Persian Computing
with
Unicode

Behdad Esfahbod
unicode@behdad.org

The FarsiWeb Project
http://www.farsiweb.info/

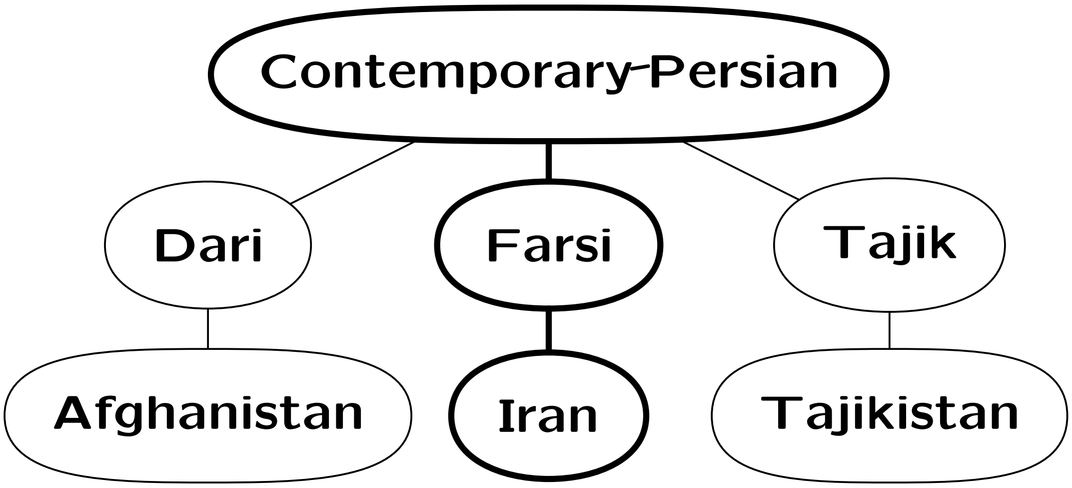
Abstract

In efforts to internationalize software, the Persian language has unfortunately merely been considered a variant of the Arabic language. That is to say, people have considered Persian to be Arabic plus a few extra letters. While this is partially true, it is only part of the story. Persian has a modified form of the Arabic script. In addition to extra letters, it has other modified letters as well as other stylistic complexities unknown to Arabic. While the script is an important similarity, one must consider that the semantics and habits behind the two languages are completely different. To successfully incorporate Persian support, a certain degree of knowledge relating to the semantics of the language is necessary. This paper focuses on providing developers and software engineers with no Persian-language background an in-depth understanding of the characteristics of Persian computing.

After briefly explaining the way Persian is handled in the Unicode standard, we will focus on other concepts that are critical for proper Persian support in internationalized software.

The inherent complexity of the language and lack of customers, and more importantly, customer feedback, have delayed Persian computing in software internationalization efforts. However, we have seen many issues addressed and improved in the recent years in regard to Persian computing. In keeping up with these improvements, this session is intended to give managers, software engineers, and developers enough information and references to add proper Persian support to their products.

What is Persian?



Persian, an Indo-European language was once spoken from the Middle East to India. Today it is spoken in Iran and several neighboring countries. It is important to distinguish between the spoken versus the written form of the language. The spoken forms of the variant dialects of Persian are generally mutually understandable across modern political borders. Being a member of the Indo-European family of languages, Persian is very similar to English, French and German. However, the script has been borrowed from the Arabic language. Arabic is, like Hebrew, a Semitic language and bears little resemblance to the Indo-European language family.

Today three major variations of the Persian language can be recognized as shown in the slide. These are *Dari* the official language of Afghanistan, *Farsi* the official language of Iran, and *Tajik* the official language of Tajikistan.

The name of the language is a source of much confusion. In English, the name of the language is "Persian" (ISO 639-1:2002). In the Persian language itself, the name of the language is "Farsi." Unfortunately, a small group of individuals outside of Iran have recently started using the word "Farsi" even when speaking English. This incorrect usage has created much chaos and difficulties for everyone. For example, one must search under both "Persian" and "Farsi" when looking for information about this language. It is absolutely necessary that professionals in the industry insist on use of the name "Persian" in all products and documentation. A further source of confusion is that the generic name Persian may either refer to the Persian language in general, or it may refer to the variant of Persian spoken only in the country of Iran. In the latter case, the Persian spoken outside of Iran maybe refered to by more local-sounding names. In Afghanistan, Persian is usually called "Dari" and in Tajikistan, Persian is called, "Tajik." In Iran, the default name, "Persian" is used or, as stated above, "Farsi" if one is speaking in the language itself. We have used the word "Farsi" in our diagram to distinguish it from "Dari" and "Tajik" but all three are in fact, Persian. The two-letter code for the Persian language as used in Iran and Afghanistan is **fa**. Compare this to "German" language with two-letter language code **de**.

