Perso-Arabic Input Methods

And Making More Emacs Applications BIDI Aware

Document #PLPC-180063
Version 0.1
November 02, 2021

This Document is Available on-line at:
http://www.by-star.net/PLPC/180063

Mohsen Banan — محسن بنان
Email: emacs@mohsen.1.banan.byname.net

Contents

List of Figures

1 Mohsen Banan Emacs Introduction — Video

Notes: Greetings. Salaam.

This is Mohsen Banan.

I am an Iranian software and internet engineer.

I converted to Emacs in 1986 – It was emacs version 17 then. By around 1988 when emacs version 18 was well in place, I started living inside of emacs. My primary digital environment has been emacs ever since. It has been a good life.

I am a native Farsi speaker and writer. I am not a linguist and I do not specialize in multilingalization, internationalization and localization.

My favorite programming language is lisp and I am a bit of an emacs developer.

2 Contours Of This Presentation

A quick overview of:

1. Perso-Arabic Scripts
2. Persian Input Methods
   - farsi-isiri-9147 — farsi-transliterate-banan
3. Making More Emacs Applications BIDI (Bidirectional) Aware
   - Problems and Challenges For Emacs Developers
4. Persian Blee (By* Libre-Halaal Emacs Environment)
   - Towards a complete convivial Persian digital environment

- Direction statements and preview of coming attractions

5. Pointers and References

Notes: This presenation is about use of Perso-Arabic Scripts with Emacs.

It is an overview presentation, I won't be digging deep in many of the mentioned topics. My goal is to make you aware of what can be done with emacs today and the potentials that emacs presents for Perso-Arabic writers.

The main topics that I'll cover are: - A brief intro to perso-arabic scripts - Two existing Emacs Persian Input Methods - The challenges involved with making emacs applictions bidirectional aware. - The ultimate goal of creating a complete digital environment for Perso-Arabic writers. - And I'll also be including various pointers.

3 Contours Of This Presentation And Intended Audience

Contours Of This Presentation

And Intended Audience

| **Perso-Arabic Writer** | **Non Perso-Arabic Writer** |

Emacs User:
Emacs Developer:
Considering To Convert To Emacs:
Persian Blee User:

Notes: So, first let's make sure that what I am presenting is of interest to you.

If you are a perso-arabic writer and if you use emacs, you are definitely my intended audiance.

If you are an emacs developer who wishes to make her emacs apps multilinugual and bidi aware, you are also my intended audiance.

4 Shaping And Bidirectionality

| Shaped Alphabet | Un-Shaped Alphabet | Non-Alphabetical |

Left-To-Right:
Right-To-Left:
Bidirectional:
Top-To-Bottom:

Notes:

For the purposes of this presenation, in this slide, I am categorizing scripts based on directioanlity and Shaping.

Latin letters are not shaped. Generally speaking the shape of a Latin letter is independent of its position in a word. Perso-Arabic letters are subject to shaping. For example, the letter mim — sounding similar to M — takes 3 shapes depending on whether it is in the beginning of a word, in the middle of a word or at the end of a word. I'll be showing more of how shaping works in an emacs session screencast later.

Shaping has ramifications for emacs application developers. For example, if you are combining initial letters to create a label, those letters can be shaped together – which is not what you want. In such cases you would need to explicitly keep them separate.

Latin based scripts are always left-to-right. Perso-Arabic scripts are right-to-left with letters but numbers are left-to-right. So, Perso-Arabic scripts are bidirectional (BIDI). Hebrew is also bidirectional. But Hebrew is not shaped.

More recently, it has become very common to mix Perso-Arabic and Latin text. This can become very confusing if paragraph directionality is not properly observed. I'll be providing some examples as screencasts.

The Emacs display engine now fully and well supports both shaping and BIDI.

5 Emacs: A Truely Multilingual Capable Editor And Environment

---

Emacs: A Truely Multilingual Capable

Editor And Environment

Emacs Fully Supports Perso-Arabic Scripts:

- Unicode: Since 1990s
- Many Quail based input methods: Since 1990s
- BIDI (Bidirectional): Since 2012 (v24)

- Shaping: Since 2000s. With Harfbuzz since v27
- Various Persian Input Methods: Since 2012 (v24)

Notes:

Since 2012, starting with emacs v24, we can say that Emacs is a truely multilingual capable environment.

Like everything else, multilingual support for emacs was added gradually.

Unicode support was added early on.

