

# Настройки системы и Emacs для RTL

Васил Дядов

Среда 18 июля 2018 г.

## Содержание

<b>1</b>	<b>Установка opam и необходимых OCaml пакетов</b>	<b>1</b>
1.1	Установка opam . . . . .	1
1.2	Установка нужной версии OCaml . . . . .	1
1.3	Установка пакетов OCaml . . . . .	2
<b>2</b>	<b>Установка texlive и pygments</b>	<b>2</b>
<b>3</b>	<b>Установка PlantUML</b>	<b>2</b>
<b>4</b>	<b>Установка Emacs</b>	<b>3</b>
<b>5</b>	<b>Настройка Emacs</b>	<b>3</b>

## 1 Установка opam и необходимых OCaml пакетов

### 1.1 Установка opam

```
1 sudo pacman-S opam
```

Существует два пакета opam: версии 2.0 и 1.2. Версия 2.0 пока ещё экспериментальная и в активной разработке, её ставить не рекомендуется.

### 1.2 Установка нужной версии OCaml

```
1 opam switch install 4.06.1
```

### 1.3 Установка пакетов OCaml

- Инициализация переменных окружения

```
1 eval $(opam config env)
```

- Установка пакетов

```
1 opam install \  
2   hardcaml \  
3   ppx_hardcaml \  
4   ppx_deriving_hardcaml \  
5   merlin
```

## 2 Установка texlive и pygments

- Установка texlive

```
1 sudo pacman -S \  
2   texlive-bin \  
3   texlive-core \  
4   texlive-fontsextra \  
5   texlive-langcyrillic \  
6   texlive-langextra \  
7   texlive-latexextra \  
8   texlive-pictures \  
9   texlive-pstricks \  
10  texlive-science
```

- Установка pygments

```
1 sudo pacman -S pygmentize python-pygments
```

Pygments нужны для раскраски исходного кода latex-пакетом `minted`.

## 3 Установка PlantUML

- Установка сборщика пакетов из AUR

```
1 sudo pacman -S yaourt
```

- Установка PlantUML

```
1 yaourt -S plantuml
```

## 4 Установка Emacs

```
1 sudo pacman -S emacs
```

## 5 Настройка Emacs

Данные кусочки кода нужно последовательно записать в файл `.emacs`.

- Настройка MELPA В начало `.emacs` нужно добавить следующий код для подключения репозитория MELPA:

```
1 (require 'package)
2 (add-to-list 'package-archives
3   (cons "melpa"
4     "https://melpa.org/packages/") t)
5 (package-initialize)
```

После добавления вышеприведённого кода, нужно перезапустить Emacs.

- Установка пакетов Следующий код нужно выполнить в каком-либо буфере `emacs-lisp` командой `eval-buffer`.

```
1 (package-install 'maude-mode)
2 (package-install 'merlin)
3 (package-install 'org)
4 (package-install 'plantuml-mode)
5 (package-install 'tuareg)
6 (package-install 'caml)
```

- Настройка файла `.emacs`

- Подключение и настройка экспорта из org в pdf через latex:

```
1 (require 'ox-latex)
2 (add-to-list 'org-latex-packages-alist
3   ' (" " "cmap" t))
4 (add-to-list 'org-latex-packages-alist
5   ' ("english,russian" "babel" t))
6 (add-to-list 'org-latex-packages-alist
7   ' (" " "libertine" t))
8 (add-to-list 'org-latex-packages-alist
9   ' (" " "unicode-math" t))
10 (add-to-list 'org-latex-packages-alist
11   ' (" " "listings" t))
12 (add-to-list 'org-latex-packages-alist
13   ' (" " "minted" t))
14 (add-to-list 'org-latex-packages-alist
15   ' (" " "color" t))
16 (setq org-latex-inputenc-alist
17   ' (("utf8" . "utf8x")))
```

- Загрузка языковых пакетов для org-babel:

```
1 (org-babel-do-load-languages
2   'org-babel-load-languages
3   ' ((ocaml . t)
4     (plantuml . t)
5     (emacs-lisp . t)
6     (ditaa . t)
7     (latex . t)
8     (shell . t)))
```

- Настройка включения/выключения картинок:

```
1 (defun do-org-show-all-inline-images ()
2   (interactive)
3   (org-display-inline-images t t))
4
5 (global-set-key (kbd "C-c C-x C v")
6   'do-org-show-all-inline-images)
7
8 (setq org-startup-with-inline-images t)
```

- Не спрашивать подтверждение для выполнения кода:

```

1 (setq org-confirm-babel-evaluate
2   'my-org-confirm-babel-evaluate)

```

– Настройка переноса длинных строк:

```

1 (add-hook 'org-mode-hook 'turn-on-auto-fill)
2 (add-hook 'org-mode-hook
3   '(lambda() (set-fill-column 80)))

```

– Конфигурирование merlin-mode режима для редактирования OCaml кода:

```

1 (let ((opam-share
2       (ignore-errors
3         (car (process-lines
4               "opam" "config" "var" "share")))))
5   (when (and opam-share
6             (file-directory-p opam-share))
7     ;; Register Merlin
8     (add-to-list 'load-path
9                 (expand-file-name
10                  "emacs/site-lisp"
11                  opam-share))
12     (autoload 'merlin-mode "merlin" nil t nil)
13     ;; Automatically start it in OCaml buffers
14     (add-hook 'tuareg-mode-hook 'merlin-mode t)
15     (add-hook 'caml-mode-hook 'merlin-mode t)
16     (add-hook 'merlin-mode-hook
17               '(lambda()
18                  (local-set-key (kbd "<backtab>")
19                                'auto-complete)))
20     ;; Use opam switch to lookup ocamlmerlin binary
21     (setq merlin-command 'opam))
22
23 (setq merlin-ac-setup 'easy)

```

– Настройка кастомизированных переменных:

- \* `enable-local-variables` - разрешение всех локальных переменных файла (может быть небезопасно, если открывать чужие файлы, будьте осторожны с этой настройкой)
- \* `org-export-with-sub-superscripts` - символ подчёркивание не трактуется как субскрипт

- \* `org-caption-above` - подписи листингов кода и картинок будут внизу
- \* `org-latex-compiler` - установка компилятора latex-кода
- \* `org-latex-listings` - latex-пакет для генерации листингов исходного кода
- \* `org-latex-pdf-process` - настройка параметров вызова latex-компилятора
- \* `org-plantuml-jar-path` - путь к plantuml java-файлу, для генерации PlantUML диаграмм
- \* `org-src-preserve-indentation` - будет ли сохранять отступы и форматирование `org-babel` при редактировании блоков исходного кода
- \* `package-selected-packages` - список установленных нами пакетов, чтобы они не были удалены при автоматическом удалении ненужных пакетов

```

1 (custom-set-variables
2   '(enable-local-variables :all)
3   '(org-export-with-sub-superscripts (quote {}))
4   '(org-latex-caption-above nil)
5   '(org-latex-compiler "xelatex")
6   '(org-latex-listings (quote minted))
7   '(org-latex-pdf-process
8     (quote
9       ("%latex -8bit -shell-escape -interaction\
10 nonstopmode -output-directory %o %f"
11        "%latex -8bit -shell-escape -interaction\
12 nonstopmode -output-directory %o %f"
13        "%latex -8bit -shell-escape -interaction\
14 nonstopmode -output-directory %o %f")))
15   '(org-plantuml-jar-path
16     "/opt/plantuml/plantuml.jar")
17   '(org-src-preserve-indentation nil)
18   '(package-selected-packages
19     (quote
20       (merlin tuareg plantuml-mode
21         org auto-complete))))

```