The variations of the language should be identified in software by presenting the territory, e.g. "Persian (Iran)", instead of the *unofficial* name of the variation, e.g. "Farsi".

In the rest of this paper, the word "Persian" will be used to refer only to the official language of Iran.

Persian in Computers

There are three relevant national standards:

- ISIRI 3342:1992 Farsi 8-bit Coded Character Set for Information Interchange (deprecated)
- ISIRI 2901:1994 Keyboard Layout for Farsi: Characters in Computer
- ISIRI 6219:2002 Information Technology -- Persian Information Interchange and Display Mechanism, using Unicode

The infamous ISIRI 3342 character set standard is not the first of its kind. It is a sequel to the old ISIRI 2900 7-bit information interchange standard. But even ISIRI 3342 which tried to solve the problems of the old document, never opened the way for the development of software in Iran.

Until recently every vendor in Iran used its own 8-bit character set and its own keyboard layout. Then Microsoft implemented the Unicode standard in its Windows operating system and by the time Internet Explorer 5 was out, Microsoft Windows was the first system widely used in Iran that implemented Unicode-based Persian support. As a consequence people turned to the new system and finally ISIRI 6219 replaced the character set standard as the Unicode standard.

Fortunately ISIRI 3342 was never used widely so Persian is one of the lucky languages that do not have an 8-bit character set still in use. This is important to note that the ISIRI 3342 standard is deprecated now and is *not* the 8-bit standard of Persian text encoding. The author does not know of any application that supports ISIRI 3342 natively, and so does not know about *any* significant amount of text encoded in this encoding. So there is no advantage in supporting ISIRI 3342 in software anymore.

ISIRI 2901 is still the standard keyboard layout. It needs to be updated to go with the Unicode-based character set standard. The updated layout would be registered as another standard due to significant improvements. It will deprecate ISIRI 2901.

While like any other Persian speaker, I appreciate the efforts that brought about Persian support in Microsoft Windows, like any other software system, the Persian support in Microsoft products was not perfect. Their problems have spread all around the globe in the past five years and unfortunately have affected the common practice and user experience of some fine details of Persian computing to some degree. To identify these problems I will point them out in the following sections whenever something has been handled in a wrong way in Microsoft Windows system. Almost all of these problems have been already fixed or considered to be fixed by Microsoft.

Modern Persian Script

- Based on Arabic Script (⟨U+0600..U+06FF⟩ block): With some extra letters, some modified letters
- But with completely different semantics and typographical habits
- Is a bidirectional script: Is written from right to left, except for numbers
- Needs cursive joining: Two adjacent letters may be *joined*, forming 1, 2, or 4 glyphs for each character: (for example س, ـس, ـسـ, سـ)

The modern Persian script is an extension and modification of the Arabic script. Before the 7th century CE, Persian was written in a very different script known as the *Pahlavi* script. With the coming of Islam to Iran, the Arabic script was adopted for writing Persian. The use of the Arabic script then spread to other lands, each language adding letters and making modifications as needed. This propagated to Central, South, and even South East Asia, as well as North Africa; from Morocco to Java, where the alphabet was extended even more: from 29 basic Arabic letters to more than hundred letters in modern use (from Kurdish to Jawi).

Arabic is called a complex script. This is mainly because it is written from right to left. Of course the text may be mixed with Latin text and numbers that are written from left to right, adding to the complexity of rendering. The Unicode Standard Annex #9: *The Bidirectional Algorithm* provides an *exact* and *explicit* mechanism for converting a logically stored stream of characters including some characters of a right-to-left script, to a visually ordered one suitable for display. This algorithm is needed for Arabic (incl. Persian, Urdu, Sindhi, …), Hebrew (incl. Yiddish), Syriac, and Thanaa scripts.

The other complicated *feature* of the language is *cursive joining*. This simply means that the characters do not have a single shape and may *join* adjacent characters, forming up to four different shapes. The Unicode standard encoding only characters not glyphs, has allocated one character for each Arabic letter. This means a render-time process should select the proper shape for each character. This process is known as Arabic joining and is described in Section 8.2 of the Unicode standard: Arabic Script.

Arabic Script Rendering

Input text	Logical order	س ل ا م
After Bidirectional Algorithm	Visual order	م ا ل س
After Arabic Joining Algorithm	Glyph list	ﻢ ﺎ ﻠ ﺳ
After Ligation	Glyph list	ﻢ ﻼ ﺳ
When Rendered	Output	سلام

With enough care, it is possible, to apply the above algorithms in a different order, and get the same result.