The framework for input methods evolved in the 1990s.

But it was not till v24 in 2012 that the display engine could fully support bidi. Hats off to Eli Zaretskii for his work on emacs bidi.

Once, full bidi support was in place, in 2012, I went ahead and added two Persian Input Methods to emacs 24.

So, now Emacs Fully Supports Perso-Arabic Scripts.

6 Significance Of Emacs Support For Perso-Arabic Scripts

- By Perso-Arabic script we mean Arabic script with various extensions used for writing Arabic languages, Persian families of languages and several other languages.
- Languages that use Perso-Arabic script as their writing system include: Arabic, Farsi, Dari, Urdu, Afghan, Pashto, Kurdish, Balochi, Lurish, Kashmiri.
- Perso-Arabic is the *second* most widely used writing system in the world by the *number of countries*.
- It is the *third* by the *number of users*, after the Latin and Chinese scripts.

Notes:

By Perso-Arabic script we are refereing to the Arabic writing system with various extensions used by a large number of languages.

Perso-Arabic is the second most widely used writing system in the world by the number of countries.

It is the third by the number of users, after the Latin and Chinese scripts.

So, by well supporting Perso-Arabic, Emacs's potential user base can be greatly enhanced.

Part I

Persian Input Methods

7 About Emacs Input Methods

Input Methods allow you to enter characters that are not supported by your keyboard. With Quail maps we can map ASCII key strings to multilingual characters. So, we can input any text from an ASCII keyboard.

Main facilities are:

- `M-x set-input-method`: Select input method — C-x C-m C-\
- `M-x describe-input-method`: C-h C-\or C-h I
- `M-x toggle-input-method`: C-\

Notes:

Before focusing on the persian input methods, let me quickly summarize Emacs's input methods model.

Input Methods allow you to enter characters that are not supported by your keyboard. With Quail maps we can map ASCII key strings to multilingual characters. So, we can input any text from an ASCII keyboard.

You select an input methods with `C-x C-m C-\`.

We'll try that in a screencast shortly.

8 Emacs Persian Input Methods

Emacs comes with two built-in Persian input methods:

farsi-isiri-9147: A Persian keyboard based on the Islamic Republic of Iran's ISIRI-9147 specification.

This is the traditional one-to-one mapping of keys on a computer keyboard to Persian letters.

farsi-transliterate-banan: An intuitive transliteration keyboard for Farsi.

> This is a more powerful method which converts sequences of characters into one letter. For example "kh" becomes خ.

Notes:

Since version 24, emacs comes loaded with two persian input methods.

farsi-isiri-9147 is the standard traditional iranian keyboard.

farsi-transliterate-banan is an intuitive transliteration keyboard for Farsi which requires near zero training for use.

I'll be mostly focusing on farsi-transliterate-banan in this presentation.

So, let's try these out.

9 Selecting Persian Input Methods — Gif-Screencast

Notes: In this gif-cast, we are going to select a persian input method and write a few simple sentences. With no training and no documentation, any farsi writer familiar with emacs can write these as the farsi-transliterate-banan input method is intuitive.

I'll be using keycast to show you keys as thay are used. Let me first describe as to what we have on the screen. There are three windows in one frame. Key cast will show commands and keys on the mode-line. The left most window is showing logs of the keycast. Transformed individual unshaped letters will appear here. The middle window is running a tail -f dribbleFile | fold -w 1. This let's you see the raw ascii characters as I type them. The right window is the emapty buffer on the ex.fa file. Anything that I describe here can be done with virgin emacs distribution with nothing added. I am using Blee (By* Libre-Halaal Emacs Environment) to show things, but you don't need that for basic writing.

First, I am going to select the farsi-transliterate-banan. I am entering C-x enter C-backslash. Notice the mode-line and the prompt at mini-buffer. With completion, I am going to select farsi-transliterate-banan. Notice that farsi-isiri-9147 was also provided as a choice.

Also, notice that the letter beh appears in the left of mode-line of ex.fa. This indicates which input method has been selected. Also notice that curser is on the top left corner of ex.fa.

Next, I am going to enter the 's' character. Notice, the curser moved to the right, and unshaped 'seen' appeared in the ex.fa buffer and on the mode-line and in the keycast-

log buffer. Next, I am going to enter the 'l' character. Notice ل in the mode-line and notice how the س was subjected to shaping.

