoft fonts, which may be used if you like the fonts. the Farsi simple font is really nice and looks like a *Nastaliq* font. A similar free font (and others fonts) **will be** added in the next release! It is used in this user guide as the Roman default font for Farsi.

7.4. How to get Outline fonts for free

If you want *Outline fonts* from existing ones, easily as a `dvips` feature.

The `PaintType` in a Postscript font has just to be changed from the default *filled* **(0)** to *outlined* **(2)**. You have to add the text `"/PaintType 2 store"` to the file `psfonts.map`, as in the following example.

Latin Font Name	Command	Arabic Name	A Little Sample
MicroSoft			
Traditional Arabic	\texttrad	تقليدي	حسن الخط حلية الكاتب
Traditional Arabic Bold	\texttradb	**تقليدي أسود**	**حسن الخط حلية الكاتب**
Simplified Arabic	\textsimpl	مبسط	حسن الخط حلية الكاتب
Simplified Arabic Bold	\textsimplb	**مبسط أسود**	**حسن الخط حلية الكاتب**
Courier	\textcour	كوريي	حسن الخط حلية الكاتب
Courier Bold	\textcourbd	**كوريي أسود**	**حسن الخط حلية الكاتب**
Arial (Times)	\textarial	أريال	حسن الخط حلية الكاتب
Arial (Times) Bold	\textarialbd	**أريال أسود**	**حسن الخط حلية الكاتب**
Andalus	\textandalus	أندلس	حسن الخط حلية الكاتب
Arabeyes Project			
Tholuth	\textthol	الثلث	حسن الخط حلية الكاتب
Yermook	\textyerm	اليرموك	حسن الخط حلية الكاتب
Mashq	\textmash	المشق	حسن الخط حلية الكاتب
Hor	\texthor	الحر	حسن الخط حلية الكاتب
Battar	\textbattar	البتار	حسن الخط حلية الكاتب
Granada	\textgranada	غرناطة	حسن الخط حلية الكاتب
Kayrawan	\textkayrawan	القيروان	حسن الخط حلية الكاتب
Dimnah	\textdimnah	دمنة	حسن الخط حلية الكاتب
Sindibad	\textsindibad	سندباد	حسن الخط حلية الكاتب
Graph	\textgraph	كراف	حسن الخط حلية الكاتب
Nice	\textnice	نايس	حسن الخط حلية الكاتب
Almohanad	\textmohanad	المهند	حسن الخط حلية الكاتب
Almothnna	\textmothnna	المثنى	حسن الخط حلية الكاتب
Almateen	\textmateen	المتين	حسن الخط حلية الكاتب
Petra	\textpetra	البتراء	حسن الخط حلية الكاتب
Nada	\textnada	ندى	حسن الخط حلية الكاتب
Cortoba	\textcortoba	قرطبة	حسن الخط حلية الكاتب
Ostorah	\textostora	أسطورة	حسن الخط حلية الكاتب
Furat	\textfurat	الفرات	حسن الخط حلية الكاتب
Salem	\textsalem	سالم	حسن الخط حلية الكاتب
Shado	\textshado	ظل	حسن الخط حلية الكاتب
Metal	\textmetal	معدن	حسن الخط حلية الكاتب
Tarablus	\texttarablus	طرابلس	حسن الخط حلية الكاتب
Khalid	\textkhalid	خالد	حسن الخط حلية الكاتب
Sharjah	\textsharjah	الشارقة	حسن الخط حلية الكاتب
Hani	\texthani	هاني	حسن الخط حلية الكاتب
Ouhod	\textouhod	احد	حسن الخط حلية الكاتب
Rehan	\textrehan	ريحان	حسن الخط حلية الكاتب
Omega Arabic font			
Omega Arabic Bold	\textomegab	**أوميكا أسود**	**حسن الخط حلية الكاتب**

Table 7.2: The fonts actually available with Arabic

Latin Font Name	Command	Arabic Name	A Little Sample
MicroSoft			
Farsi Simple Bold	`\textfrsimpl`	فارسی أسود	حسن الخط حلية الكاتب
Farsi Simple Outline	`\textfrsimplout`	فارسی مجوف	حسن الخط حلية الكاتب
Simplified Arabic	`\textfrsimplarabic`	مبسط	حسن الخط حلية الكاتب
Simplified Arabic Bold	`\textfrsimplarabicb`	**مبسط أسود**	**حسن الخط حلية الكاتب**
Traditional Arabic	`\textftrad`	تقليدى	حسن الخط حلية الكاتب
Traditional Arabic Bold	`\textftradb`	**تقليدى أسود**	**حسن الخط حلية الكاتب**
Arial (Times)	`\textfrarial`	اريال	حسن الخط حلية الكاتب
Arial (Times) Bold	`\textfrarialb`	**اريال أسود**	**حسن الخط حلية الكاتب**
Andalus	`\textfandalus`	أندلس	حسن الخط حلية الكاتب
FarsiWeb Project			
Nazli	`\textnazli`	نازلی	حسن الخط حلية الكاتب
Nazli Bold	`\textnazlib`	**نازلی أسود**	**حسن الخط حلية الكاتب**
Nazli Outline	`\textnazliout`	نازلی مجوف	حسن الخط حلية الكاتب
Nazli Bold Outline	`\textnazlibout`	نازلی أسود مجوف	حسن الخط حلية الكاتب
Titr Bold	`\texttitr`	**تيتر**	**حسن الخط حلية الكاتب**
Titr Bold Outline	`\texttitrout`	تيتر مجوف	حسن الخط حلية الكاتب
Homa	`\texthoma`	**هما**	**حسن الخط حلية الكاتب**

Table 7.3: The fonts actually available with Farsi

	Roman (n) **b/bx, sl/it, sc**	Typewriter Type (n) **b/bx, sl/it, sc**	Sans Serif (n) **b/bx, sl/it, sc**
`free`	Nazli (N)	Homa (H)	Nazli bold (Nb)
	Titr, N, N	H, H, H	Nb, Nb, Nb
`mscore`	Farsi Simple (FS)	Arial (A)	Simplified Arabic(SA)
	FS out, FS, FS	A bold, A, Andalus	SA bold, SA, SA

Table 7.4: The two font choices for Farsi

For the font `nazli`, for example. You must make a copy of the `nazli.tfm` file into `nazliout.tfm`, create the appropriate `lfenazliout.fd`, just like `lfenazli.fd`, and add a definition of the new `\textnazliout` in the file `farsifnt.sty` like

```
% Nazli Outline (Normal)
\DeclareTextFontCommand{\textnazliout}{\fontfamily{nazliout}\selectfont}
```

What makes the Magic happens is of course the following line, (the first one is here just to guide you, you must enter a line similar to the second one), in `psfonts.map`.

```
nazli Nazli " FarsiWebEncoding ReEncodeFont " <FarsiWebEncoding.enc <nazli.pfb
nazliout Nazli "FarsiWebEncoding ReEncodeFont /PaintType 2 store"
                                        <FarsiWebEncoding.enc <nazli.pfb
```

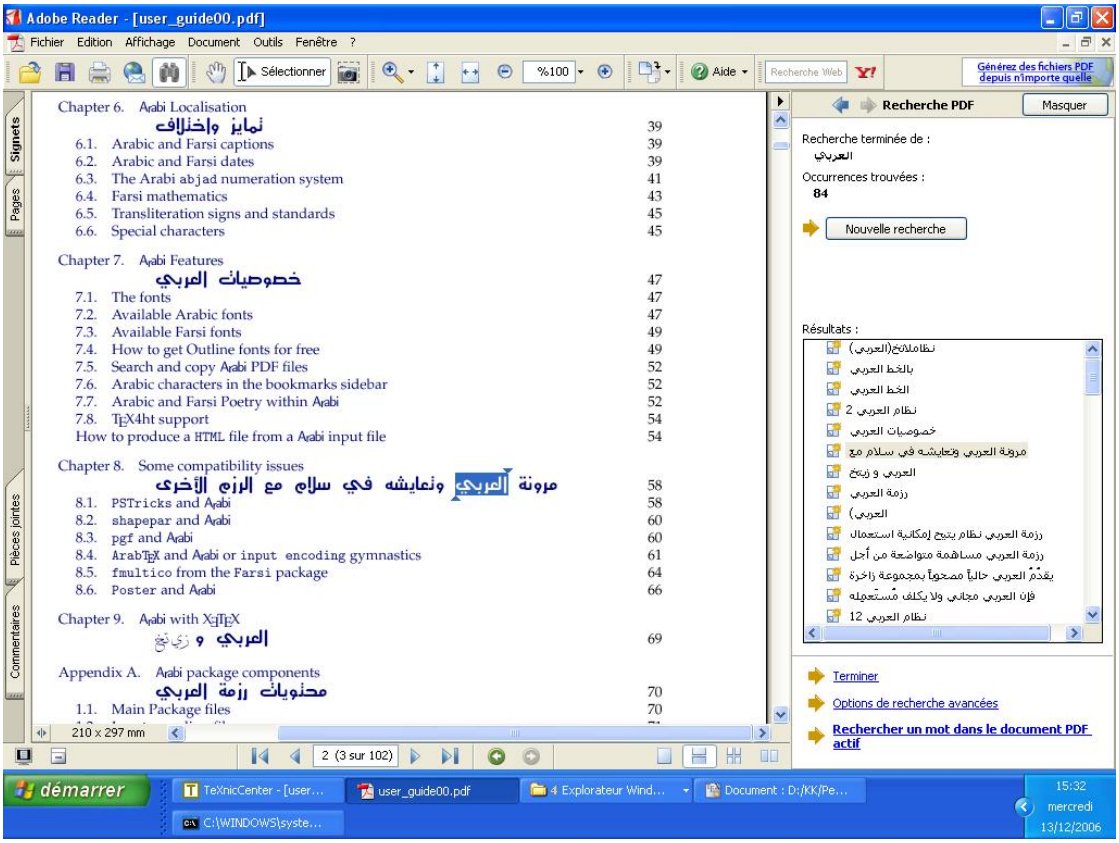


Figure 7.1: Searching Arabi PDF files

7.5. Search and copy Arabi PDF files

The `cmap` package by V. Volovich makes the PDF files generated by pdfTEX both searchable and copyable in acrobat reader and other compliant PDF viewers, for texts encoded in the supported font encodings. For the moment, the version that exists on CTAN supports only the Latin and cyrillic encodings `T1`, `T2A`, `T2B`, `T2C` and `T5`.

Thanks to this package for whom we wrote the needed `cmap` files `lae.cmap` and `lfe.cmap` for the `LAE` and `LFE` encodings that are respectively used by Arabic and Farsi with Arabi. So that now, when using the pdfTEX engine, you can search and copy the generated PDF files.

To get this *features*, you have to call `cmap` package before any other package that may load font encodings. This includes BABEL of course! The best place is just after the `\documentclass` control sequence, for e.g.

```
\documentclass[a4paper]{article}
\usepackage{cmap}
```

This will load the required cmap files with the `CMap translations` in the PDF file to be generated. It works for Type 1 fonts, and also for Type 3 fonts with recent versions of pdfTEX.

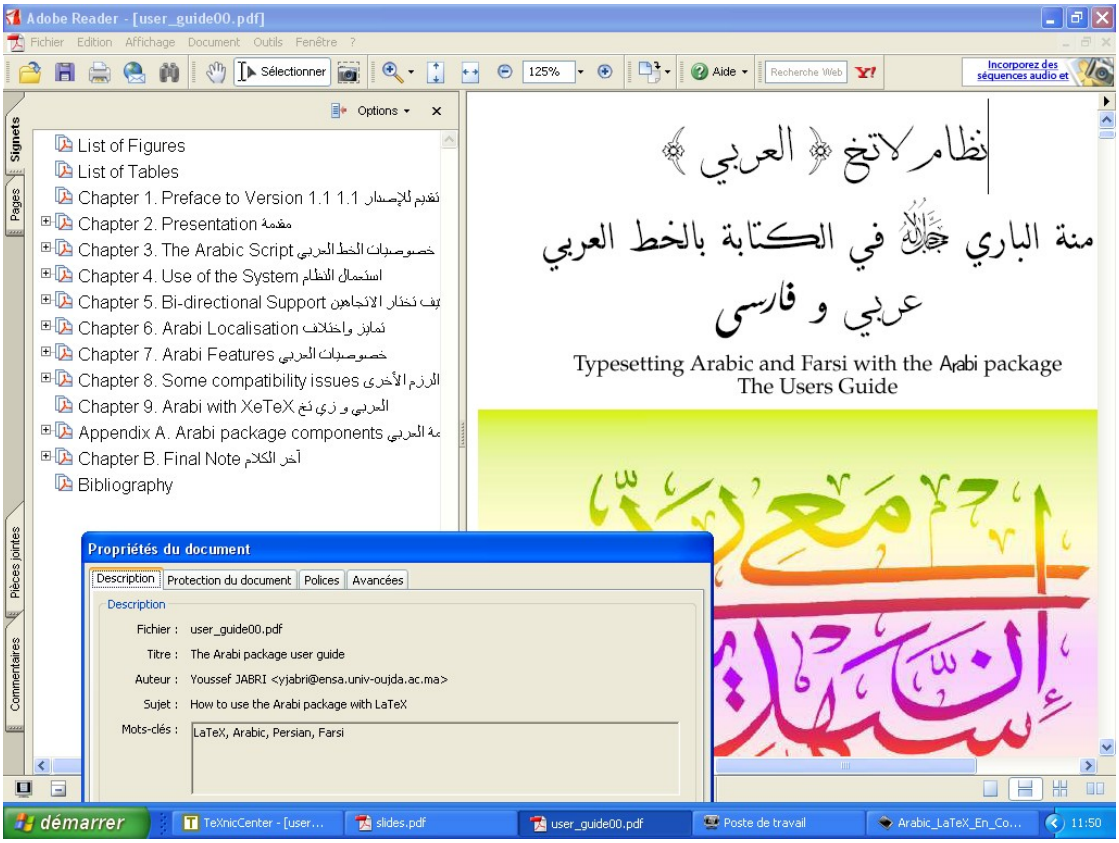


Figure 7.2: Arabic characters in the Bookmarks side bar

7.6. Arabic characters in the bookmarks sidebar

Now we can have Arabic characters (This includes Farsi too of course) in the bookmarks sidebar, when using `hyperref`. The bookmarks are stored in the `.out` file created by `hyperref`. Of course this is done automatically, no postprocessing is needed, but you have to use the `UTF-8` encoding for the Arabic characters.

