

Расширенные концепции Vim

# Как расширить функциональность Vim, добавив плагины

Функциональность Vim можно расширить с помощью плагинов. Вот как можно устанавливать и использовать плагины в Vim. Также узнайте об обновлении и удалении плагинов.

Маск прибыл на мероприятие в Калифорнии на прототипе роботакси

СЕ  
ИГ



Будучи одним из самых популярных текстовых редакторов, Vim также влияет на другие современные текстовые редакторы. Фактически, в Microsoft Visual Studio Marketplace есть расширение Vim для VS Code; оно установлено более 4 миллионов раз .

А как насчет улучшения самого Vim? Что ж, это тоже возможно! Функциональность Vim можно расширить с помощью "плагинов" . Давайте посмотрим, как можно добавлять плагины в Vim.

Существует два основных способа установки плагинов Vim.

- Самый простой способ – использовать менеджер плагинов. Это можно считать аналогом менеджера пакетов, но для Vim.
- Второй способ – вручную клонировать репозитории в определенные каталоги.

Я бы не советовал устанавливать плагины вручную. Ручной метод может стать запутанным и сложным, когда количество плагинов увеличивается. Он может испортить ваши плагины Vim, если вы по ошибке клонируете плагин в неправильный каталог.

Итак, я покажу, как можно добавлять плагины в Vim с помощью менеджера плагинов.

## Менеджер плагинов Vim

Как я уже упоминал ранее, использование менеджера плагинов для Vim окупается в долгосрочной перспективе, когда у вас есть несколько плагинов. Итак, давайте посмотрим, как установить и использовать менеджер плагинов Vim.

Вам доступны несколько менеджеров плагинов Vim на выбор. Некоторые из самых популярных:

- vim-плагин
- VAM (vim-аддон-менеджер)
- Удобрять
- vim-патоген

Поскольку я сам использую менеджер плагинов vimplug, именно его я и буду использовать в этой статье для демонстрации.

## Установка менеджера плагинов vim-plug

Чтобы установить менеджер плагинов vim-plug, вам нужно всего лишь выполнить одну из следующих команд:

```
curl -fLo ~/.vim/autoload/plug.vim --create-dirs https://raw.githubusercontent.com/j
```

Вот и все! Менеджер плагинов `vim-plug` теперь установлен.

## Установка плагинов с помощью менеджера плагинов `vim-plug`

В этом уроке я буду использовать менеджер плагинов `vim-plug` для установки следующих плагинов Vim:

- [`fzf.vim`](#) : нечеткий искатель
- [`rust.vim`](#) : Подсветка синтаксиса для языка Rust
- [`NERDTree`](#) : проводник по древовидным файлам



---

— Расширенные концепции Vim / Как расширить функциональность Vim, добавив плагины

---

```
call plug#begin()
```

```
Plug 'junegunn/fzf.vim'
```

```
Plug 'preservim/nerdtree'
```

```
Plug 'rust-lang/rust.vim'
```

```
call plug#end()
```

В приведенном выше фрагменте кода есть три отдельных раздела.

