Emacs 编辑器对于我的启发与帮助

应急管理大学

文一

内核
一周一审

Emacs 编辑器简介

~/project/plan/emacs-introduce.md

```
# Hello Emacs

Emacs 是一款由自由软件基金会孵化运营的开源编辑器软件，
以其优秀的可拓展性著称于软件行业。

他的作者是 Richard Stallman，其发起的 GNU 计划影响
极其深远。

U:---  emacs-introduce.md    All    (7,10)
```

如我们在左图所展示的那样，Emacs 是一款编辑器软件，以其强大的可拓展性闻名于开源软件世界。

而建立对于 Emacs 编辑器的深刻了解，必须要同时要对 GNU 计划建立认识，这项工程由许多软件组成，目标在于打造一个值得信赖的可供所有计算机用户使用的**自由软件生态系统**（更进一步：终结专有软件）。

GNU Emacs Logo

GNU Logo

我在 Emacs 社区的探索历程

（Emacs 自带帮助）

学习 Emacs 的源动力，来自于对于《大教堂与集市》、《程序员修炼之道》的阅读，而起步的教材，则是 Emacs 自带的帮助指南（为什么 Vim 最终没能吸引到我：第一，我更喜欢 Emacs 的键盘操作模式以及这种纯粹的代码编辑感；第二，Emacs 与 Vim 实际上是殊途同归，两者用不同的方式办成了一样的事情）。

但是光靠自带的指南，最多只能记住一些普通的光标指令，这就需要我们阅读进一步的进阶教程，并且积极主动同社区沟通。（毕竟，“**与世隔绝对于你的职业生涯可能是致命的**”）。

我在 Emacs 社区的探索历程

有那么夸张吗？我是不是还没有说过这能节省大量时间？千真万确：以一整年为跨度，即使编辑效率只提高了4%，只要每周花在编辑上的时间有 20小时，你每年就能凭空多出一周时间。

但这还不是真正的好处——真不算是。如果你操作编辑器游刃有余，最主要的收益来自于变得“顺畅”——不必再把心思花在编辑技巧上面。从思考到将想到的东西呈现在编辑器中的整个过程，没有阻塞，一气呵成。思考变流畅，编程就会受益。（如果你教过新手开车，就能理解他们和老司机的确不一样。新手一定会去仔细考虑要做的每个动作，而老司机则是靠本能在开车。）

一旦你发掘出一个新的有用的特性，需要尽快把它内化成一种肌肉记忆，这样在使用的时候就能不假思索。据我们所知，能做到这点的唯一方法只有不断重复。有意识地寻找机会使用这些新获取的超能力，最好是一天多次。一周左右，你会发现自己已经在下意识地运用。

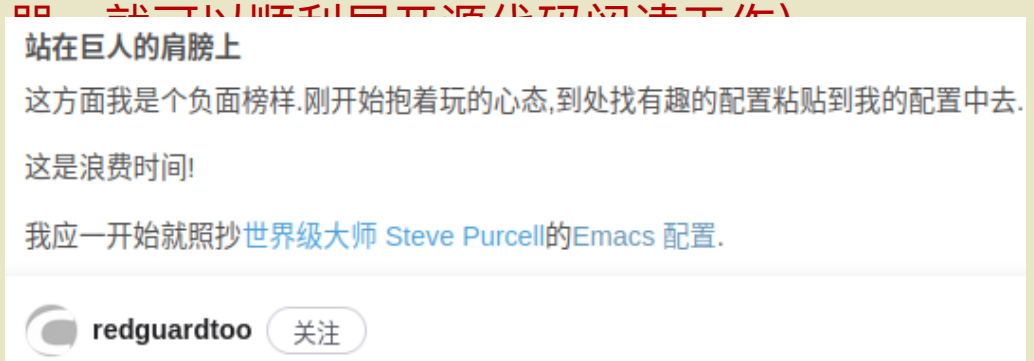
培育你的编辑器

大多数强力代码编辑器都是基于一个基础核心创建的，之后可以用扩展的方式不断增强这一核心。有些扩展是编辑器自带的，有些则留待日后添加。

（《程序员修炼之道（第二版）》的有关论述）

感谢陈斌的《一年时间成为 Emacs 高手》，这段话让我直接选择起 Purcell 的 Emacs 配置，这就让我少走了很多弯路。

而 Emacs China 社区则解答了我的很多 Emacs 问题，让我逐渐走上了正规（从中也可以瞥见 Emacs 的发展，在 eglot 进入 Emacs 之后，我可以直接在项目工程下运行 eglot 结合 lsp 服务



使用 Emacs 阅读大型工程的经验与想法

【咨询】转向LSP-BRIDGE以后，如何阅读大型项目的源代码?