The input text is said to be in *logical* order. This is the order that one reads and types in the text. After applying the bidirectional algorithm the order of characters is called *visual* order. This is the order that they should appear on screen. Then the Arabic joining algorithm determines which shape of a character should be rendered. After that some ligatures may form. And that is the final list of glyphs that would appear on screen.

There is an egg and chicken problem here. That is, to correctly apply the bidirectional algorithms, paragraphs should be broken into lines. But to break lines in almost all modern rendering engines, the final glyph widths should be known. And that is not known before applying Arabic joining algorithm and ligation! This simple argument adds to the complexity of the rendering mechanism, such that the two algorithms cannot be separated from each other and interact in some sense. This again adds to the complexity of the script, when it comes to computers.

There is another feature of the Unicode standard hidden in the Bidirectional algorithm. That is the *mirroring* property of some characters. For example the character U+0028 *Left Parenthesis* "("is defined to actually be **Opening Parenthesis**. This means that the same character is used in Arabic script too as an opening parenthesis, but because of the different direction of the script, it is *mirrored* and is rendered like this: ")".

Alphabet

- Extra letters:
  U+067E *Peh* (پ), U+0686 *Tcheh* (چ),
  U+0698 *Jeh* (ژ), U+06AF *Gaf* (گ)
- Modified letters:

Character	Isol	Fina	Medi	Init
U+064A *Arabic Letter Kaf* (Arabic Kaf)	ك	ـك	ـكـ	كـ
U+06CC *Arabic Letter Keheh* (Persian Kaf)	ک	ـک	ـکـ	کـ
U+064A *Arabic Letter Yeh* (Arabic Yeh)	ي	ـي	ـيـ	يـ
U+06CC *Arabic Letter Farsi Yeh* (Persian Yeh)	ی	ـی	ـیـ	یـ

The Persian alphabet shares with the Arabic alphabet most of its letters. There are four main extra letters that are neither written nor pronounced in traditional Arabic. Many rendering engines or fonts that write the code for their joining tables manually instead of extracting from Unicode data files, have problems joining these extra letters properly.

The Unicode standard has identified a different character for the Persian Kaf letter (Keheh is the Sindhi name). This is basically because of the different look of the final and isolated shapes of the character as can be seen above. Many font designers have ignored the difference in the past. As a consequence many people do not differentiate between the two shapes anymore and so use Arabic Kaf in Persian context. The *Courier New* font even mixes the appearance of the two, which adds to the confusion. Many Persian web sites have text encoded using Arabic Kaf.

The situation for Arabic Yeh is worse than that. It was impossible to show the Persian letter Yeh (called Farsi Yeh in the Unicode standard because of compatibility issues) with fonts shipped by early Microsoft Windows products. As you know, the inability to type one letter is as good as not being able to type at all!

Moreover, the Persian Yeh is also mapped incorrectly on the keyboard layout in Microsoft Windows products. To get around this problem, Persian webmasters (as well as those doing Persian word-processing) resorted to use of the Arabic Yeh to enable them to type Persian in one fashion or another. It was deemed preferable to see the two dots on the final Yeh to the completely wrongly shaped Yeh in medial position.

As a consequence, you see more than a half of Persian web pages use Arabic Yeh with two dots below instead of Persian Yeh. This is unfortunate to see people do not even complain about the two extra dots. This has been further complicated by *helpful* software vendors in Iran selling fonts with the dots removed from Arabic Yeh! Others mixed both Arabic and Persian Yeh to achieve a perfect visual presentation, while making it impossible to search the content using any search engine.

Alphabet (continued)

- Three shapes of composed Hamza Above:
  U+0623 *Alef with Hamza Above* (أ),
  U+0624 *Waw with Hamza Above* (ؤ),
  U+0626 *Yeh with Hamza Above* (ئ)
- Never used characters:
  U+0649 *Alef Maksura* (ى): Like Yeh, but no dots at all
  U+06C0 *Heh with Yeh Above* (ۀ): Should **never** be used instead of ⟨*Heh, ZWNJ, Farsi Yeh*⟩ or ⟨*Heh, Hamza Above*⟩ sequence

Hamza is the most ambiguous letter in the Persian alphabet. It is essentially one letter, but appears in three different shapes as can be seen in the slide. Sometimes people with different writing styles use one instead of the other. Sometimes they drop the Hamza sign and use a Alef (ا), Waw (و), or Yeh (ی) instead! We will come back to this with examples when discussing the loose searching problem.