You have also to call the file `puenc-ar.def` for the moment until its contents is added to the standard file `puenc.def` that works with the `unicode` option of `hyperref`. Notice that `hyperref` is still not supported yet but you can use it if your main document direction is Left-to-Right, as you can see with the Arabi documentation.

7.7. Arabic and Farsi Poetry within Arabi

This section applies to both Farsi and Arabic, so Arabic refers to the text not the language unless the word “language” is explicited.

To write poetry, we use the same idea we proposed once to K. Lagally who implemented it using TEX, in the verses package for ArabTEX. It consists in defining two lengths, one for the size

for a half verse and the second for the space separating the two half verses, and to spread each half verse in a box with the first dimension and separating them with a space (or a connector contained in a box) of width the second dimension.

In ArabTEX there is a command `\spreadbox` that takes two arguments, a legal *width* and an Arabic text to be spread. Unlike Latin text, spreading Arabic text for a given width does not mean adding a space but lengthening the *keshida* character (in fact the connections between connected characters!) In ArabTEX, the keshida is a rule that will get the desired length on demand. With all existing modern (postscript and True type) fonts, there is a glyph (with some fixed width depending on the font of course) that may be used as a keshida. So defining a `\spreadbox` macro like in ArabTEX will not be so easy since we have to put keshidas one after the other with no space between so that the final length of the text to be spread will get *exactly* the desired length. We will certainly get a desired length by stacking characters (with fixed widths). The idea, that was stolen from the `border` macros to typeset borders, a part of the `midnight` package of van der Goot, was to get overlapping keshidas if necessary so that the final width of the `spreadbox` gets exactly the desired one. And we get a `\Spreadbox` with the same syntax that does exactly that, only the constraint is that you have to type some keshida to get to work.

The following commands

```
\Spreadbox{5.5cm}{\textAR{بسـم اللـه الرحمـن الرحيـم}}
```

```
\Spreadbox{6cm}{\textAR{بسـم اللـه الرحمـن الرحيـم}}
```

```
\Spreadbox{8cm}{\textAR{بسـم اللـه الرحمـن الرحيـم}}
```

```
\Spreadbox{10cm}{\textAR{بسـم اللـه الرحمـن الرحيـم}}
```

give the following result:

بســـــم اللـــــه الرحمـــــن الرحيـــــم

بســـــم اللـــــه الرحمـــــن الرحيـــــم

بســـــــــم اللـــــــــه الرحمـــــــــن الرحيـــــــــم

بســـــــــــــم اللـــــــــــــه الرحمـــــــــــــن الرحيـــــــــــــم

To typeset *Arabic or Farsi poetry* with Arabi, you have to load first the `poetry` package, then choose the right dimensions for the half verse and the connector as in

```
\setversedim{75mm}{5mm}
```

Then just write the verses of poetry in two part almost like in ArabTEX, the difference is a capital `H` instead of a lowercase `h`

```
\Halfverses{ إن سمعت بـه ، إذا ما ناعب نعبـا } { لا نفرحن بفأل ولا نطير }
```

```
\Halfverses{ فالخطب أفظع من سراء، نأملهـا } { والأمر أيسر من أن نضمر الرعبـا }
```

...

To get for example:

لا تفرحـــن بفـــأل ، إن سمعـــت بـــه ولا تطيـــر ، إذا ما ناعـــب نعبـــا
فالخطـــب أفظـــع مـــن سراء تأملهـــا والأمـــر أيســـر مـــن أن تضمـــر الرعبـــا
إذا تفكرت فكـــرا ، لا يمازجـــه فســـاد عقل صحيح ، هان ما صعبـــا

You may also use the macro `\Connverses` that gives you the possibility to add a character or something small whose width that does not exceed the reserved dimension to act as an ornament in the middle, like:

```
\Connverses{ فاللب إن صح أعطى النفس فترتها }{*} { حتى تموت ، وسمى جدها لعبا }
```

...

To get:

فاللب إن صـــح أعطى النفس فترتهـــا ❋ حتى تمـــوت ، وسمى جدها لعبـــا
وما الغـــواني الغـــوادي ، في ملاعبهـــا ❋ إلا خيالات وقـــت ، أشبهـــت لعبـــا
زيـــادة الجسم عنت جسم حاملـــه ❋ إلى التـــراب ، وزادت حافرا تعبـــا

7.8. TEX4ht support

Documents written using LATEX have a similar form to HTML documents used on the world wide web. They both markup languages with similar structures.

There are many tools that are used to convert TEX/LATEX documents to HTML. They are quite different and do not have similar capabilities. If you are interested you should get a look at *The LATEX Web Companion*, [19] by Goosen *et al*.

These can be divided into two classes:

1. Systems that parse the `*.tex` file themselves.
2. Systems that use TEX as the file parser.

In fact, they all do their own parsing, except TEX4ht. It is the only one that uses TEX as the parser. The first category will certainly have a lot of trouble to understand and reproduce correctly all the commands a user may use and this certainly the case of new systems like Arabi. This makes TEX4ht a serious option that cannot be easily discarded. And indeed, it did not took more than few lines of TEX Code and the creation of appropriate hyperfont files for Arabi to get supported by TEX4ht!

With this version 1.1 of Arabi, we added the necessary files to add the possibility to convert Arabi files to HTML and XML easily using TEX4ht of E. Gurari. Get a look to http://www.cis.ohio-state.edu/~gurari/TeX4ht. It is still experimental, just like the whole Arabi at this stage, but seems to work quite well on the examples we tried till now. Arabi support will be a part of future TEX4ht distributions.

A little description of how to get HTML files from Arabi LATEX files follows.

Notice first that until some *official* version of TEX4ht is released, we will provide some files that do that. The way professor Gurari does with TEX4ht is certainly the best as he know his system better any one else.

TEX4ht and Arabi العربي و تخ فور إنش تي

```
\documentclass[a4paper]{book}

\usepackage[utf8]{inputenc}
\usepackage[LFE,LAE,OT1]{fontenc}
\usepackage[farsi,english,arabic]{babel}

\Novocalize

\begin{document}
%\selectlanguage{arabic}  %the default in this case

\title{%
شمس النهار%
}

\author{توفيق الحكيم}

\maketitle
\tableofcontents
\chapter{توفيق الحكيم}

في هذا الشهر، يحتفل العالم العربي بالذكرى المئوية لمولد توفيق الحكيم ( \I{1898} الإسكندرية - \I{1987} القاهرة ).
وقد حظي الحكيم بمكانة متميزة على خريطة تطور الكتابة الإبداعية العربية؛ فهو كاتب واحدة من أولى الروايات العربية: عودة الروح (\I{1933}). وهو - أيضا - كاتب أول مسرحية عربية ناضجة بمعايير النقد الحديث: أهل الكهف (\I{1933}). وقبلها، كان الحكيم قد كتب أولى مسرحياته في سن مبكرة بعنوان الضيف الثقيل . وذلك إبان ثورة \I{1919} ضد الاحتلال البريطاني.
وصف الحكيم بأنه مؤسس المسرح المصري المعاصر . وبأنه من جعل المسرح في العالم العربي جنسا أدبيا مرموقا . وقد انسمت لغة أعماله المسرحية باليسر والمرونة، وحرص على أن يبرز فيها روح
...

\end{document}
```

Figure 7.3: TEX4ht and Arabi input

If you want this functionality, you should of course install the TEX4ht package if it is not yet installed on your system. First we thought to distribute the needed files ourselves, and the user would have to copy all these files (*hyperfont files* and a file `arabicore.4ht` among others). These files should have been removed when you would install an official version of TEX4ht that supports Arabi, that should be more up-to-date certainly, to avoid any clash or misuse! But since there is a possibility to get the last version of the TEX4ht distribution from the web at http://www.cis.ohio-state.edu/~gurari/TeX4ht/bugfixes.html, this seems us not necessary. The

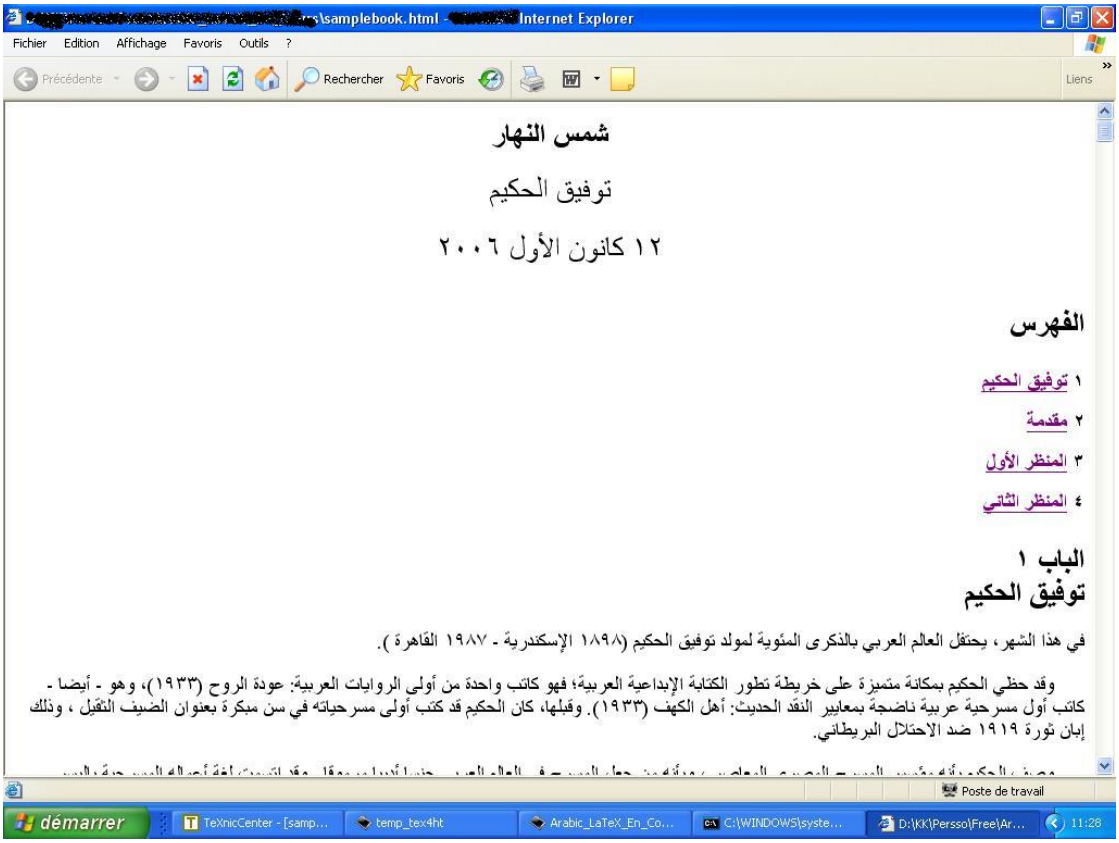


Figure 7.4: The file `samplebook.html` as shown in a web browser

link points to the last changes to TEX4ht, including possibly, additions that are not yet described in this documentation. Just just download that version and update your TEX4ht files.

How to produce a `HTML` file from a Arabi input file

We will proceed by a small example that looks like the following figure 7.3 (see the file `samplebook.tex`):

Then, just issue this command in a shell (where `samplebook.tex` is replaced by your `filename` of course):

```
htlatex samplebook.tex
```

For UTF-8 compilations, however, as pointed out by Gurari, one should use commands like the following

```
htlatex␣samplebook.tex␣"xhtml,uni-html4"␣"␣-cunihtf␣-utf8"
```

TEX4ht will call LATEX to compile your Arabi LATEX file three (3) times to resolve *all* LATEX possible cross references (more may be needed some times, you are the only one who can decide) and

Figure 7.5: The file `samplebook.html` with a slightly modified CSS

then calls the programs `tex4ht` and `t4ht` in a final stage to generate the corresponding `HTML` file `samplebook.html` with possibly images to be embedded in it and a *cascading style sheet* file `samplebook.css`.