Next, I am going to write the following: سلام ـــحال شما چه طوره؟ ـــبا ایمکس همه کار میشه کرد! — Hello, How are you? You can do everything with Emacs.

Generally, same sounding lating characters are used. As usual, vowels are ignored, unless called for. Notice that in order to get ح, I repeated 'h' twice. ش is the obvious 'sh'. چ is the obvious 'ch'. ط is upper case T.

That is it. We managed to write in farsi with a querty keyboard, intuitively.

Next we are going to switch back to globish. And write "Back to Globish".

Notice that the globish sentence, started from the left side. This is due to proper detection of paragraph directionality by emacs.

10 Emacs Built In Documentation

Emacs is a self-documenting editor. Input methods' keyboard layouts can be displayed with `describe-input-method` and BIDI is well documented in Emacs Manual.

- farsi-isiri-9147:: (describe-input-method 'farsi-isiri-9147)
- farsi-transliterate-banan:: (describe-input-method 'farsi-transliterate-banan)
- Emacs Manual: Bidirectional Editing:: https://www.gnu.org/software/emacs/manual/html_node/emacs/Bidirectional-Editing.html
- Emacs Manual: Bidirectional Display:: https://www.gnu.org/software/emacs/manual/html_node/elisp/Bidirectional-Display.html

---

Notes: For the most part, Emacs is self-documenting.

Here we are pointing you to some relevant self contained emacs resources.

The BIDI documenation applies to all bidi scripts, not just perso-arabic.

Refering to the code can also be useful for some.

11 Pointers To Code

The code for the quail keyboard mappings are at:

- Part of Emacs Installation At: /usr/share/emacs/27.1/lisp/leim/quail/persian.el.gz
- As a Git Repo At: https://github.com/bx-blee/persian-input-method

---

Notes: Here are some pointers to the quail translation code for Persian input methods.

The persian.el file has full details of the mapping and some documentation.

Next, we'll show the keyboard layouts as a gif-cast.

12 Keyboard Layouts For Persian Input Methods — Gif-Screencast

---

Notes: You can get relevant documentation for any input method with the describe-input-method command.

So, let's try that for farsi-transliterate-banan.

We are back in the ex.fa buffer as one window. We don't need the keycast logging and the dribble windows anymore. With a C-backslash, I reactivate the farsi input method. Notice that keycast is still active on the mode-line.

Next, with a C-H C-backslash, I get the input method's documentation. I then delete other windows and keep the help buffer visible.

Notice that beh is this input-method's identifier.

Here is the url for full documentation on the web.

The keyboad layout itself is a one-to-one mapping, but towards making transliteration intuitive, multiple keys are sometimes mapped to the same letter. For example both 'i' and 'y' can produce yeh.

The usual two letter transliterations ending with 'h' – zh, ch, sh and kh are provided.

The '&' prefix is used to support often invisible bidi markings.

In additiion to this internal documentation, full documenation is also available.

13 Complete Documentation

Complete Documentation For Persian Input Methods Is PLPC-120036: http://mohsen.1.banan.byname.net/PLPC/120036

Persian Input Methods
For Emacs And More Broadly Speaking
شیوه‌هایِ درج به فارسی

Various Related Information Is Also Available At PersoArabic.org: http://www.persoarabic.org

---

Notes: Complete documenation for Persian Input Methods is available as PLPC-120036.

Next, we'll take quick look at this on the web.

14 Persian Input Methods Full Documentation — Screencast

Notes: You can click on links in the Reveal web-based form of this presentation.

So, let's visit PLPC-120036. This document fully describes Persian Input Methods. In addition to html, you can also obtain it in pdf. Let's do that.

Of particular interest in this document are various tables that enumerate lists of letters with their association to both Persian input methods.

Let's take a look at few of these.

Table 3, Mapping of isiri-6219 (The Farsi Character set) to emacs persian input methods could be of interest to you.

As well as table 8, for bidi related control mark ups.

And table 9, for vowels and other signs.

Part II

Making More Emacs Applications BIDI Aware

15 Ramification Of BIDI And Perso-Arabic On Emacs Applications

---

BIDI And Perso-Arabic related glitches (or more than glitches) in various Emacs Apps:

- Gnus
  - For perso-arabic, Gnus columns don't line up in the Summary mode.
  - For perso-arabic, Subject and from fields should repect direction in Summary mode.
- bbdb
  - Paragraph and field directionality is not respected.
- calendar and calfw
  - For calendar, print persian and print islamic, can now produce perso-arabic letters. (Starting point in place)
  - For calfw, perso-arabic entries don't line up and don't respect direction.
- AUCTeX, XeLaTeX
  - For right-to-left documents, the resonable approach is to create persian aliases for all LaTeX commands. (Starting point in place)

Notes:

Having covered input methods, let's turn our attention to ramifications of BIDI and Perso-Arabic on various Emacs applications.