There is another native Arabic letter that is not used in Persian: *Alef Maksura*. *Alef Maksura* is like a Persian or Arabic Yeh letter but with no dots at all. Add this to the confusion already discussed. By the way, these two letters, and the Arabic Kaf and Arabic Yeh letters are allowed to appear in Persian documents when quoting Arabic text.

Among the modifications to the Arabic script is the letter Heh with a small Hamza above. This sequence is unknown in Arabic and has proven to be a major challenge to implement in Persian. Unfortunately, this was first encoded as U+06C0 *Heh with Yeh Above*. Unfortunately, the Heh in this sequence was defined as a certain Arabic variant of the Heh which is not used in Persian. In appearance, this was not a problem, however, when search engines break down the sequence, they will not be able to process this sequence correctly for Persian. Therefore, the U+06C0 was deprecated in the Persian subset and now it is necessary to type the *Heh* and the U+0654 *Arabic Hamza Above* as two separate characters. The result is visually identical with the deprecated U+06C0. It should be mentioned that even in WinXP, the U+06C0 has been mapped on the Persian keyboard (called "Farsi" there) and so the user unknowingly propagages this error.

Special Characters

- U+0640 *Arabic Tatweel*, for a longer joining stem

  کتاب → کـــتاب

- U+200C *Zero Width Non-Joiner*, to prevent joining

  کتابها → کتاب‌ها

- U+200D *Zero Width Joiner*, to choose a joined glyph when it would not join naturally

  ه.ش → هـ.ش

- U+200E *Left-to-Right Mark*, U+200F *Right-to-Left Mark*, and other bidirectional control chars (⟨U+202A..U+202E⟩)

Tatweel is used when the author wants to force a longer joining stem as shown. As an isolated character, it look like a dash character, perhaps thicker (ـ).

ZWNJ is one of the essential features for proper Persian support. It is widely used in Persian texts. Think of it as the hyphen character in phrases like “home-brew”, or “pseudo-random”. One may drop the hyphen in the first example and write it “homebrew”. Or one may replace it with a space in second example and write it “pseudo random”. In Persian, when to use ZWNJ and when to use a space are governed by complex rules of style and esthetics.

ZWJ is not used as regularly as ZWNJ. It forces a character into its joined form when it normally would be in its isolated form. This is usually used in for two purposes:

- To distinguish between U+0665 *Arabic-Indic Digit Five* (٥) and U+0647 *Arabic Letter Heh* (ه‍). As shown in the slide.

- To present a specific shape of a character. For example to draw this initial Yeh glyph: يـ. These shapes are sometimes used in abbreviations.

Bidirectional control characters are used to control the behavior of the Bidirectional Algorithm explicitly. *LRM* and *RLM* are specially needed in, for example, the paragraph direction of a paragraph starting with Latin text to right-to-left and vice versa.

Numbers

- ⟨U+06F0..U+06F9⟩ *Extended Arabic-Indic Digits*:

  ۰ ۱ ۲ ۳ ۴ ۵ ۶ ۷ ۸ ۹

  instead of ⟨U+0660..U+0669⟩ *Arabic-Indic Digits*:

  ٠ ١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩

- U+066C *Arabic Thousands Separator* and U+066B *Arabic Decimal Separator*:

  ۹٬۸۷۶٬۵۴۳٫۲۱۰

- Western numerals in Latin context. Persian numerals everywhere else (page numbers, section numbers, …)

Three of ten decimal digits used in Persian look different from their Arabic counterpart. For this and some other technical reasons the Unicode standard has allocated a set of ten characters for Persian digits (called Extended Arabic-Indic Digits), as well as the ones used in Arabic (called Arabic-Indic Digits). Neither of these two set should be confused with the Western set of ASCII digits also known as Arabic numerals.

Persian digits should be used with Arabic Thousands Separator and Arabic Decimal Separator. Using comma and a dot-shaped decimal separator is not allowed with Persian digits. In Iran people always read and write Persian digits. This means that page numbers, section numbers, monetary values, font sizes, spreadsheet cells, are all supposed to be in Persian digits. This level of support needs the host system to be able to parse Persian digits as numerical data.

In Iran people read and write western digits in a Latin context. This is unlike most of Arabic countries where they write numbers with Arabic-Indic digits even in the middle of an English text. So turning all western digits into Persian digits automatically is not an option. Microsoft software does not interpret Persian digits characters as numerical data yet. As a consequence they have put western digits on their Persian keyboard (called “Farsi” in their context). And finally you see Persian sites with western digits typed in everywhere.

Arabic-Indic digits should not be used in Persian text.