This looks like Figure 7.4 on page 57 on a Web browser.

You can change the layout of your HTML file by adding *cascading style sheet* (`CSS`) directives in a configuration file (see the TEX4ht documentation please) to customize your output. Here (see Figure 7.5) is what we get with the former example by adding just few lines that will be added to `samplebook.css`.

CHAPTER 8

Some compatibility issues
مرونة العربي وتعايشه في سلام مع الرزم الأخرى

We will see now some packages that were tested to work with Arabi with (may be) no problems!

8.1. PSTricks and Arabi

Concerning, Postscript, the system works perfectly well with the PSTricks macros and we found really no problem! You may get a look at different figures used in this user guide produced using PSTricks. Get a look at Figure 8.1 on page 59.

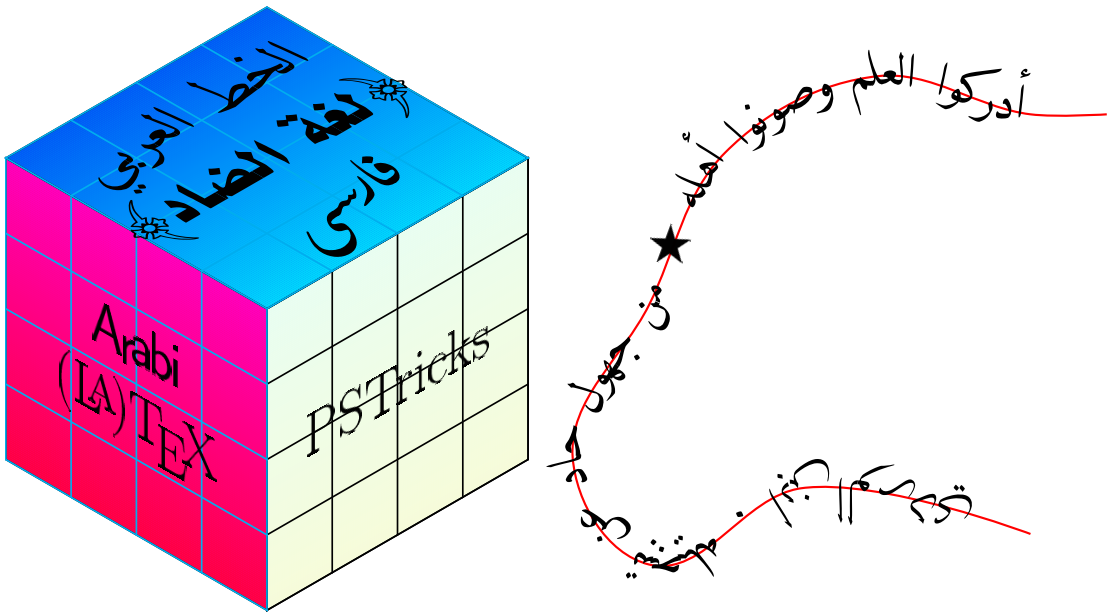


Figure 8.1: PSTricks and Arabi

The two figures were produced with the following standard PStricks code:

PStricks and Arabi العربي و بستراىكس

```
\begin{figure}[htbp]
  \centerline{%
  \scalebox{1.1}{
   \begin{pspicture}(6,6)
    \psset{unit=1.1cm,dimen=middle,viewpoint=-1 -1 1}
    \ThreeDput[normal=0 -1 0](2,-2,0)
    {
    \psframe[fillstyle=gradient,gradmidpoint=1,%
      gradend=LemonChiffon,gradbegin=LightCyan](4,4)
    \psgrid[unit=1.1cm,subgriddiv=0,gradmidpoint=.5,%
      gradend=cyan,gridlabels=0](4,4)
    \rput(2,2){\Huge PSTricks}
    }
    %
    \ThreeDput[normal=-1 0 0](2,2,0)
    {
    \psframe[fillstyle=gradient,gradmidpoint=1,%
      gradend=red,gradbegin=magenta](4,4)
    \psgrid[unit=1.1cm,subgriddiv=0,gridcolor=cyan,gridlabels=0](4,4)
      \rput(2,2.5){\Huge \Arabi}
      \rput(2,1.5){\Huge \AllTeX}
    }
    %
    \ThreeDput[normal=0 0 1](4,0,2)
    {
    \psframe[fillstyle=gradient](4,4)
    \psgrid[unit=1.1cm,subgriddiv=0,gridcolor=cyan,gridlabels=0](4,4)
    \rput(2,2){\Huge \begin{tabular}{c}\textAR{اللغة}\\
    \textAR{العربية}\\ \textAR{\textmateen{[ لغة الضاد ]}}\end{tabular}}
    }
  \end{pspicture}
  }
%
%
  \begin{pspicture}(8,8)
    \psset{linecolor=red}
    \pstextpath[c]
    {\pscurve(6,1)(3,1.6)(1,0.6)(0,2)(1,4)(2,6)(4,7)(6,6.5)(7,6.5)}%
    {\LARGE \textAR{ أدركوا العلم وصونوا أهله  $\bigstar$ %
    من جهول حاد عن نبجيله . \texttradb{ابن الوردي}}}
  \end{pspicture}
}
%
\caption{\pkg{PSTricks} and \pkg{\Arabi}}
%
\end{figure}
```

Figure 8.2: PSTricks and Arabi input

8.2. `shapepar` and Arabi

Arabi works also fine with Donald Arseneau's `shapepar` package. You must have noticed the `NUTSHELL` figure on page 21. For fun, we have developed a new paragraph shape with the form of a `MOSQ DOOR`. See Figure 8.3 on page 61.

Figure 8.3: `MOSQ DOOR`. Arabi and the `parshape` package II

8.3. `pgf` and Arabi

Arabi is compatible with the `pgf` package `http://sourceforge.net/projects/pgf` by Tantau [24], that provide "similar" features to `PSTricks`. Of course, it does not accept raw PostScript

as PSTricks does and may lack some other functionalities provided by packages that use PostScript through PSTricks to do PostScript programming and calculations. For now, TEX can do some calculations, but nothing similar to PostScript of course!

The main advantage of pgf over PSTricks seems to be the possibility to use any TEX engine, including pdfTEX.

The documentation of Arabi 1.0 was prepared with the help of PSTricks. The dvi file was converted to PostScript using dvips. And the PostScript resulting file was finally converted to PDF using Ghostscript.

This scheme, unfortunately, will end in a PDF file that cannot be searched or copied except if we use fonts with glyphs that contain Unicode information. This is not the case of Arabeyes fonts for the current version. This is a drawback that can be overcome using cmap files for the LAE and LFE font encodings and the cmap package.

For this version 1.1 of Arabi, the documentation was prepared differently. We tried to use pgf instead of PSTricks whenever possible, just to be able to use pdfTEX and hence to be able to use the cmap package and finally get a PDF file where Arabic and Farsi text can be searched and copied!

1. When some functionality was provided by both pgf and PSTricks with a similar quality, we used pgf. This is the case for example of colored boxes with text inside.
2. When some functionality was provided only by PSTricks, or some package that used it, we generated corresponding encapsulated PostScript files and converted them to PDF to use them with pdfTEX.
3. And finally, we used pdfTEX with pgf and the PDF images output by PSTricks. That way, cmap also works for us, and our Arabic and Farsi text is searchable and can be copied!

8.4. ArabTEX and Arabi or input encoding gymnastics

If you are a LATEX user that needs Arabic or Farsi, you may have already used the system ArabTEX by Prof K. Lagally, personally, I did and I liked it. Cheers!

In the sequel Arabic means Arabic or Farsi. If you used either Windows CP 1256, or ISO 8859-6 or Unicode UTF-8 input encodings to write your Arabic text with ArabTEX, you will have no problem to use your documents with Arabi since it supports these encodings.

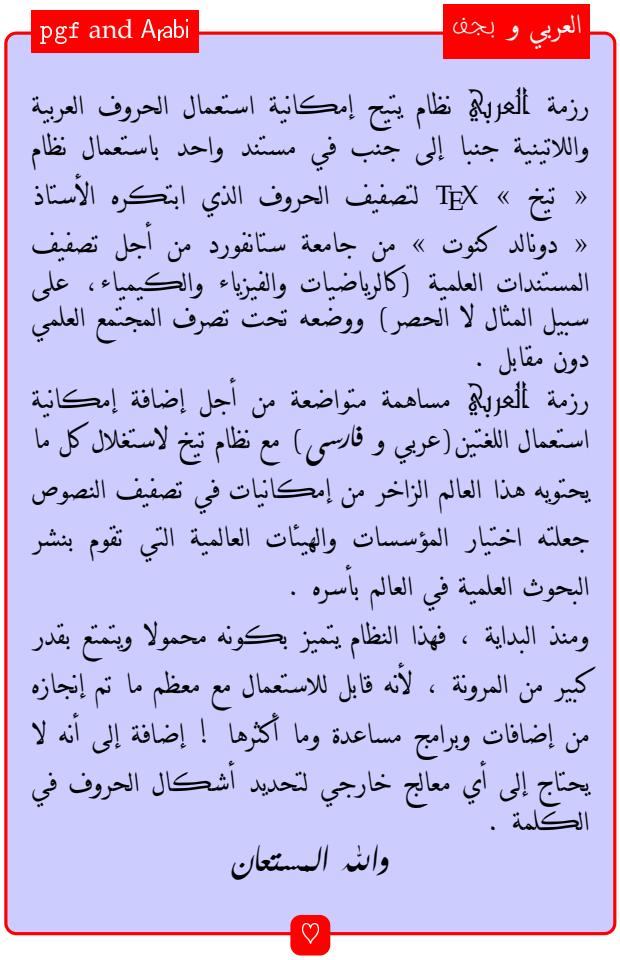


Figure 8.4: `pgf` and Arabi

If you were using the `ASCII` ArabTEX encoding, or you need a functionality that is not yet implemented in Arabi or that ArabTEX does better, but you are interested in getting your Arabic text typeset with Arabi, either for the available fonts, or just because you need to write your text on some `textpath` with `PSTricks` for example...

Then, you have always the possibility to use your ArabTEX with Arabi in your documents since Arabi is **fully** compatible with ArabTEX. You can get a look at the following example.

Nevertheless, you must not use the < and > symbols in your text. They are used in Arabi to get the *left* and *right* Arabic double quotes!

There is only a little thing you have to bear in mind. ArabTEX supports Windows `CP 1256`, `ISO 8859-6` and Unicode `UTF-8` encodings as well as the Mac Arabic encoding. There is only one thing one should know. ArabTEX does not use the standard `inputenc` for that. It has its own macros.

So, when you want to use 8-bit Arabic or Farsi text in some format, `CP 1256`, for example in both arabtext environments and outside with Arabi, a clash will occur, both `inputenc` and ArabTEX complaining! While you can use the `arabtext` environment with ASCII input encoding without any problem!

The solution to this *apparent* problem is the following easy trick. Remember that ArabTEX uses the control sequence

`\setcode{encoding}`

where `encoding` is either `arabwin` or `cp1256` for Windows `CP 1256`, `utf8` for `UTF-8`, or `iso8859-6` for `ISO 8859-6`. See the ArabTEX documentation for the details!

While, to switch between different encodings, `inputenc` uses the control sequence

`\inputencoding{encoding}`

So the solution is just to issue, at the beginning of you document either

```
\setcode{arabtex}
```

or

```
\setcode{none}
```

to hide Arabic characters you may type to ArabTEX.

And before any `arabtext` environment, type a command like

```
\inputencoding{latin1}
```

you may use `latin1` or any other input encoding that does not support Arabic characters so that any Arabic character you may type inside the `arabtext` environment will be invisible to Arabi and LaTeX 2ε in general. And just inside issue a command

```
\setcode{cp1256}
```

of course you have to use the encoding you are using to type Arabic characters instead of `cp1256`.

And after the `arabtext` environment, type a command like

```
\setcode{arabtex}
\inputencoding{cp1256}
```

or any input encoding other than `CP 1256` you may use!

Get a look at the Examples 8.5 on page 65 and 8.6 on page 66.

But don't forget after to deactivate ArabTEX Arabic input encoding and activate that of the package `inputenc`:

```
\inputencoding{cp1256}
\setcode{arabtex}
```

The following text is produced in an `arabtext` environment with the options `\transtrue` and `\setmaghribi` and `\setcode{arabtex}`. Some special characters input with two ASCII characters in ArabTEX and few verses from معلقة امرئ القيس are typeset.

ArabTEX and Arabi

العربي و عرب تخ

As an example the following text with Arabi using `CP 1256`**:**

```
\begin{center}
\selectlanguage{arabic}
{\Huge\dthuluth
مثال من رزمة \textLR{\Arabi}
لاحظ الخط
}
\end{center}
```

gives the following result:

مثال من رزمة Arabi لاحظ الخط

And now the following code with ArabTEX using the same `CP 1256`**:**

```
\selectlanguage{english}
\inputencoding{latin1}
\begin{center}
\begin{arabtext}
\setcode{cp1256}
{\Huge
مثال من رزمة <\ArabTeX>
لاحظ الخط
}
\end{arabtext}
\end{center}
```

gives the following result:

مثال من رزمة ArabTEX لاحظ الخط

Figure 8.5: Arabi and ArabTEX side by side

8.5. `fmultico` from the `Farsi` package

A modified version of `multicol` version 1.4 that works from Right-to-Left. Unfortunately, no comment is given to indicate where the changes were made to the original file.