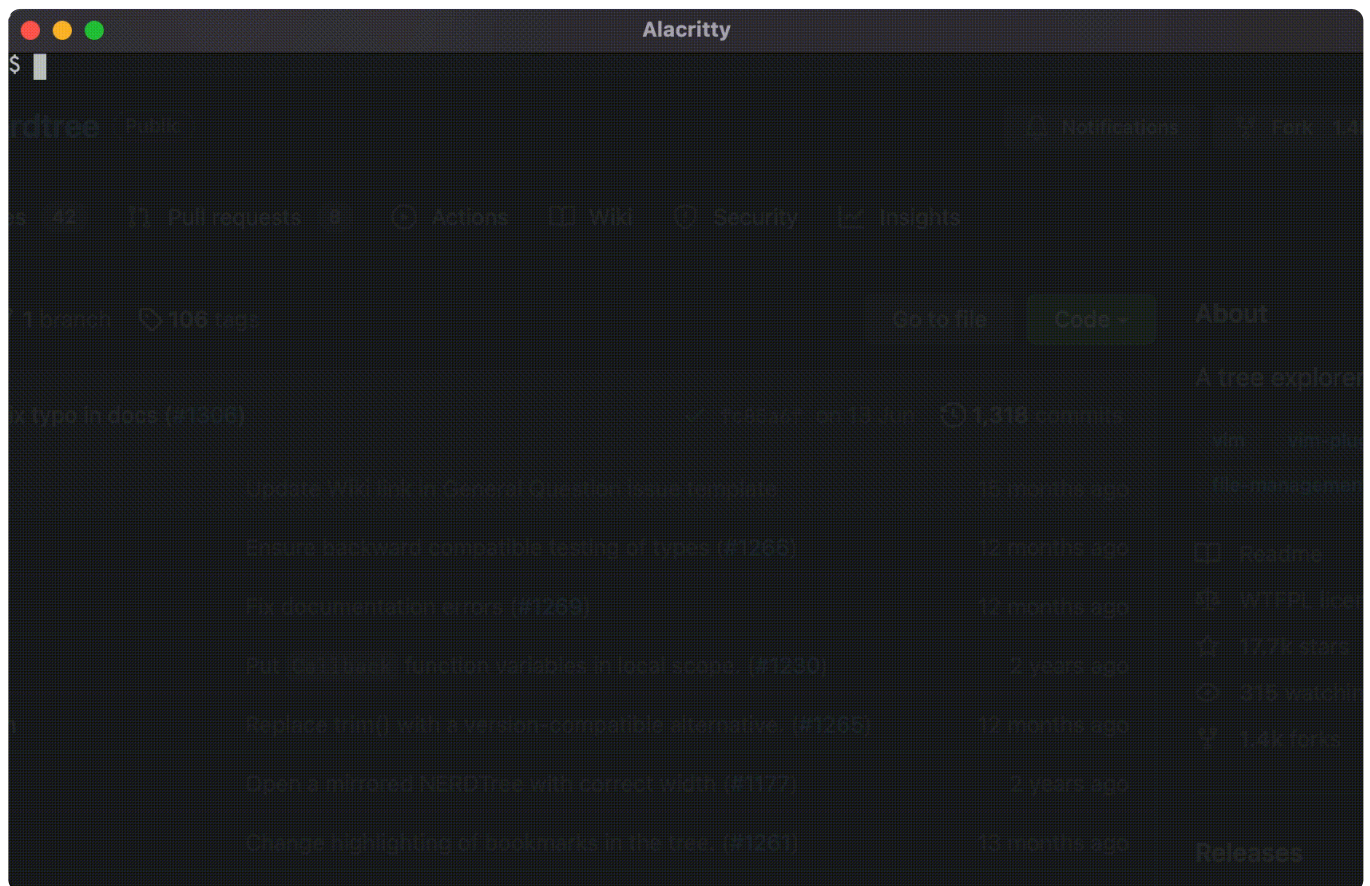
Первый раздел — `call plug#begin()`. Эта строка сообщает Vim, что мы начинаем раздел, в котором определим все плагины, которые хотим установить.

Вторая, третья и четвертая строки имеют узор `Plug 'user/repo'`. Чтобы понять, что это значит, мы должны вернуться к тем плагинам, которые я сказал, что установим. Первый плагин был [`fzf.vim`](#). Если вы перейдете по данной гиперссылке, она откроет проект в Github. Если вы немного знакомы с Github, вы поймете, что владельцем/автором плагина `fzf.vim` является 'junegunn', а имя проекта - 'fzf.vim'. Теперь, если вы посмотрите на приведенный выше фрагмент кода, вы заметите, что `user` было заменено на 'junegunn' и `repo` было заменено на 'fzf.vim', полная строка - `Plug 'junegunn/fzf.vim'`. Вот как указывается плагин. Я сделал то же самое, чтобы добавить `rust.vim` и плагин `NERDTree` Vim.

Наконец, последний раздел `call plug#end()` – сообщает Vim, что все плагины определены и что раздел плагинов окончен.

После того, как плагины определены в `.vimrc` файле, их необходимо загрузить и установить. Но не волнуйтесь, вам не нужно загружать их вручную.

Чтобы загрузить и установить плагины, откройте Vim и введите `:PlugInstall` команду. Это откроет панель слева, и вы увидите в реальном времени процесс клонирования и установки плагинов.



После завершения установки вы увидите строку `Finishing ... Done!` в левой панели. Закройте Vim и откройте снова, чтобы плагины были загружены в холодном виде.