Other Characters

- Harakat (Vowel) Non-spacing Marks: ـٰـّـٌـٍـًـُـِـَ …
- Arabic Punctuation Marks:
  U+060C *Arabic Comma* (،),
  U+061B *Arabic Semicolon* (؛),
  U+061F *Arabic Question Mark* (؟),
  U+066A *Arabic Percent Sign* (٪),
- ⟨U+00AB, U+00BB⟩ *Double Angle Quotation Marks* (( ))
- Shared Punctuation Marks:
  Latin full stop, exclamation mark, parenthesis, square brackets, …

Almost all Arabic Harakat (vowels) are allowed in Persian text. But they are usually used rarely. There is an exception for U+0650 *Arabic Kasra* (ـِ) that is widely used at the end of words.

Arabic punctuation marks are used instead of their Latin counterparts, because of the difference in shape. Latin quotation marks are not allowed in Persian text and Double Angle Quotation Marks should be used instead. In Persian fonts these Double Angle glyphs usually have a rounded shape. Other Latin punctuation marks are allowed.

The ISIRI 6219 standard contains the complete list of characters that should be supported, are optional, or are forbidden in Persian text.

Keyboard Layout

ISIRI 2901:1994, features:

- All letters and punctuation marks
- Reasonable placement
- Persian digits
- Regular and shifted keys only
- Some empty slots

The ISIRI 2901:1994 national standard is an update to the ISIRI 2901:1989 old keyboard layout. This is important to support the second edition, not the old one. Many companies in Iran provide the old style and claim to be conforming to ISIRI 2901 standard.

This standard has the best design among different layouts used by different companies in Iran. Unfortunately it is not supported by Microsoft yet. But free drivers are available in different formats and for different environments. Many sites are changing their keyboard layout to this standard and many Persian sites provide this keyboard through a JavaScript code. They say that once you've experienced the feel of this layout, you'll never leave it! Not that it is designed for fast typing or being ergonomic; Just that it *is* better than the other alternatives.

The mapping to Unicode characters is available at `http://www.farsiweb.info/table/2901-unicode.txt` and the layout itself can be seen at `http://crl.nmsu.edu/~mleisher/keyboards/persian.html`

Keyboard Layout (continued)

Proposed update, features:

- Fully backward-compatible with ISIRI 2901:1994
- Unicode 4.0 repertoire, and complete support for ISIRI 6219:2002
- Support quoting Arabic text
- Uses AltGr to add required but rarely used characters
- Adds all ASCII punctuation marks, useful for editing XML, …
- Adds bidirectional control characters

The layout has already been out as an experimental driver for Microsoft Windows for a few months now, and is passing its final stages to become the new national standard. It features a wide range of characters that used in Iran.

For many years people have suffered from the problem of not having many ASCII marks on the keyboard, like double quotation, or number sign, that are needed in editing markup languages, like HTML and XML. While they would need to switch to the English layout to type ASCII letters and digits, everything other ASCII character can be entered in this layout at most by holding AltGr.

A preview of this layout is available at:
`http://www.farsiweb.info/standard/ir-kb-layout-preview.pdf`
And the windows driver at:
`http://prdownloads.sourceforge.net/farsitools/persiankeyboard.zip?download`

Fonts

- Microsoft fonts are Arabic
- Tahoma is the best looking one
- Persian fonts are not Unicode compatible yet
- The only ligature: Lam-Alef
- Nastaliq is desired, but not possible yet

There are almost no Unicode compatible fonts suitable for Persian available to the public. The best option are the Microsoft core fonts which ship with Windows. But they were designed for Arabic and do not have a Persian look to them. The best among them is Tahoma, which is widely used in Persian web sites today, but it looks comic.

Even though completely functional, many Persian users simply refuse to use them, purely for esthetic reasons and instead prefer the completely incomplete and non-standard Persian hack fonts widely available for download on the internet.

These popular Persian fonts are not available in Unicode compatible form yet, but some beta releases are available for download. The whole package should be out in a couple of months. The project is supported by High Council of Informatics and fonts will be available for free. Thus, the font problem should be solved within the next few years although it may take longer for the non-compatible fonts to completely disappear.

One important note in Persian font design is that unlike in Arabic countries, in Iran people usually do not like any ligatures. The only ligature which is used and is mandatory by the Unicode standard, is the Lam-Alef ligature. It would be nice if Arabic fonts had the ability to turn off other ligatures for Persian text through OpenType's LanguageSystem tables.

Historically, Persian typography has been using Nastaliq style since its invention in 15th century CE. Nastaliq is an artistic style of writing Persian, with complicated joining and curves. With lead typography it switched back to Naskh, which is what used today. With late 1990s' digital typography tools, Nastaliq became public again, but the popularity dropped because of unreadability. It is still used in rare occasions. Persian Nastaliq is completely different from Pakistani Nastaliq. There are rumors that a Persian Nastaliq OpenType font is possible, but the support for needed OpenType features is not implemented in any system yet.