It works well and may produce text for at most 5 columns.

We typeset an example with respectively 3 and 4 columns! Text in 5 columns looks too ugly in our case because the `linewidth` is *too small* and hyphenation is forbidden in modern Arabic! This should work better for larger pages! This works also better with fonts whose characters are

ArabTEX inside Arabi

عرب تخ من وسط العربي

ā ḥ ṭ ḏ š ǧ ṣ ḍ ġ ḫ ṭ ẓ

ى خ ث ذ ش ج ص ض غ ح ط ظ

b-ism al-lāh wa-'l-ḥmd li-llāh wa-'lṣlāh wa-'s-slām ʿlā rsūl al-lāh.

باسم الله وَالحمد لِلّه وَالصلَاة وَالسّلَام علَى رسُول الله.

muʿallaqatu āmrāʾ al-qays

مُعَلَّقَةُ امرىء القَيْس

qifā nbki mn ḏikrā ḥbybin wamanzili bsiqṭi 'l-liwā bayna 'd-daḫūli faḥawmali

قِفَا نبكِ من ذِكرَى حبيبٍ وَمَنزِلِ بِسِقْطِ اللِّوَى بَينَ الدَّخُولِ فَحَومَلِ

fatūḍiḥa f-al-miqrāti lam yaʿfu rasmuhā lmā nasağathā mn ğanwbin wšamʾali

فَتُوضِحَ فَالمِقْرَاةِ لَمْ يَعْفُ رَسْمُهَا لِمَا نَسَجَتْهَا من جَنُوبٍ وَشَمْأَلِ

trā baʿara ālʾarāmi fā ʿaraṣātihā wqyʿānihā kʾannahu ḥabbu fulfuli

تَرَى بَعَرَ الأَرْآمِ فِي عَرَصَاتِهَا وَقِيعَانِهَا كَأَنَّهُ حَبُّ فُلْفُلِ

kʾanny ġadāta ālbayni yawma taḥammalū ldā samurāti ālḥayyi nāqifu ḥanḍli

كَأَنِّي غَدَاةَ البَينِ يَومَ تَحَمَّلُوا لَدَى سَمُرَاتِ الحَيِّ نَاقِفُ حَنظَلِ

wqūfan bihā ṣaḥby ʿalā maṭiyyahum, yqwlwn lāthlik ʾasā'n wtağammali

وُقُوفًا بِهَا صَحبِي عَلَى مَطِيَّهُم، يَقُولُونَ لَاتَهلِك أَسًى وَتَجَمَّلِ

wa-ʾinna šifāʾī ʿabratun muharāqatun fhl ʿnda rasmin dārisin mn maʿwali

وَإِنَّ شِفَائِي عَبرَةٌ مُهَرَاقَةٌ فَهَلْ عِندَ رَسمٍ دَارِسٍ من مَعوَلِ

kdʾabka mn ʾummi 'l-ḥuwayriṯi qablhā wğāratihā ʾumma 'r-rabābi bi-mʾasali

كَدَأبِكَ من أُمِّ الحُوَيرِثِ قَبلَهَا وَجَارَتِهَا أُمِّ الرَّبَابِ بِمَأسَلِ

Figure 8.6: ArabTEX at work inside Arabi

more or less the same height and width. For example, with Arabic, this works better with `al mohanad` than with `traditional Arabic`!

مقدمة المؤلف رب يسر وأعن وصلى الله على محمد وآله وسلم قال الشيخ الإمام العالم العلامة شمس الدين أبو عبد الله محمد بن أبي بكر بن أيوب الزرعي الحنبلي إمام الجوزية رحمه الله هذا كتاب سميته جلاء الأفهام في فضل الصلاة والسلام على محمد خير الأنام وهو خمسة أبواب وهو كتاب فرد في معناه لم يسبق إلى مثله في كثرة فوائده وغزارتها بينا فيه الأحاديث الواردة في الصلاة والسلام عليه وصحيحها من حسنها ومعلولها وبينا ما في معلولها من العلل بيانا شافيا ثم أسرار هذا الدعاء وشرفه وما اشتمل عليه من الحكم والفوائد ثم في مواطن الصلاة عليه ومحالها ثم الكلام في مقدار الواجب منها واختلاف أهل العلم فيه وترجيح الراجح وتزييف المزيف ومخبر الكتاب فوق وصفه والحمد لله رب العالمين. مقدمة المؤلف رب يسر وأعن وصلى الله على محمد وآله وسلم قال

الشيخ الإمام العالم العلامة شمس الدين أبو عبد الله محمد بن أبي بكر بن أيوب الزرعي الحنبلي إمام الجوزية رحمه الله هذا كتاب سميته جلاء الأفهام في فضل الصلاة والسلام على محمد خير الأنام وهو خمسة أبواب وهو كتاب فرد في معناه لم يسبق إلى مثله في كثرة فوائده وغزارتها بينا فيه الأحاديث الواردة في الصلاة والسلام عليه وصحيحها من حسنها ومعلولها وبينا ما في معلولها من العلل بيانا شافيا ثم أسرار هذا الدعاء وشرفه وما اشتمل عليه من الحكم والفوائد ثم في مواطن الصلاة عليه ومحالها ثم الكلام في مقدار الواجب منها واختلاف أهل العلم فيه وترجيح الراجح وتزييف المزيف ومخبر الكتاب فوق وصفه والحمد لله رب العالمين.

مقدمة المؤلف رب يسر وأعن وصلى الله على محمد وآله وسلم قال الشيخ الإمام العالم العلامة شمس الدين أبو عبد الله محمد بن أبي بكر بن أيوب الزرعي الحنبلي إمام الجوزية رحمه الله هذا كتاب سميته جلاء الأفهام في فضل الصلاة والسلام على محمد خير الأنام وهو خمسة أبواب وهو كتاب فرد في معناه لم يسبق إلى مثله في كثرة فوائده وغزارتها بينا فيه الأحاديث الواردة في الصلاة والسلام عليه وصحيحها من حسنها ومعلولها وبينا ما في معلولها من العلل بيانا شافيا ثم أسرار هذا الدعاء وشرفه وما اشتمل عليه من الحكم والفوائد ثم في مواطن الصلاة عليه ومحالها ثم الكلام في مقدار الواجب منها واختلاف أهل العلم فيه وترجيح الراجح وتزييف المزيف ومخبر الكتاب فوق وصفه والحمد لله رب العالمين.

8.6. `Poster` and Arabi

It works fine too with the `poster` package by Timothy Van Zandt, the creator of PSTricks, that may produce many normal (small) pages with parts of the text to be "glued" together to get a larger one. Look, beginning from the next page, at the following small example that spreads on 8 pages as you can see on the next page (a landscape page made of the 8 produced pages stacked, in a 24 disposition, besides each other) and whose code follows on the page after.

﴿ كلمات من مشرقة ﴾

. . . فجعل لل الله الإيمان تطهيرا لكم من الشرلشرك ، والصلاة تنزيها لكم عن الك الكبر ، والزكاة تزكية لبة للنفس ونماء في الرزق ، والصيام يام تثبيتا للإخلاص ، والحج تشييدا يبدا للدين ، والعدل تنسيقا يقا للقلوب ، وطاعتنا نظاما للملة ، ، وإمامتنا أمانا من الفرقة والجهاد عز عزا للإسلام وذلا لأهل الـالكفر والنفاق ، والصبر معونة علىعلى استيجاب الأجر ، والأمر بالمعروبعروف والنهي عن المنكر كر مصلحة للعامة ، وبر الوالدين وقا وقاية من السخط ، وصلة الأرحام ام منسأة في العمر ومنماة فاة في العدد ، والقصاص حقنا للدماءماء ، والوفاء بالنذر تعريضا للمغفرة نرة ، وتوفية

العمر ومنماة فاة في العدد ، والقصاص حقنا للدماءماء ، والوفاء بالنذر تعريضا للمغفرة نرة ، وتوفية
المكاييل والـ والموازين تغييرا للبخسة ، والنهي عن عن الخمر تنزيها عن الرجس ، واجتناجتناب القذف حجابا عن اللـ اللعنة ، وترك السرقة ايجابا للعفة نة ، وحرم الله الشرك إخلاصا له باه بالربوبية ، فاتقوا الله حقحق تقاته ولا تموتن إلا وأنتم مسلمون ن ، وأطيعوا الله فيما أمركم به ونهاكم اكم عنه فإنما يخشى الله من من عباده العلماء . . .

الزهراء فاطمة بنت محمد (. (صلى الله عليه وآله)

poster and Arabi

العربي و بوستر

```
\begin{Poster}[vcenter=true,hcenter=true,clip=pstricks]
\font\bigtrad=DTPN at 2cm
\setlength{\fboxsep}{.8truein}
\setlength{\fboxrule}{.1truein}
\fbox{\begin{minipage}{22.2truein}
$$
\psframebox[framesep=5mm,linewidth=0.7mm,framearc=.18,fillstyle=solid,
fillcolor=LemonChiffon]{
\parbox{.9\hsize}{\selectlanguage{arabic}
\centerline{\textAR{\bigtrad [  كلمات مشرقة  ]}}
\bigtrad

... فجعل الله الإيمان تطهيرا لكم من الشرك . والصلاة تنزيها لكم عن الكبر . والزكاة تزكية للنفس ونماء
في الرزق . والصيام تثبيتا للإخلاص . والحج تشييدا للدين . والعدل تنسيقا للقلوب . وطاعتنا نظاما للملة .
وإمامتنا أمانا من الفرقة والجهاد عزا للإسلام وذلا لأهل الكفر والنفاق . والصبر معونة على استيجاب الأجر
. والأمر بالمعروف والنهي عن المنكر مصلحة للعامة . وبر الوالدين وقاية من السخط . وصلة الأرحام
منسأة في العمر ومنماة في العدد . والقصاص حقنا للدماء . والوفاء بالنذر تعريضا للمغفرة . وتوفية المكاييل
والموازين تغييرا للبخسة . والنهي عن الخمر تنزيها عن الرجس . واجتناب القذف حجابا عن اللعنة .
وترك السرقة ايجابا للعفة . وحرم الله الشرك إخلاصا له بالربوبية . فاتقوا الله حق تقاته ولا تموتن إلا وأنتم
مسلمون . وأطيعوا الله فيما أمركم به ونهاكم عنه فإنما يخشى الله من عباده العلماء ...
\centerline{\textAR{\bigtrad ( صلى الله عليه وآله ) الزهراء فاطمة بنت محمد}}
}}
$$
\end{minipage}}
\end{Poster}
```

♡

Figure 8.7: Arabi and poster coding

ى الكبر ،
يدا للدين ،
عزا للإسلام
عروف والنهي
ام منسأة في
فرة ، وتوفية

ك مشرقة

لشرك ، والصلاة تنزيها لكم عن الك
يام تثبيتا للإخلاص ، والحج تشييدا
، وإمامتنا أمانا من الفرقة والجهاد عز
على استيجاب الأجر ، والأمر بالمعرو
وقاية من السخط ، وصلة الأرحام
دماء ، والوفاء بالنذر تعريضا للمغفرة

كلمات م

عل الله الإيمان تطهيرا لكم من الشر
ية للنفس ونماء في الرزق ، والصيام
يقا للقلوب ، وطاعتنا نظاما للملة ،
الكفر والنفاق ، والصبر معونة على
كر مصلحة للعامة ، وبر الوالدين وقا
انا في العدد ، والقصاص حقنا للدماء

. . . فجعل ا
والزكاة تزكية ل
والعدل تنسيقا
وذلا لأهل الك
عن المنكر
العمر ومنماة ف

ا ي
فرة ، وتوفية
جتناب القذف
ه بالربوبية ،
اكم عنه فإنما

دماء ، والوفاء بالنذر تعريضا للمغفرة
عن الخمر تنزيها عن الرجس ، واجتنا
نة ، وحرم الله الشرك إخلاصا له با
ن ، وأطيعوا الله فيما أمركم به ونهاكم

د (صلى الله عليه وآله)

ة في العدد ، والقصاص حقنا للدماء
والموازين تغييرا للبخسة ، والنهي عن
اللعنة ، وترك السرقة ايجابا للعفة
حق تقاته ولا تموتن إلا وأنتم مسلمون
من عباده العلماء . . .

الزهراء فاطمة بنت محمد (

العمر ومنماة ف

المكاييل والم

حجابا عن الل

فاتقوا اللّٰه حق

يخشى اللّٰه من

CHAPTER 9

Arabi with XeTeX
العربي و زي تخ

Arabi specific support for the XeTeX engine and its documentation are under *preparation*. Nevertheless, You can use Arabi with XeTeX as it is a TeX engine that has the four primitives needed and used by Arabi to do Right-to-Left and Left-to-Right writing.

XeTeX[1] is a TeX system by Jonathan Kew that began with Mac OS X on modern Apple Mac machines and which exists now for Linux and Windows flavored machines is a TeX typesetting engine that natively handles the Unicode character set and modern *intelligent* and includes ε-TeX extensions. It produces an extended `dvi` format that is transformed by a `dvitopdf` like utility to `PDF`.