Ваши плагины установлены, ура!!!

## Обновление установленных плагинов

Как вы видели, `:PlugInstall` команда установит все плагины, которые определены в `.vimrc` или в `init.vim`. Но как обновить плагины, которые уже установлены?

Ну, в таком случае используйте `:PlugUpdate` команду. Использование этой команды обновит все плагины.

Допустим, из всех установленных вами плагинов вы хотите обновить только несколько, выборочно. Чтобы обновить определенные плагины, вы можете добавить имена плагинов после `like` `:PlugUpdate` так:

```
:PlugUpdate plugin0 plugin1
```

Итак, чтобы обновить только плагин NERDTree, я бы выполнил следующую команду Vim:

```
:PlugUpdate nerdtree
```

## Обновление самого менеджера плагинов

Команда Vim предоставляется вместе с менеджером плагинов `vim-plug` для обновления самого менеджера плагинов. Для этого выполните следующую команду:

```
:PlugUpgrade
```

## Удаление установленных плагинов

Чтобы удалить плагин, удалите строку определения плагина для данного плагина.

Например, у `fzf.vim` меня в файле был определен плагин `.vimrc`; я удалю эту строку.

После этого снова откройте Vim и выполните `:PlugClean` команду. Это удалит все плагины, которые установлены, но не перечислены в `.vimrc` файле.

```
:PlugClean
```

# Заключение

В этой статье я показываю, как можно расширить функциональность Vim с помощью плагинов. Я также показываю, как можно обновлять и удалять эти плагины, включая обновление самого менеджера плагинов.

Надеюсь, вы найдете это полезным для правильного использования плагинов в Vim. Дайте мне знать, если у вас есть какие-либо вопросы.

←

Предыдущая - Расширенные концепции Vim  
Полное руководство по буферам в Vim

Далее - Расширенные концепции Vim  
Используйте vimdiff как профессионал Vim

→





## Пратам Патель


Навигация по миру Rust, RISC-V, Podman. Учимся, делаем, и делимся, записывая.


📍 Гуджарат, Индия


Какова ваша реакция?


 10

 0

 0

 0

 0

 0

превосходно

любовь

ух ты

грустно

смех

злой

Мы призываем к вежливому и цивилизованному обсуждению.

Write your comment...


Вершина 



V Вайбхав. 1 месяц назад

Здравствуйте, хорошее краткое руководство.



Опечатка: в обоих примерах для PlugUpdate ошибочно используется «PlugInstall».

 0  0 Ответить



Абхишек Команда ЛХБ 1 месяц назад

Действительно. Спасибо, что заметили. Исправил сейчас.

 0  0 Ответить

R Рис 5 месяцев назад



Очень ясно и понятно, спасибо, бро! 🙌🙌 И есть примеры, у хороших людей благополучная жизнь

 0  0 Ответить



Абхишек Команда ЛХБ 5 месяцев назад



Спасибо :)

 0  0 Ответить



Руди 9 месяцев назад

Отлично! Очень ясно и точно. Спасибо

 0  0 Ответить



Абхишек Команда ЛХБ 9 месяцев назад



Рад, что тебе понравилось, Руди :)

 0  0 Ответить



Роксана 11 месяцев назад

Сначала у меня были проблемы. Мне пришлось использовать sudo с командой curl, чтобы заставить его загрузиться и создать каталоги на Ubuntu 22.04 по какой-то причине, но теперь все работает отлично.

 0  0 Ответить



Абхишек Команда ЛХБ 11 месяцев назад

Интересно... но хорошо, что теперь это работает :)

👍 0 🗨 0 Отвечать



Независимое, поддерживаемое читателями издание, посвященное командной строке Linux, серверам, самостоятельному размещению, DevOps и облачному обучению.

### Подпишитесь на рассылку новостей LHB Linux Digest

Получите подборку советов, руководств и мемов по Linux прямо на свой почтовый ящик. Более 18 000 пользователей Linux пользуются им дважды в месяц.

[Контакт](#) [Профессиональные ресурсы](#) [Часто задаваемые вопросы о членстве](#)

[Информационные бюллетени](#) [политика конфиденциальности](#) [Напишите нам](#)



Справочник Linux © 2018 - 2024 • Размещено на DigitalOcean , опубликовано Ghost , с использованием темы Spiritix