Emacs-general

BeginnerC 2022 年 11 月

经过一番折腾以后，我再次转回 Emacs 阵营，因为：

1. Emacs 提供了一种整体的美感，尽管感觉不如 VSCODE 那么“专业”，但是它提供了一种“立刻写代码”的快感
2. 我以后的主力开发语言应该就是 C语言，因为我计划以后从事 PostgreSQL 相关的数据库开发工作（纯C），再不需要什么复杂的东西
3. 还是舍不得 Emacs China 这么好的社区（哈哈哈哈）

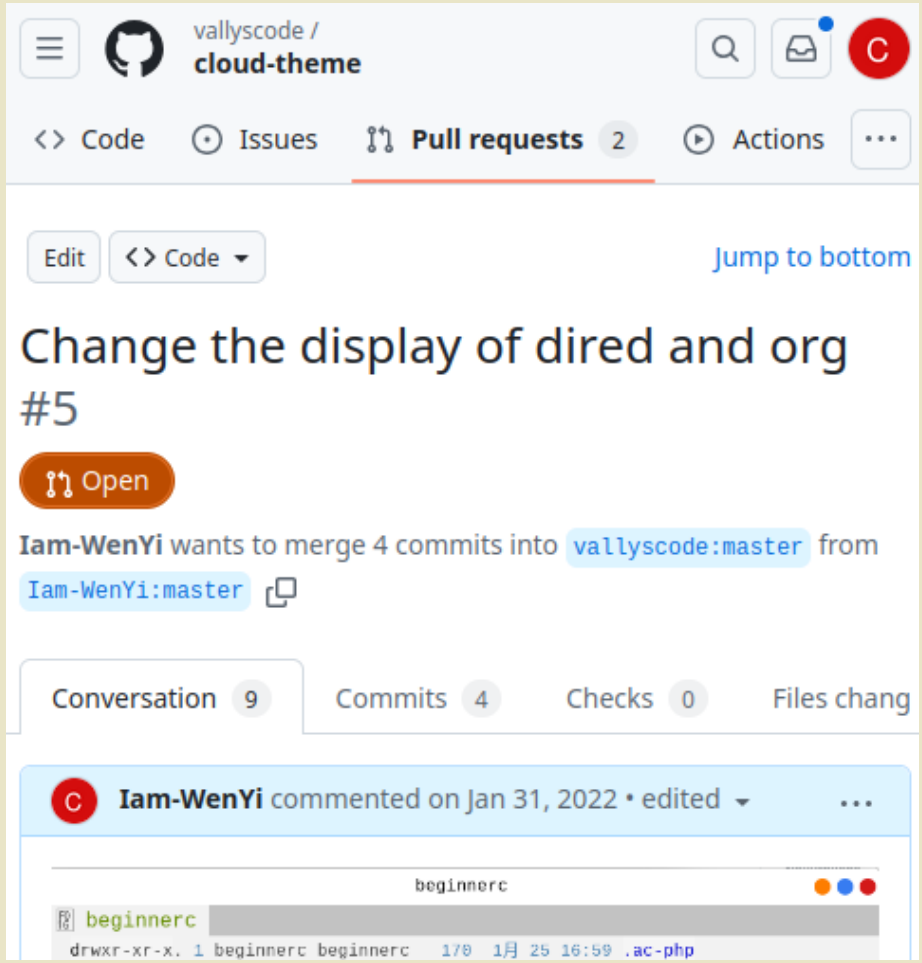
然后我刚刚再次安装了 Emacs 并且启动了配置，现在我下载了一套 PostgreSQL 的数据库源代码，并安装了 Lsp-Bridge 进行源代码解析。但是，我发现 Lsp-Bridge 似乎“不能理解整个项目”的代码逻辑关系（因为我并不清楚怎么为Lsp-Bridge指定“一个项目的代码解析”）（它在单文件工作下很好）。有没有一个解决方案呢？还是说我只能如之前那样，手动生成TAGS文件呢?

最早，感谢 @oldosfans 的指教，使用 etags 为项目生成 TAGS 文件（可以理解为一种对于项目的“标记文件”，静态的，如果项目出现更改，要整体重建这个静态文件），并以此展开工作。