Date and Time

- Three calendars in use!
  Gregorian, to synchronize with the rest of the world
  Jalali, the official calendar
  Islamic, for some holidays and ceremonies
- Islamic calendar depends on moon-sighting once a year
- Week starts Saturday
- Business weekdays from Saturday to Thursday
- 24-hour preferred in media
- No AM/PM equivalent

You usually read three of them on a typical Iranian wall-mounted or pocket calendar. Jalali is the most commonly used system and is the default for everyday use; but the two others should be around too. So it is important that software systems support these hybrid combination. A list of Iranian national holidays is available at `http://www.farsiweb.info/table/iran-holidays.txt`

While Jalali is a solar calendar, it is not synchronized with Gregorian system. For example, my date of birth is 27 September 1982, but it does not mean that my birth day is 27th of September each year. For example, my next birth day is September 26th 2004, according to Jalali system. The reason is that 2004 is a leap year. Jalali has its own leap years. This simply means that periodic events in Jalali system cannot be translated in Gregorian system for storage. A sample Gregorian to Jalali two-way converter is available at `bamdad.org/date`. Sources for the converter are also available for free.

With the Islamic calendar, the situation is even worse. A once-yearly moon-sighting is used ot determine the length of one month, but any changes are reversed in the following month, such that the global offset of the system can be pre-computed.

Week days start on Saturday and end with Thursday. The *weekend* consists only of Friday. There is nothing simpler than supporting this in an application written from scratch. On the other hand, there is no design deficiency bigger than only supporting Sunday and Monday as week start days.

Finally, 24-hour format is what used officially. Moreover, there is no direct equivalent for AM/PM. In Iran people use different words after the hour to indicate they mean AM or PM. These words are equivalents to mid-night, early morning, morning, noon, afternoon, evening, night.

Collation

- Like Arabic basically
- ن < هـ < و < ي → ن < و < هـ < ی
- Some L2 equal pairs:

  ک $<_3$ ك   ی $<_3$ ي   ة $<_3$ ت

  ۴ $<_3$ ٤   ۵ $<_3$ ٥   ۶ $<_3$ ٦
- Traditional rules: Hamza variants are L2 equal:

  أ $<_3$ ؤ $<_3$ ئ
- Modern rules: L2 equal with their base letter:

  ا $<_3$ أ   و $<_3$ ؤ   ی $<_3$ ئ

Persian collation rules are basically the same as Arabic. There is a major difference in the order of two main letters Heh and Waw. The modified letter are considered basically equal and only make a difference in third level in Unicode collation algorithm. The preference is trivially for Persian letters to come first in case of a tie.

The confusing part is about Hamza. Traditionally Hamza has been considered a single letter. So three different variants would be sorted as one letter and before Alef. But recently this habit is changed in favor of the base characters that Hamza sits on. There are both good and bad examples of why each way is good and is bad. We will soon see examples that different variants of Hamza are used for the same word, and will see examples that the Hamza can be replaced by the base characters. But generally speaking, the modern rules make more sense, but each one has its own users. So providing both is desired.

There is a national project under way to identify Persian collation rules completely and precisely.

Loose Searching

ZWNJ ≃ Space

حجة الاسلام ≃ حجت الاسلام

کـتاب ≃ کتا ب

ي ≃ ی

ZWNJ ≃ empty string

دایره المعارف ≃ دایرة المعارف

خانهٔ ≃ خانه ی

ك ≃ ک

سؤال ≃ سوال　　پائیز ≃ پاییز　　تأخیر ≃ تاخیر

مسئول ≃ مسؤول　　مؤمن ≄ مأمن　　مسأله ≃ مسئله

ملک ≃ مِلک　　مِلک ≄ مُلک　　مُلک ≃ ملک

۰ ≃ ٠ ...　　۴ ≃ ٤　　۵ ≃ ٥　　۶ ≃ ٦　　۹ ≃ ٩ ...

Most of the world's "Find" dialogs has an option to turn case sensitivity on or off. Also almost all search engines have their own rules to find matches that queries with accents removed. The equivalent for these for Persian is a set of simple rules, that when applied makes a huge difference in the quality of the matched results.

Many of these rules that are shown above are due to different orthographical practices. Unfortunately there is not a general framework for loose searching implemented in any software as far as I know. But many applications already implement some of these rules when searching in Arabic script in general. It is a real problem of the Iranian computer users today that when searching for a word in Google they have to examine different cases, with Arabic Yeh, or Persian Yeh, with Waw with Hamza Above, or with Waw only, etc.