Since 2012, I have been using persian text in various emacs applications.

In short my experience has been that most emacs apps are usable, but they all have glitches that could at a minimum annoy perso-arabic users.

In this slide, I am presenting a summary.

The glitches with Gnus, are not all that significant for me.

The bbdb glitches can easily be fixed.

For calendar, I have customized my own setup to support persian and islamic dates in perso-arabic. Perhaps they should be merged upstream.

Instead of dealing with apps one at a time, I think it is more reasonable to consider them collectively.

16 BIDI Aware Emacs Applications

Making More Emacs Applications

BIDI And Perso-Arabic Aware

- BIDI Aware Topics
  - Consider use of explicit specificatin of directioanlity at buffer, paragraph and field level. For example, bbdb can easily be fixed to allow for right-to-left fields.
  - Sometimes Combine left-to-right and right-to-left to convey more information. For example, In calendar applications, Christian and Islamic dates can face each other on the same line.
  - Use explicit html direction specification in email and other communications. While emacs detects paragraph directionality properly, Firefox and Chrome don't.
- Perso-Arabic Topics
  - Shaping makes fixed width character counting impractical. Count pixels not letters. Gnus columns don't line up in the Summary mode. Subject and from fields should repect direction in Summary mode.
  - Use zero-width non-joiner (ZWNJ) to avoid undesired shapings (e.g., when using initials).

Notes:

The glitches that I mentioned in the previous slide, have two roots.

Some are bidi specific and some are perso-arabic specific.

In this slide, I have classified them as such and have made some general suggestions.

But, all of these at best amount to tactical approaches. I think a more strategic approach is called for.

17 Emacs Native Markup Language (ENML)

Emacs Native Markup Language - ENML:

A Model For Apps Development

Let's put support for BIDI in an apps dev framework that all Emacs apps can use.

- Let's create "Emacs Native Markup Language – ENML" as a lispish super-set of html5.
- Let's mimic the web apps development frameworks in Emacs.
- Let's consider ENML as the primary Native Emacs Mode.
- Let's use ENML in all basic emacs buffers. Help buffers, doc-strings, etc.
- Let's merge ENML and org-mode.
- Let's make ENML BIDI aware.
- Let's transition all emacs-apps to use ENML.

Notes:

The right way, to address bidi-awareness and other awarenesses is to build them in frameworks that emacs apps can then use.

So, I am proposing that we first create ENML, the Emacs Native Markup Language, as a lispish (perhaps even not fully secure) super-set of html5.

With that in place we can then build on the 2 decades of experince that have produced various web application development frameworks by mimicing one of them.

I don't have any running code for any of this.

But discussing startegy need not always be futile.

Part III

About Persian Blee

18 About ByStar And BISOS

---

http://www.by-star.net

The Libre-Halaal By* (ByStar) Digital Ecosystem

For Preservation Of The Individual's Autonomy and Privacy

A Moral Alterantive To The Proprietary American Digital Ecosystem

http://www.by-star.net/PLPC/180054

Notes: Emacs has immense potentials. But those potentials can not be realized unless we integrate emacs in the totality of a specific complete digital ecosystem.

Over the past two decades I have been building the contours of The Libre-Halaal By* (ByStar) Digital Ecosystem.

Emacs can then be fully integrated into ByStar. It is through such integration that full conviviality of Emacs can be experienced.

19 About Blee And Persian Blee

Blee: ByStar Libre-Halaal Emacs Environment is full integration of:

- Emacs +
- Lots Of Emacs Apps
- Lots Of Blee Apps
- Lots Of Debian Facilities
- Lots Of ByStar Services
- Lots Of BISOS (ByStar Internet Services OS) Facilities

Persian Blee: Blee for Persian Speakers

Notes:

Blee, the ByStar Libre-Halaal Emacs Environment, is emacs plus a whole lot of emacs apps integrated with Debian, with ByStar Services and with BISOS, the ByStar Internet Services OS.

Perhaps this could be the topic of a presentation for the 2022 Emacs Conference.

References