XeTeX simplifies languages support greatly, it can work with Asian, Middle Eastern, including the languages that use the Arabic script like Arabic and Farsi and other traditionally "difficult" languages just as readily as with European languages.

According to the system on which it runs, it uses *modern font technologies provided by today's operating systems and text layout services* to quote Jonathan Kew.

It uses *OpenType* or/and *Apple Advanced Typography* (AAT) layout features (for example, for the Arabic script case, using appropriate tables all the contextual analysis and diacritics placement in the font) in modern fonts to support complex nonLatin writing systems.

Arabi uses too the idea used by XeTeX that consists on relying on the font to do, for the example of languages that use the Arabic script, the contextual analysis (shaping) and reserves TeX macros to formatting and localization of LaTeX captions.

We are actually working on making Arabi also XeTeX compliant, It will use XeTeX capabilities to handle fonts and Unicode text with the actual Arabi formatting and localization macros.

[1] The name XeTeX was inspired by the idea of a Mac OS X extension (hence the 'X' prefix) to ε-TeX; and as one of its intended uses is for bidirectional scripts such as Hebrew and Arabic, the name was designed to be reversible. Thee name is pronounced as if it were written zee-TeX.

APPENDIX A

Arabi package components
محتويات رزمة العربي

عليك بأخلاق الكرام فإنها تديم لك الذكر الجميل مع النعم

The Arabi system provides the following packages and files for the Arabic script support:

1.1. Main Package files

`arabic.ldf` This is the Language Definition File for the Arabic language that supports the BABEL system. It defines all the language specific macros like Arabic captions (the names of the table of contents, list of figures, list of tables, ...), The Arabic form of the date, etc.

`farsi.ldf` This is the Language Definition File for the Farsi language that supports the BABEL system. It defines all the Language specific macros for Farsi too like Farsi captions. It defines also the Farsi form of the date, etc.

`arabicore.sty` This file contains the script specific definitions shared both by Arabic and Farsi.

`arabifnt.sty` This file contains the specific definitions of the font names used by Arabi to typeset Arabic.

`arabnovowel.sty` This package defines two macros `\Vocalize` and `\Novocalize` for activating and deactivation vowelization from being rendered in the output, even if it has been typed in the source file. They can be called anywhere in the text, the number of times they are needed. They work both for Arabic and Farsi.

`poetry.sty` This package defines a set of macros for formatting Arabic and farsi poetry. It's an immediate rewriting of the package `verses.sty` from ArabTEX by K. Lagally in such a way to get *both packages* working within Arabi, so that ArabTEX may still be used with Arabi. It contains also a macro `\Spreadbox` that spreads a given text to some declared dimension provided that at least a keshida character has been typed somewhere in that text. Otherwise, it will just insert space between words to get the desired width as with Latin text.

`tranlit.sty` This package provides the *experimental* Arabic transliteration module that renders transliteration of Arabic text if Arabic text is typed *without* any language switching command, just inside the Latin text. (It's a test version and may change).

`arabic.cfg` This configuration file for `arabic.ldf`. If you want to customize `arabic.ldf`, please DO NOT hack into the code, copy this file into a directory searched by TEX, preferably a personal one on multi-user systems, and customize it as you like.

But be careful. If you exchange your documents with colleagues using a different TeX installation, it is best **not to have** a `arabic.cfg` file, and add instead the customization commands to the preamble of your documents after BABEL and Arabi have been loaded.

`bblopts.cfg` This configuration file adds Arabic and Farsi to the "declared" options of BABEL.

1.2. Input encoding files

Arabic input encoding definition files that are used with the Arabi system are:

`8859-6.def`, `cp1256.def` These definition files provide Arabic input encodings for the norms `ISO 8859-6` and MS Windows `code page 1256`. See § 4.2 on page 23 for more.

`laeenc.dfu`, `lfeenc.dfu` These files are the `UTF-8` support ones for the standard `inputenc` package. See also § 4.2 on page 23 for more.

1.3. Font encoding files

The Local Arabi font definitions files are:

`laeenc.def` contains the *actual* `Local Arabic font Encoding`, related font definition files and commands that provide Arabic font switching are defined in `arabifnt.sty`.

`lfeenc.def` contains the *actual* `Local Farsi font Encoding`, related font definition files and commands that provide Arabic font switching are defined in `arabifnt.sty`.

Both are not final and may/will change in next versions.

1.4. Specific files to access Fonts

`psfonts.map` The necessary information for `dvips` mapping real fonts names, encodings and TEX fonts names.

`<Encoding_Name>.enc` contains Postscript Vector Encodings related to the names of glyphs given by Manufacturers in their respective fonts. There are so much, that one can believe hardly in standardisation! The names of the encoding files used by Arabic begin with `AR` and the names of those used by Farsi begin with `FR`.

`lae<font_pfb_name>.fd` The font description files for the many fonts used by the system.

Files added with version 1.1

1.5. TeX4ht support files

`arabicore.4ht` This file contains the necessary corrections and changes in the Arabi specific TEX code to avoid problems with TEX4ht.

`*.htf` This are the hyperfont files for TEX4ht that map glyphs contained in the real fonts to their equivalent in specific code pages for use with `HTML`.

1.6. CMap translation files

`lae.cmap` This is the CMap translation file for the Local Arabic Encoding used by Arabi for the `cmap` package to be able to search and copy Arabic text in `PDF` files.

`lfe.cmap` And this is the CMap translation file for the Local Farsi Encoding used by Arabi for the `cmap` package to be able to search and copy Farsi text in `PDF` files.

`puenc-ar.def` A file, with the necessary additions that should be added to the definition file `puenc.def` used by `hyperref`, to get Arabic and Farsi on the bookmarks side bar, as in this document.

`changes.txt` A change file in text format, with the last changes to the package.

APPENDIX B

Final Note
آخر الكلام

2.1. Acknowledgment

I would like to thank all the people who encouraged me and supported me during the development of this package. I mention especially my wife and my little daughter. This package is a project done by the author, who is a mathematician, at his spare time for which he receives nothing else than the satisfaction to see it working!

The package used many ideas and codes from many people. We tried to mention every one where this is due. We apologize if any are missing. Please contact me and I will correct the situation.

Finally, I wish especially to thank the numerous users who sent messages. Their feed back is essential to guide the direction it may take in the future.

2.2. Shortcomings, Desiderata

قال أبو الطيب المتنبي :

ولم أر في عيوب الناس عيبـــــا كنقص القادرين على التمـــــــام

Here are some of the Basic things/work that still may need to be done. This wish list order will not necessarily be respected and mat change with time.

1. Complete the missing Farsi captions (The Arabic ones have been completed!)
2. Make the slanted forms of available fonts
3. Tune the headers and clean up more the language specific commands
4. Enable the Arabic script in mathematical formulae.
5. Work on a more elaborated and comprehensive guide for Arabi.
6. Think to a linguistic ligature for الله to get it automatically (if not too time and font space consuming) and add some more aesthetical ligatures from the fonts that have the appropriate glyphs!
7. Add `ISIRI 3342` Code as input encoding for Farsi? or some other standard more widely used.
8. Fine tune the Arabic typography!
9. Fine tune the vowelization and remove the "`*`" used to force the final form when vocalizing!
10. Adapt *fully* Arabi for use with an `ASCII` input encoding;???

2.3. Arabi License

بسم الله الرحمن الرحيم
﴿ قل لا أسألكم عليه أجرا إلا المودة في القربى ﴾

This package is *copyrighted* to its author. It is author-maintained and can be redistributed under the terms of the LaTeX Project Public License (LPPL). You can find it in CTAN at

`macros/latex/base/lppl.txt`.

This system 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.

2.4. Bugs and Error Reporting

For known error and bugs, see `bugs.txt`.

Please do not request updates from me directly, primary distribution will be through the CTAN archives. Nevertheless, to report any problems or bugs, or if you have any comment, please contact the author

YOUSSEF JABRI
يوسف الجابري

either by e-mail at `yjabri@ensa.univ-oujda.ac.ma` or at the address:

École nationale des sciences appliquées,
Box 696, 60000 Oujda,
Morocco

Notice that the author does not have Internet access at home **deliberately** and we do not consult our mail box every day, so if you don't get an answer quickly, please be patient.

Copies of materials that use Arabi are welcome for the author's collection.

APPENDIX C

Changes
مستجدات

3.1. Version 1.1

New features.

*) *Arabic in the bookmarks sidebar*. Now we can have Arabic in the bookmarks sidebar, when using `hyperref` (stored in the `.out` file created by `hyperref`). This is done by calling `puenc-ar.def` and passing the option unicode to `hyperref`, although `hyperref` is still not fully supported yet.

*) We support now the cmap package that [The `cmap` package is intended to make the PDF files generated by pdflatex *searchable and copyable* in Acrobat reader and other compliant `PDF` viewers.]
Usage: Just put the line `\usepackage{cmap}` at the beginning of your preamble, and it will pre-load the needed `CMap` translations for the fonts used in the document.

We provide a file `lae.cmap` and `lfe.cmap` so that now Arabi `PDF` files can use *search* and *copy-n-paste* with Arabic text.

*) In Farsi contexts, Digits (0, 1, . . . , 9) are written with the extended Indo-Arabic (Farsi) digits. This behavior can be changed to the usual one and back using the two control sequences `\farsimathdigits` and `\arabicmathdigits`.

*) Now, the Arabi works TEX4ht to convert LATEX document to `HTML`. It is still experimental but seems to work pretty well! See the documentation for more.

Changes to existing material.

*) Changed some control sequences to more standard forms and less problematic ones, (request made during TUG 2006)
`\R` → `\textRL`
`\L` → `\textLR` (`\L` is used to write the Polish character Ł)
`\embox` → `\LRmbox`
`\ambox` → `\ARmbox`
`\fmbox` → `\FRmbox`

*) Added also new commands:
`\textAR` (like `\AR`)
`\textFR` (like `\FR`)

*) Only the following Farsi fonts from the Farsiweb project are Free (GPL): (`Nazli`, `Nazli bold`, `Homa`, `Titr`). The others should be `GPL`'ed in their next release. Thank you R. Pournader. I modified the free fonts Notices to say they are `GPL`, as suggested by Pournader, and removed the non free ones from Arabi.

Corrections and improvements.

*) Completed the *Arabic captions*, *Farsi cations* are still *incomplete*
*) All filenames changed to *lowercase* and made the necessary modifications to Arabi files to reflect this change!
*) Corrected some typos reported by some users (Thank you Juan and Ja'far جعفر, United Kingdom). Now, the content of the users guide has been modified a little more.
*) Now the parentheses around equations numbers appear correctly. It works with standard classes article, report and book (with or without amsmath package) and with ams[*] classes. Nevertheless the user still should pass the `reqno` option to the class he/she is using! (reported by Mamoun Elkheir مامون الخير, Sudan)

3.2. Version 1.0

First released on July 2006.