When applying these rules in a computer system, especially a word processor, it is important to let the user turn each instance of loose matching on or off. For example while generally users like the the Arabic Yeh to be matched to Persian Yeh and vice versa; then there are this rare but important situations where the user may like to find just Arabic Yehs in the document to change them to Persian Yeh! Then the user should be able to turn loose matching of Arabic Yeh to Persian Yeh off. Same for other cases.

Much like the case for collation, this is not a complete and exact list of all cases that should be considered, but the most important one. The same national project is working to identify and document the loose searching requirements of Persian computing precisely.

Last Notes to Application Developers

- Typesetting Persian paragraphs:
  Justified lines
  No inter-letter spacing
  No word hyphenation
  Almost no inter-word spacing
  Use Tatweel instead: کتاب من → کـــتـــاب مـــن
- All text fields Right-to-Left
- Persian numbers
- Right-to-Left layout
- Beware: Right and Left are swapped!

Persian paragraphs usually have justified lines, but right-justified paragraphs are allowed too. Justified lines can be achieved in CSS by `p text-align: justify;`. Moreover, renderers should avoid inter-letter spacing and word hyphenation with Persian text. Inter-word spacing should be avoided as much as possible too. Instead, the joining stem can be stretched to adjust the text to the line width. This is like inserting Tatweel characters in special places that characters join together.

Another thing to note is that all text fields should be right-to-left. In CSS it means `direction: rtl` for almost all elements. This is specially needed for fields like date that have no Persian letter in them.

And do not forget that numbers should be presented in Persian. This usually means that you should have special number output handling functions that when working in Persian mode, generate Persian digits. No programming environment does handle generating Persian digits automatically in general.

Last but not least is to remember to mirror the general layout of the user interface. In CSS this is almost automatically done when you set direction for HTML tags. But this is not the whole story. All the places that you specify `left` or `right` in your code, the value most probably should be switched. By the way, it would be easy if that were all, but there are places that the switch should not happen, for example a text entry field for *English Name* better always be aligned to left, not right.

This is very important, when writing code, that sometimes the left and right directions are swapped. For example it is quite common that in Persian application with automatic right-to-left layout, in a menu bar, you press the left arrow key, but it would bring you to the right neighbor!

Current Status – Microsoft Windows System

- Renders correctly
- Shipped fonts work
- Keyboard layout is terrible
- No Persian digits support
- No Iranian calendar
- Locale data is wrong in places
- No interface translation
- Not trivial to enable Persian support

As mentioned before, Microsoft Windows systems are the widely used system in Iran. The current status as of Microsoft Windows XP is that the shipped fonts finally work as expected with Persian data, but the keyboard layout is next to unusable and wrong. Standard keyboard driver and Persian fonts are available to download for free. There are chances that they find their way in the next major version of the system, but there may be a problem with font licenses.

There is not much more about Persian support in yet. Persian digits does not work, and so they do not appear on the keyboard layout. There is no Iranian calendar support, and the other localization work is still weak. As an example, AM/PM schema has been translated to Persian, which is never used by Iranians.

Microsoft has never had a Persian translation of the system interface, so people usually use Windows in English. This, plus the compexity of enabling Persian support in Microsoft Windows has proved to be a stopper for end-users. There are a handful of different software vendors in Iran selling software to *enable* or *add* Persian support to Microsoft Windows XP!

Fortunately Microsoft has started the work on Persian interface translation. So we can expect that as of the next major release, Persian is in the initial list of supported languages, and selecting Persian would automatically enable all other needed options. This can put an end to a complete business in Iran!

Needless to say, Microsoft Office inherits the same level of support and so the same shortcomings. There is an excellent tutorial by Connie Bobroff on setting up Microsoft Word to produce Persian documents. It covers in detail Windows versions from 98 to XP. The tutorial is available online at:
`http://students.washington.edu/irina/persianword/persianwp.htm`

Current Status – Linux

- Important systems support rendering
- No good fonts yet
- Standard keyboard layout
- No Persian digits support yet
- KDE claims Iranian calendar support
- Some interface translation done
- Not trivial to enable Persian support

It is harder to talk about Persian support under Linux as there is no reference Linux distribution. Another reason is that not all applications use the same library and toolbox. But the good news is that almost all important platforms have basic Persian support already. This includes GNOME, KDE, Mozilla, OpenOffice, and a few others. Needless to say, GNOME uses Pango to display internationalized text, including Persian.