Bibliography

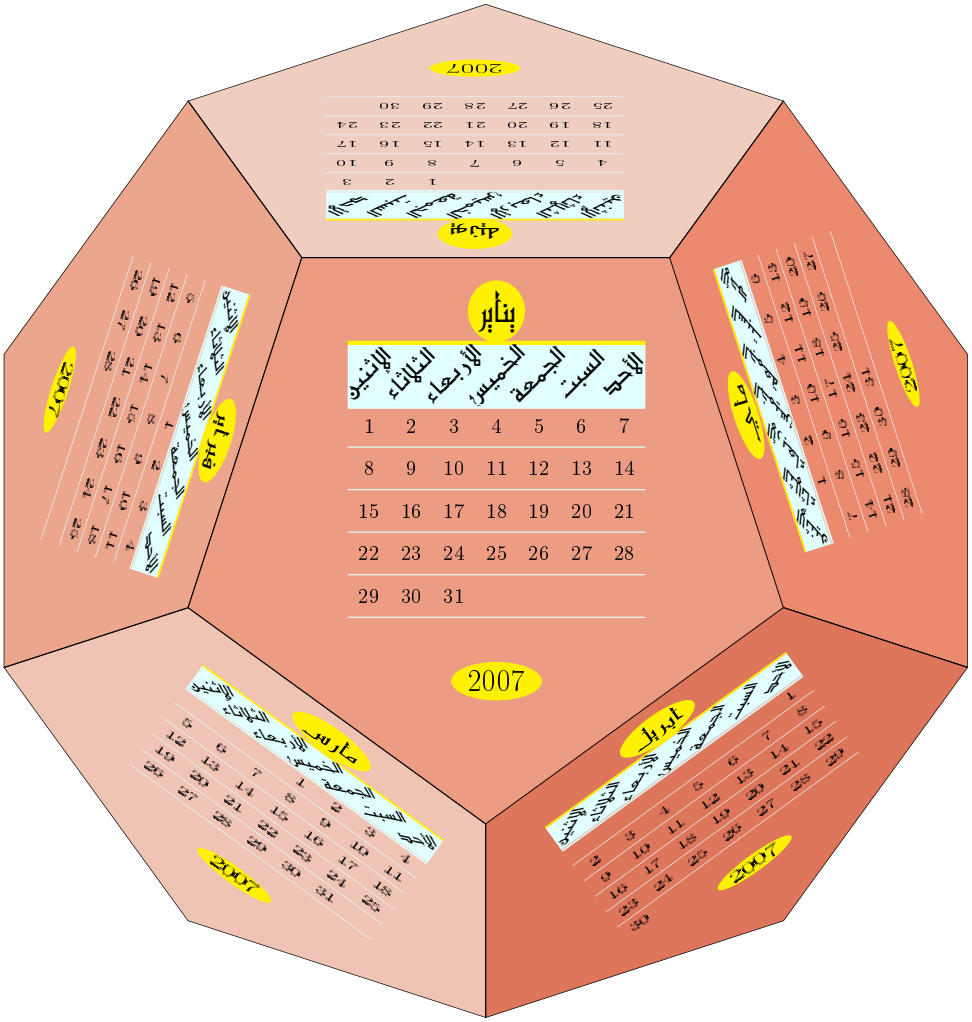
[1] B. Esfahbod and R. Pournader, FarsiTEX and the Iranian TEX Community, TUGboat, 22(3), 2003.
[2] J. Braams. *BABEL, a multilingual style-option system for user with LATEX's standard doc styles*. TUGboat, 12(2) 291–301, 1992.
[3] J. Braams. *An update on the BABEL system*. TUGboat, 14(1) 60–61, 1993.
[4] M. Goossens, F. Mittelbach, and A. Samarin. *The LATEX Companion*. Addison-Wesley, Reading, MA, USA, 1994.
[5] Y. Haralambous, Towards the revival of traditional Arabic typography… through TEX, Proceedings of the EuroTEX92 conference, Prague, 1992
[6] Y. Haralambous, *Typesetting the Holy Qur'ān with TEX*, Proceedings of the 2nd International Conference on Multilingual Computing (Latin and Arabic script), Durham, 1992.
[7] A. Hoenig. *TEX Unbound: Strategies for Fonts, Graphics, and More*. Oxford University Press, 1998.
[8] D.E. Knuth. The TEXbook. Addison-Wesley, Reading, MA, USA, 1986.
[9] D.E. Knuth. The METAFONT book. Addison-Wesley, Reading, MA, USA, 1986.
[10] D.E. Knuth. Virtual Fonts: *More Fun for Grand Wizards*. TUGboat, 11(1), 13–23, April 1990.
[11] Peter R. Wilson, *A Brief Guide to LATEX Tools for Web Publishing*, March 2000.
[12] D.E. Knuth and P. MacKay. *Mixing Right-to-Left texts with Left-to-Right texts*, TUGboat, 8(1), 14–25, (1987).
[13] A. Lakhdar-Ghazal, *Caractères arabes diacritiques selon l'ASV-CODAR (pour imprimer les langues arabes)*, Institut d'Études et de Recherches pour l'Arabisation, Rabat, 1993.
[14] K. Lagally, *ArabTEX— Typesetting Arabic with Vowels and Ligatures*, Proceedings of the EuroTEX92 conference, Prague, 1992
[15] K. Lagally, *ArabTEX Arabic and Hebrew. (Draft) User Manual Version 4.00*, March 11, 2004.
[16] L. Lamport. *LATEX: A Document Preparation System: User's Guide and Reference Manual*. Addison-Wesley, Reading, MA, USA, second edition, 1994.
[17] P. MacKay, *Typesetting problem scripts*, BYTE 11, **2** 1986, 201–218.
[18] The FarsiTEX Project at http://www.farsitex.org/
[19] M. Goosen and S. Rahtz with E. Gurari, R. Moore, and R. Sutor, The LATEX Web Companion, *Addison-Wesley*, 1999.
[20] Institute of Standards and Industrial Research of Iran at http://www.isiri.com
[21] Free download of the Arabic font pack from Microsoft (`arafonts.exe`). http://office.microsoft.com/arabicregion/Downloads/2000/arafonts.aspx
[22] Fabrice Popineau, Affichez vos documents LATEX sur le Web avec TEX4ht, Cahiers GUTenberg 37–38, December 2000, 5–43 (French, PDF).
[23] TEX4ht web site: http://www.cis.ohio-state.edu/~gurari/TeX4ht
The last version of the distribution can be downloaded from: http://www.cis.ohio-state.edu/~gurari/TeX4ht/bugfixes.html
[24] Till Tantau, The Ti*k*Z and PGF Packages. Manual for version 1.10, October 26, 2006 http://sourceforge.net/projects/pgf
[25] XETEX web site and mailing list http://scripts.sil.org/xetex
[26] The Unicode Standard at http://www.unicode.org/

Index

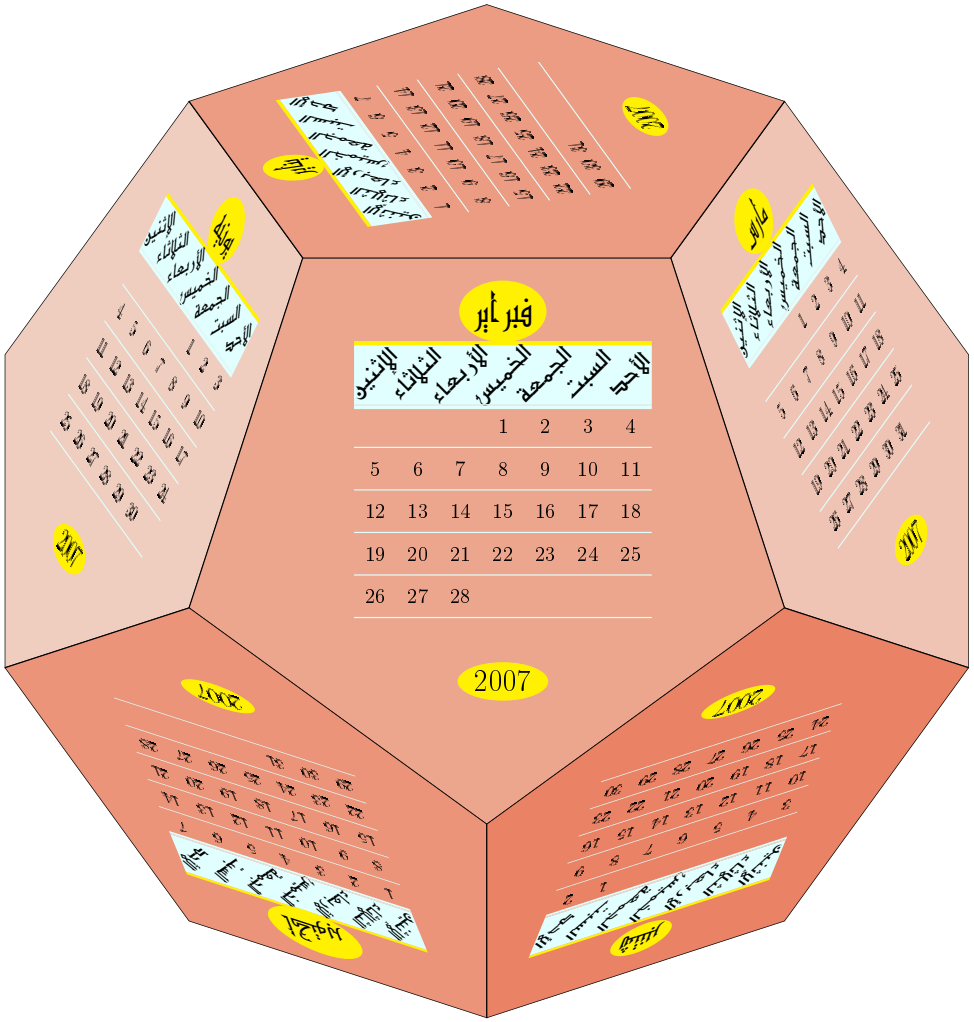

Wherever you go,
my best wishes
will trace your steps.