There is no good Persian font available under Linux yet, but the fonts we mentioned earlier, that are passing their beta stage rapidly would become available and can be included in any Linux distribution.

Standard keyboard layout is available, both under the X Window system and the Linux console. Interestingly, KDE claims to have Iranian calendar support, but we have never tested it. The locale data seems to be more accurate under Linux. Moreover, there is work that would enable Persian digits in the GNOME platform in the near future.

There is some Persian interface translation work available, but still far from a good quality usable thing. The good point is that the mechanisms are open and mature. So anyone is welcome to translate any application he would like.

Much like the case with Microsoft Windows, Linux users suffer from the nontriviality of enabling Persian support. In the case of Linux it mostly means to just setup the keyboard and fonts, but since Linux is much less popular in Iran, not everyone knows how to do this.

Current Status – MacOS

- Supports rendering
- No good fonts
- Legacy and standard keyboard layouts

Apple hardware and MacOS systems are rare in Iran. The latest system MacOS 10.3 is known to render Persian pretty good. But again the shipped fonts are not Persian. MacOS provides both the standard keyboard layout and their own legacy layout. Apple used to ship Persian translated interface in the old pre-Unicode days. But seems like they have stopped it long ago. The good news is that translation infrastructure is in place so again anyone can do the translation.

The author does not know about the level of Persian support under MacOS systems in more detail.

References and Resources

- The Unicode Standard at http://www.unicode.org/
- Institute of Standards and Industrial Research of Iran at http://www.isiri.com (documents in Persian)
- The FarsiWeb Project at http://www.farsiweb.info/
- PersianComputing list at http://lists.sharif.edu/mailman/listinfo/persiancomputing
- Typing Persian Word Documents with Windows Tutorial at http://students.washington.edu/irina/persianword/persianwp.htm

The Unicode standard is considered a great resource for Persian computing. After that you may like to look at the FarsiWeb Project web pages that contains documents and products of the project available for free.

Unfortunately all national standards are in Persian. FarsiWeb project is translating the important ones to English, but this is not done yet.

None of the discussed issues in this paper have been covered fully with all details here, as that would be well beyond the bounds of this general overview. For more information you are invited to subscribe to the PersianComputing public mailing list where the FarsiWeb Project Group members as well as many other volunteers will help you find the answers to your questions and provide you with latest information. The list archives also are considered one of the best resources in this area.

Typing Persian Word Documents with Windows Tutorial is an excellent website on setting up Microsoft Word to produce Persian documents. It introduces most of the the problems on the platform and provides solutions under different versions of the system.

Acknowledgement

The author wish to thank C. Bobroff for taking the hard task of editing the final version of this paper multiple times.

About the FarsiWeb Project

The FarsiWeb project started as a research project in the Computing Center, Sharif University of Technology in early 1999, which later moved to a startup company called Sharif FarsiWeb, incorporated in late 2003, that still has its research lab at SUT.

FarsiWeb has close relations to all of the language and computing authorities of Iran and the Persian language, including the Persian Academy of Language and Literature, the High Council of Informatics, and ISIRI (the Iranian national standardization body). In the last five years, FarsiWeb has been representing those organizations in various standard bodies and international organizations, including the Unicode Consortium, ISO JTC1/SC2, World Wide Web Consortium, and IETF. It has helped refine the specifications of those bodies to incorporate the requirements of Persian and other languages written in Arabic script.

FarsiWeb is considered one of the main authorities of standard Persian computing, and has published many recommendations and a national Iranian standard (on Persian information interchange using Unicode) on matters related to implementation of Persian language, which has won the approval of all the authorities. A national standard on a national keyboard layout, a reference set of Persian fonts for web and printing usage, and a few specifications on requirements of standard Persian support on GNU/Linux platforms are under preparation.

In early 2003, FarsiWeb also co-developed a report on "Computer Locale Requirements of Afghanistan" with Everson Typography of Ireland, which won the approval of the transitional government of Afghanistan.

About the Author

Behdad Esfahbod is the maintainer and main developer of FriBidi, a Free Software implementation of the Unicode Bidirectional Algorithm. FriBidi is used in many Open Source projects including the GNOME desktop and AbiWord word processor, where it is used as a requirement for rendering scripts like Arabic, Hebrew, and Syriac. Behdad is also a key member of the FarsiWeb Project Group, working on tasks ranging from adding Persian support to Free Software applications around the world, to writing national standards on Persian computing issues.

Behdad is a member of Unicode Consortium's Bidi Committee, and a member of the FarsiTeX Project Team. He is currently pursuing graduate studies at the University of Toronto, Department of Computer Science. It is a pity that he enjoys mountain climbing so much, while there are only lakes around Toronto.