هدية العربي بمناسبة السنة الميلادية الجديدة. كل عام وأنتم بخير

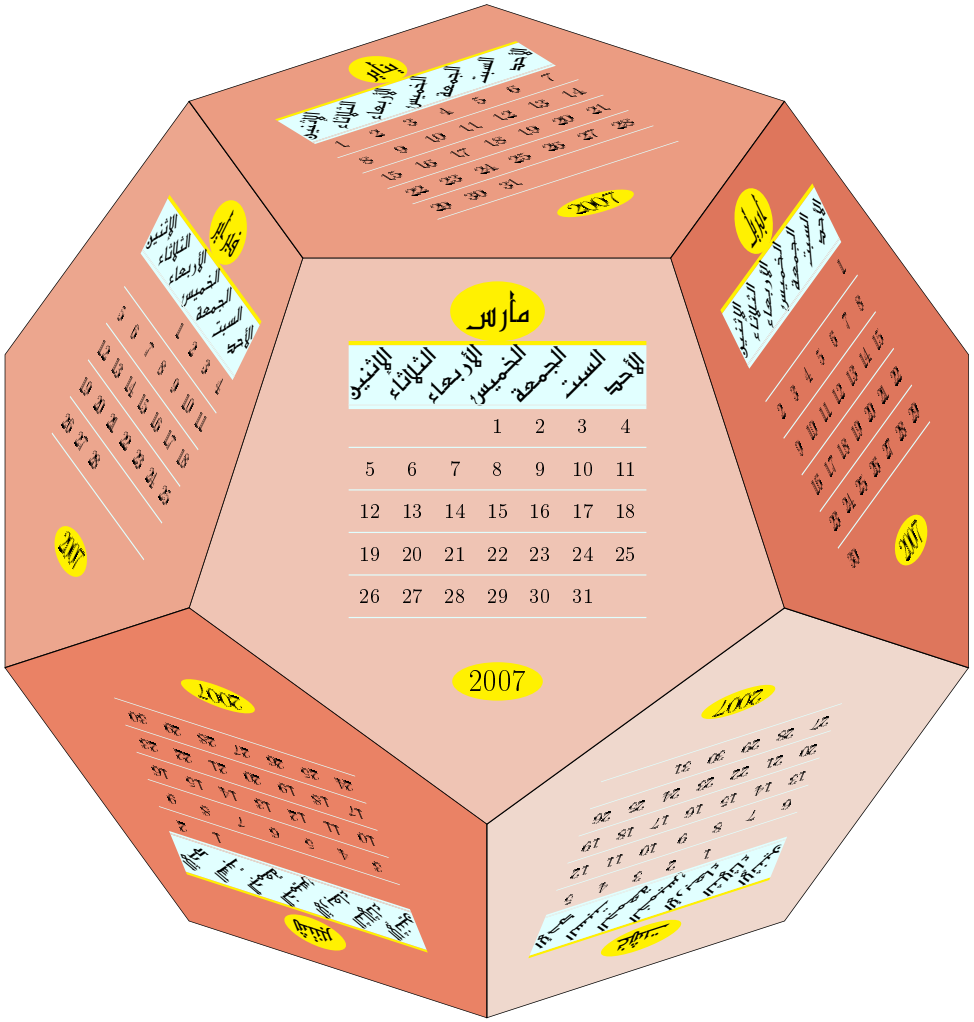
هدية نظام العربي



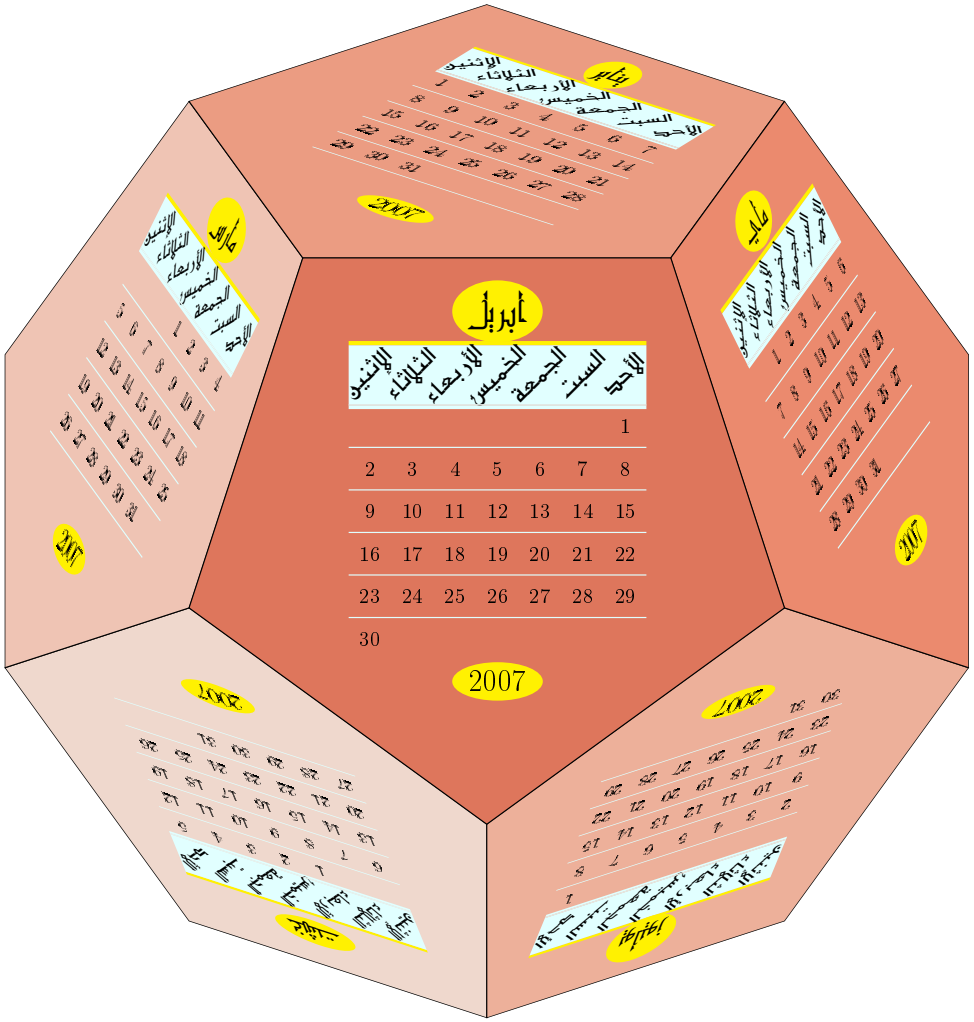
هدية نظام العربي



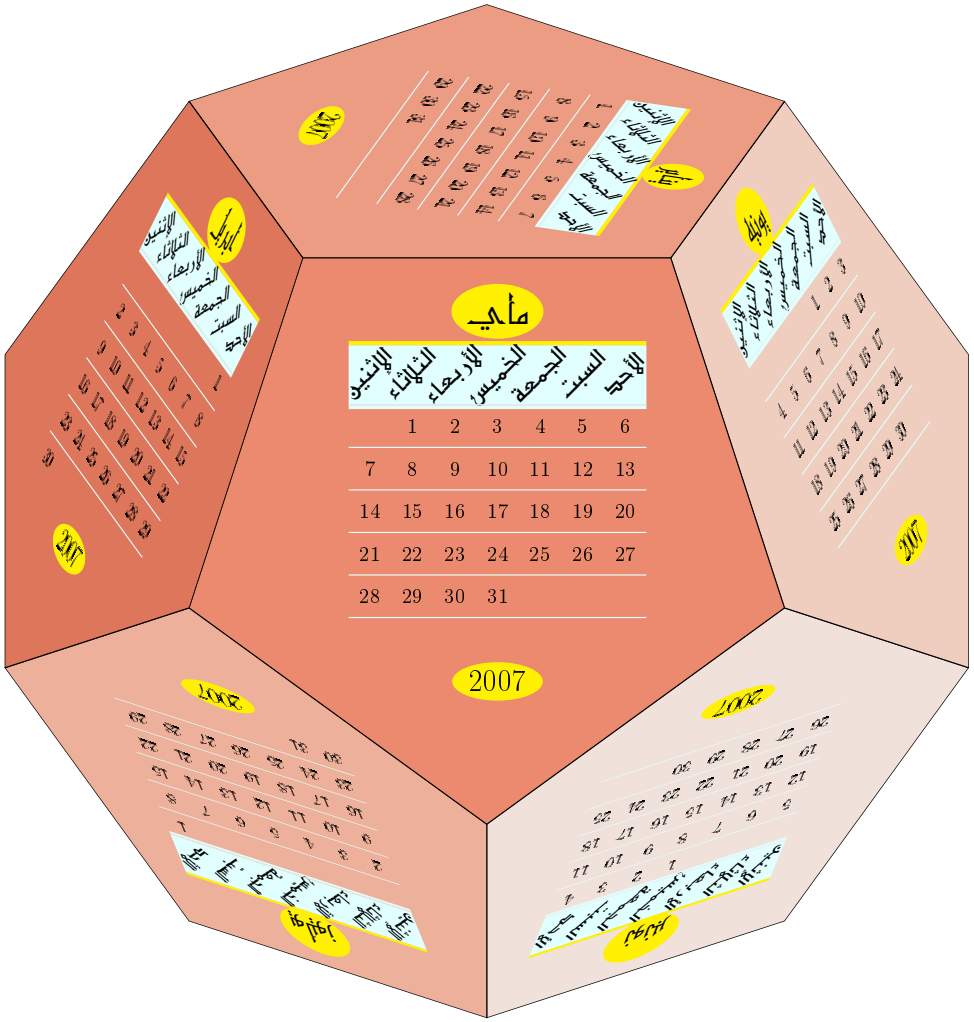
هدية نظام العربي



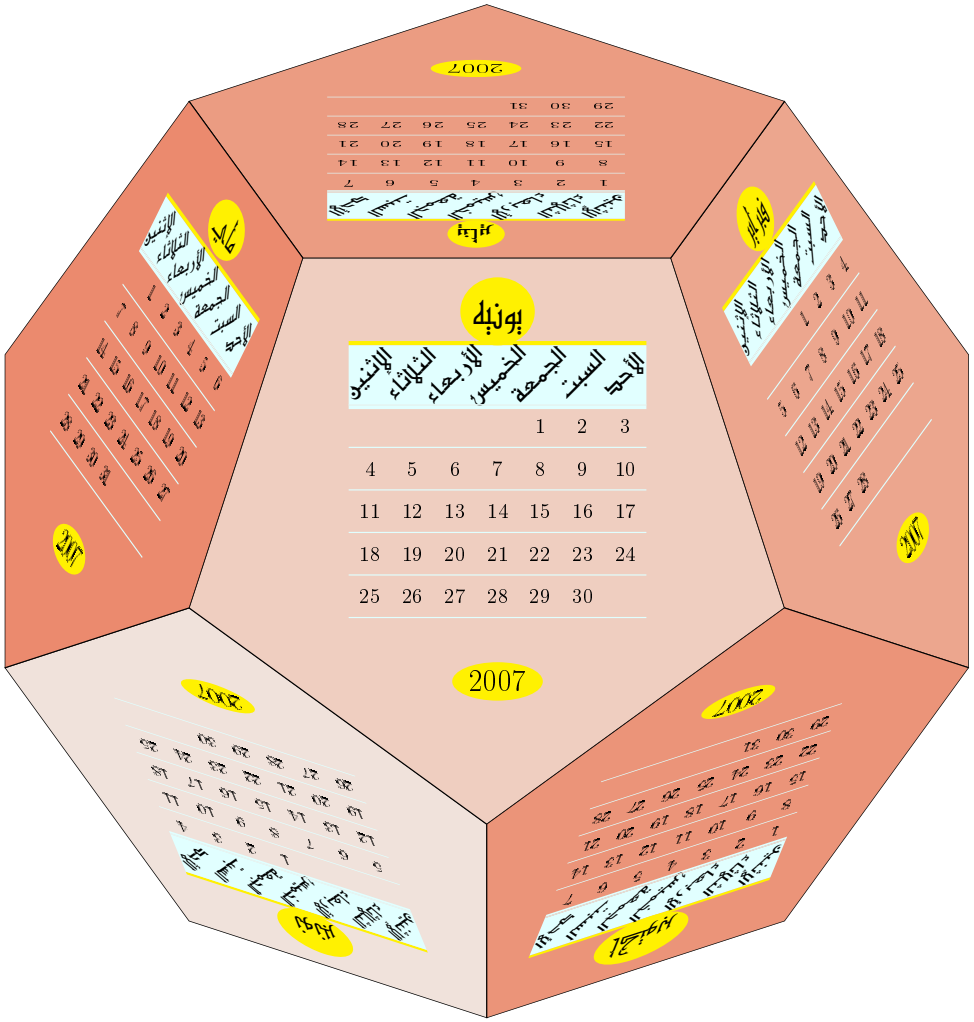
هدية نظام العربي



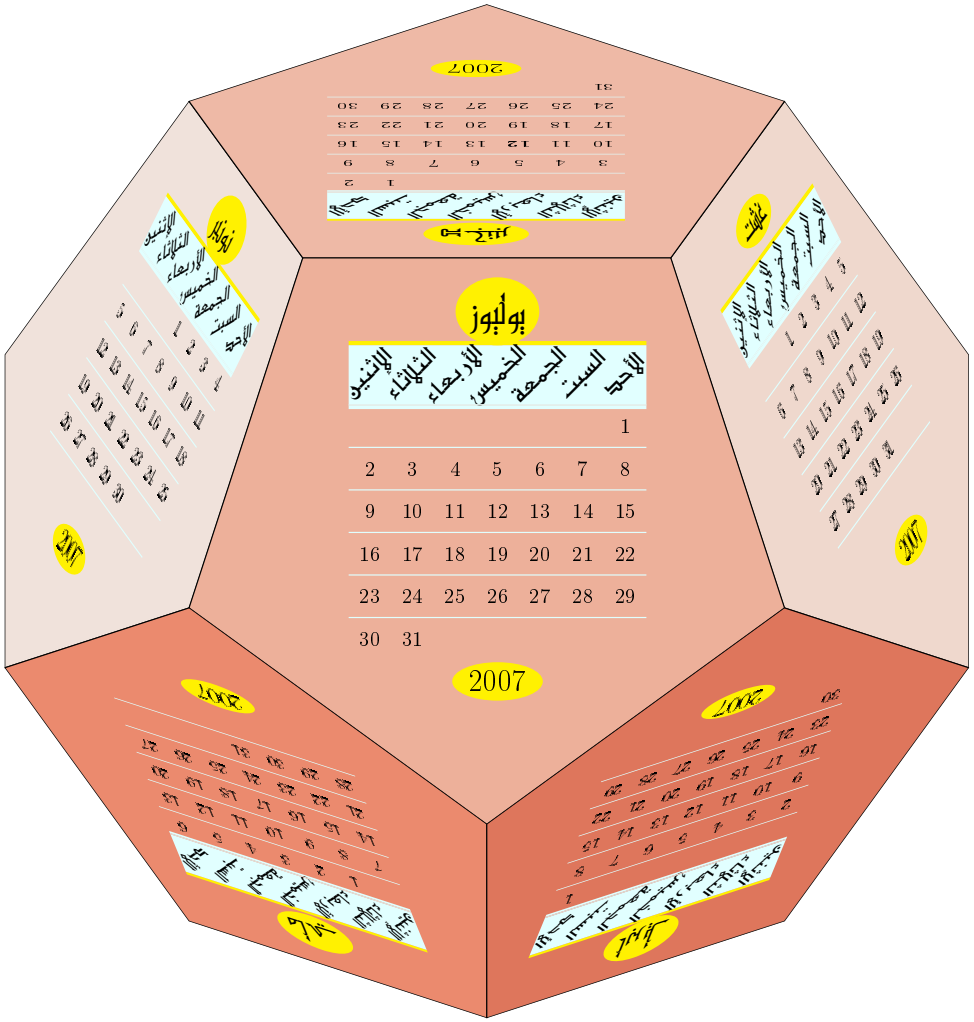
هدية نظام العربي



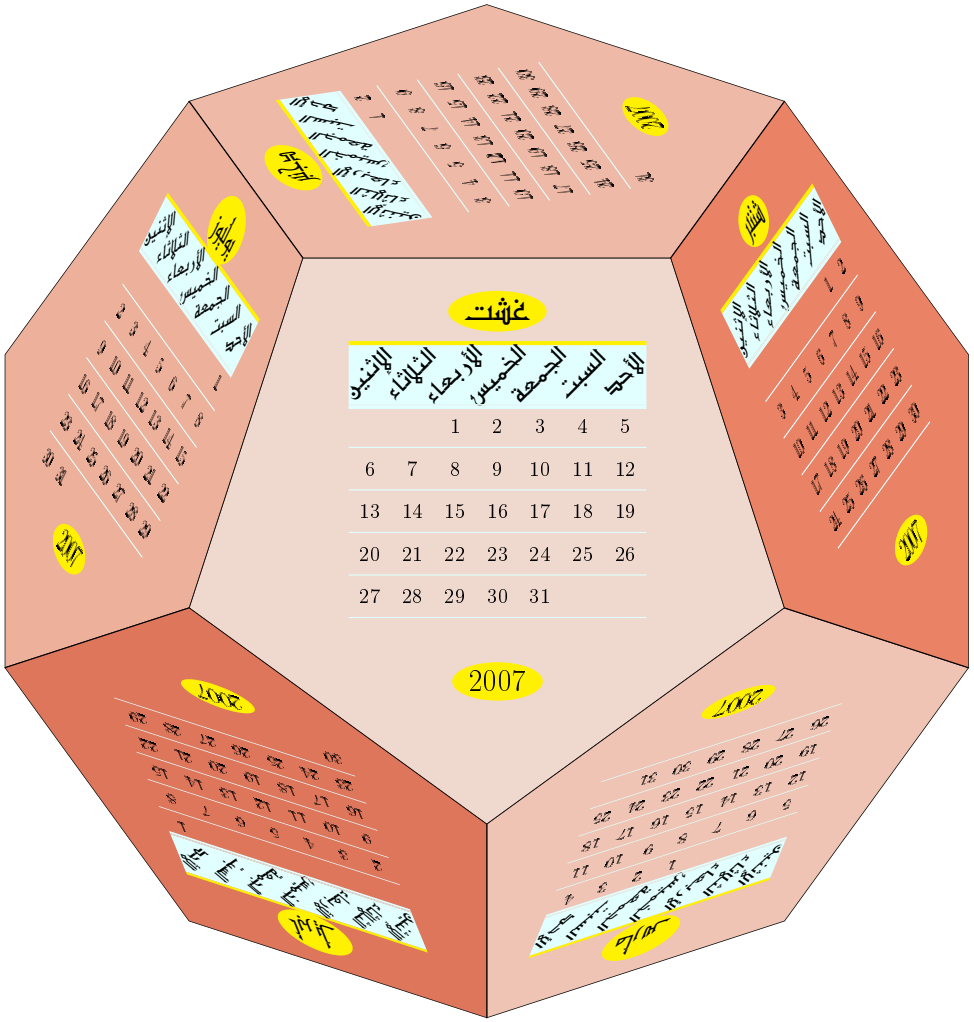
هدية نظام العربي



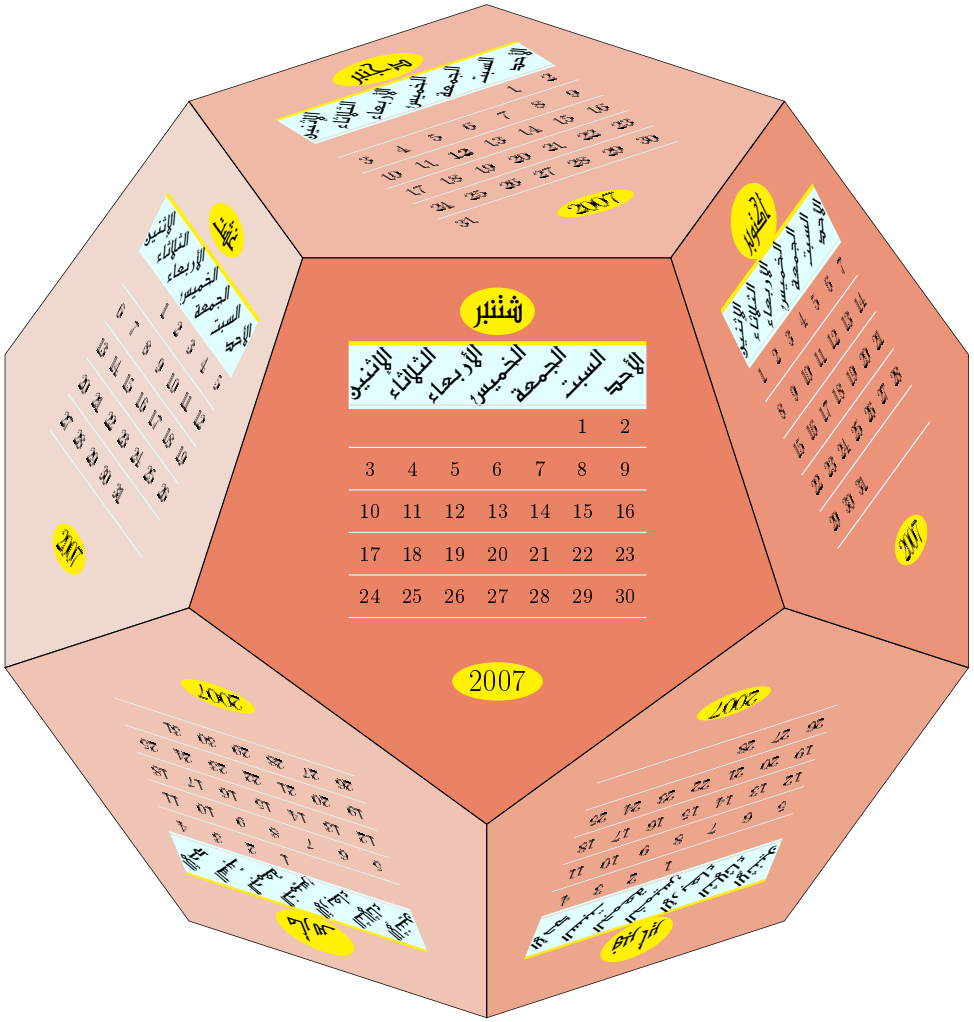
هدية نظام العربي



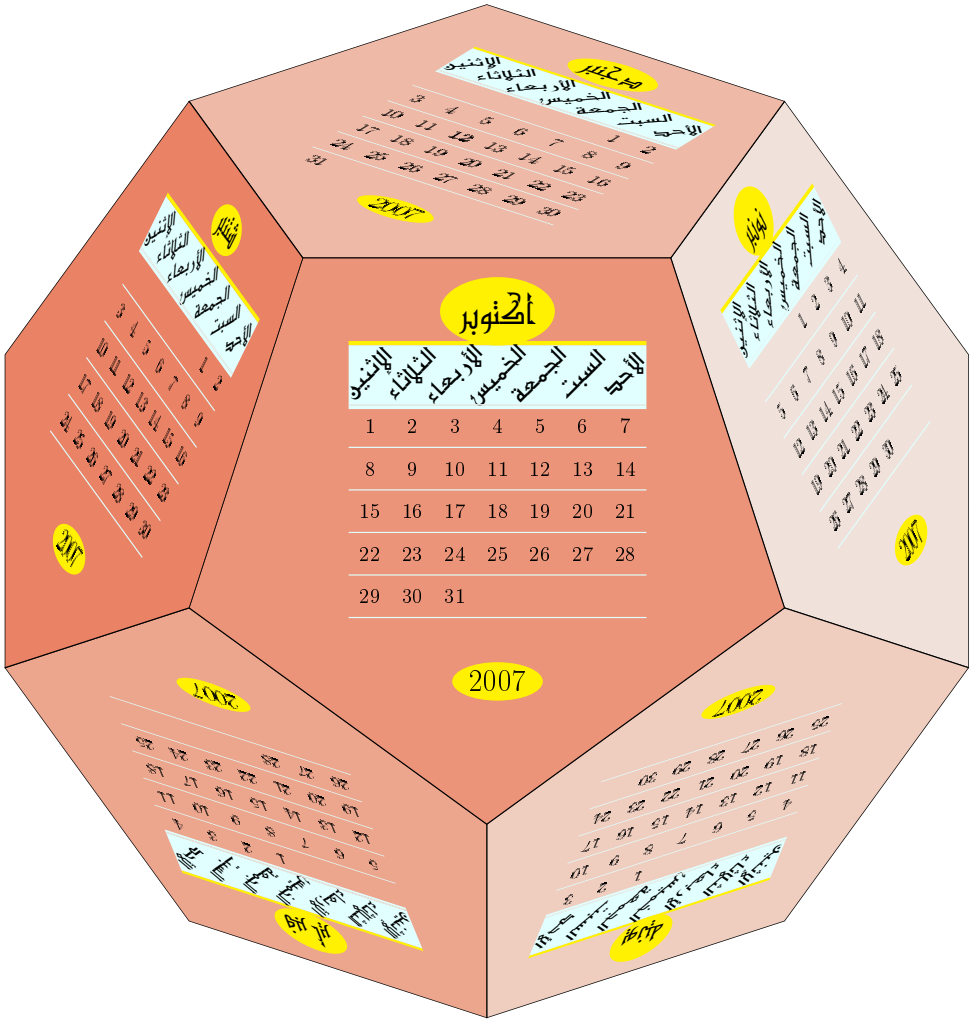
هدية نظام العربي



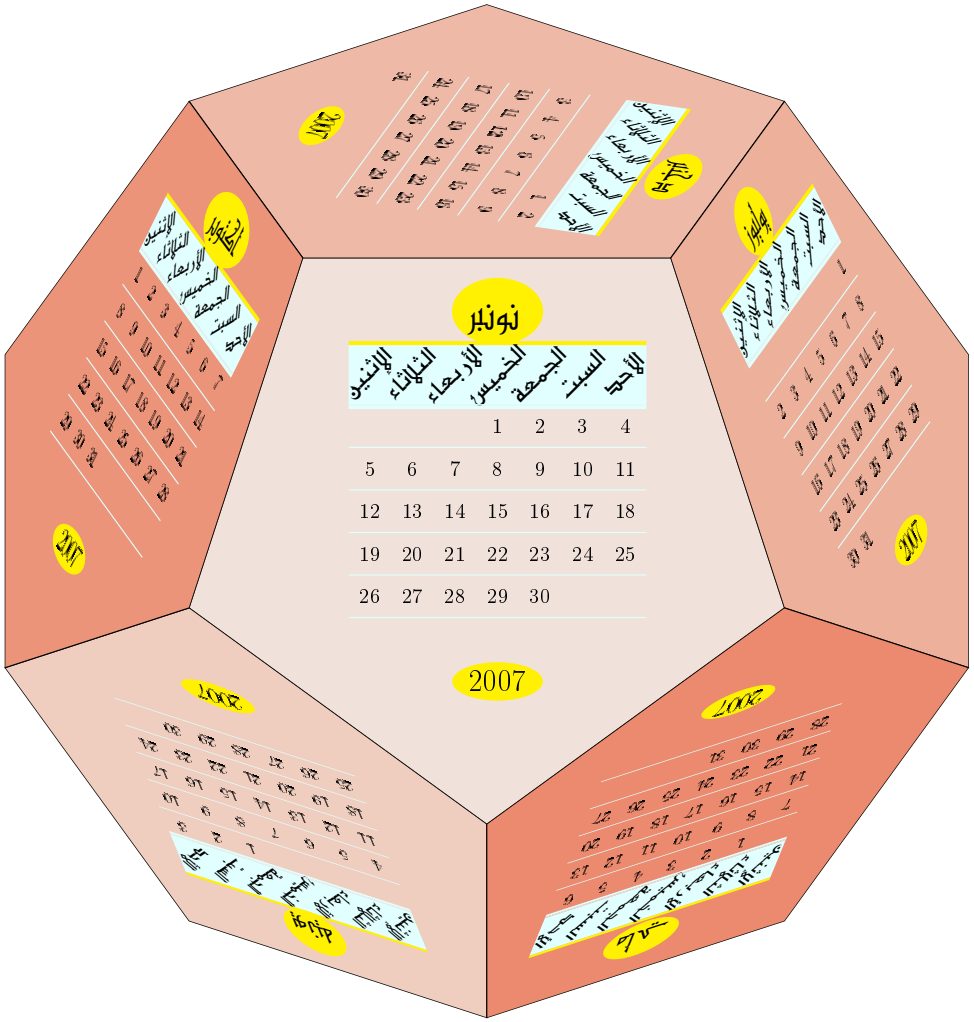
هدية نظام العربي



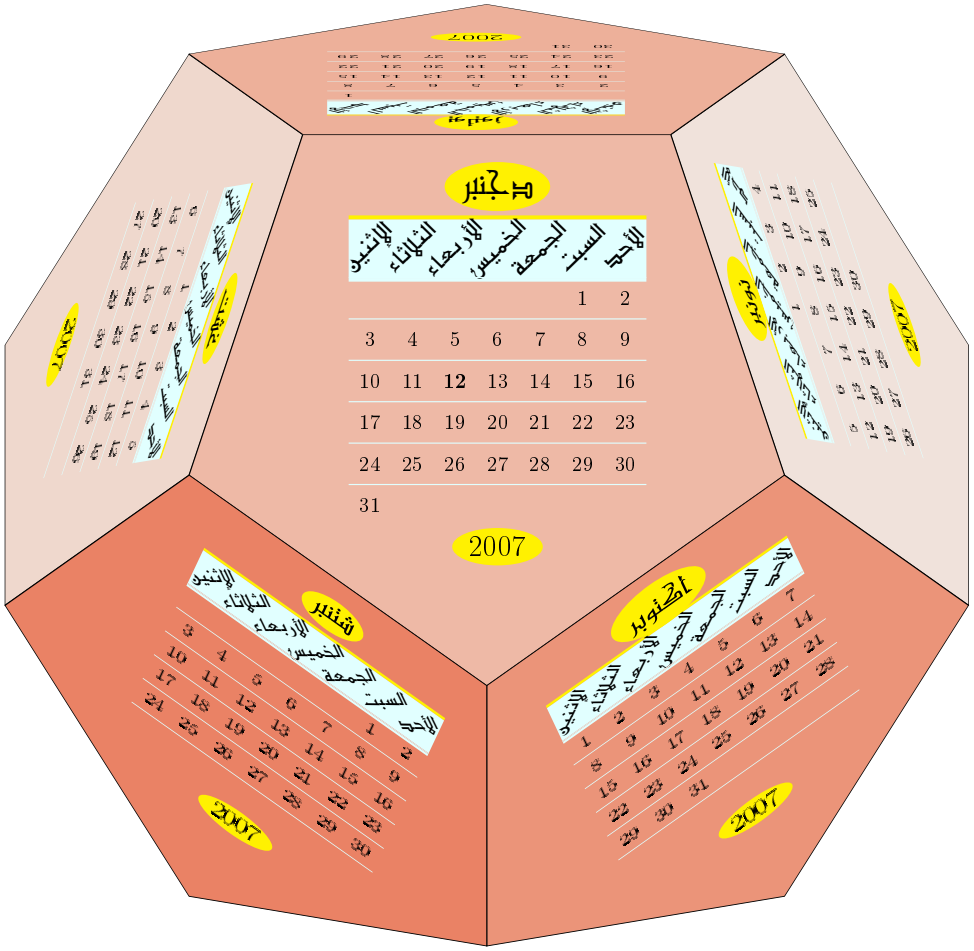
هدية نظام العربي



هدية نظام العربي



هدية نظام العربي



هدية نظام العربي