后面，转向 lsp-server，认为它相较于 TASG 更加灵活。

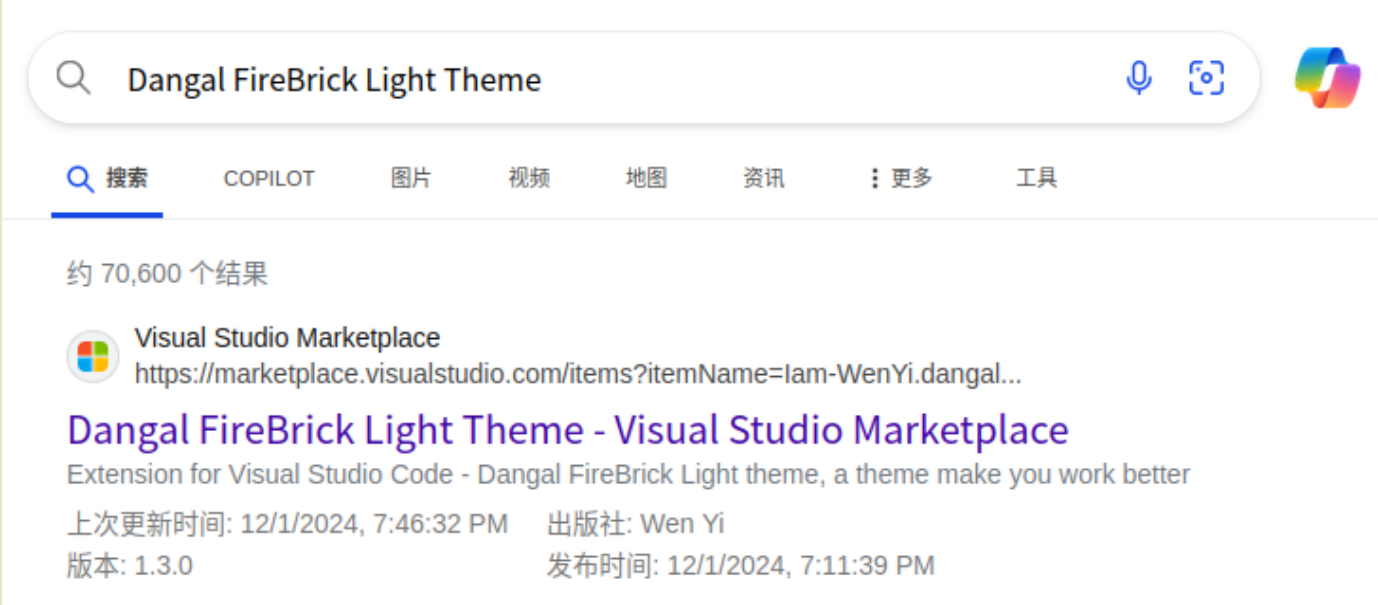
再后面，意识到想要理解大型工程，最重要的并不是一头扑入源代码之中，而是在于理解项目本身的设计目标，乃至于为了实现这一个目标的基本思路，再在这个基础上了解更细节的信息。

应用 Emacs 中所学，改造 VSCODE

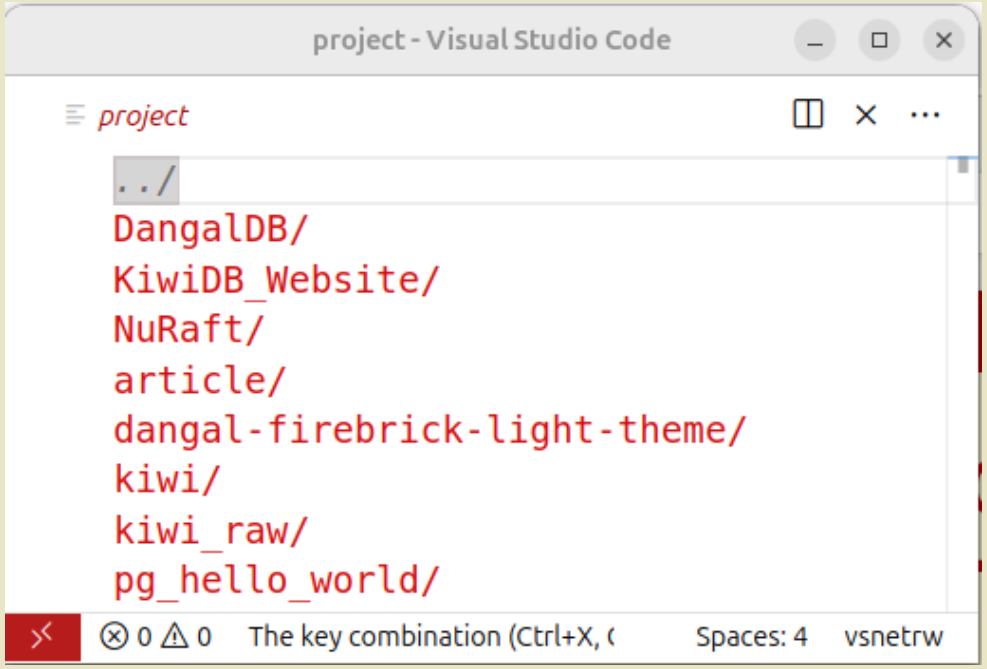


2022年7月，曾经参与改进 Emacs 主题 “cloud-theme”，这为我理解编辑器主题拓展的原理，奠定了扎实的基础。

2024年12月，迁移经验到 VSCODE 中，自己设计了 “Dangal FireBrick Light Theme”（雄心壮志火砖红 亮色主题）。



应用 Emacs 中所学，改造 VSCODE



除了 Theme 的设计之外，我的 VSCODE 也都是参照着 Emacs 的习惯而进行打造的。

因此，使用 VSCODE 拓展插件 “vsnetrw”，调整 VSCODE 文件浏览模式为类 Emacs “dired”。

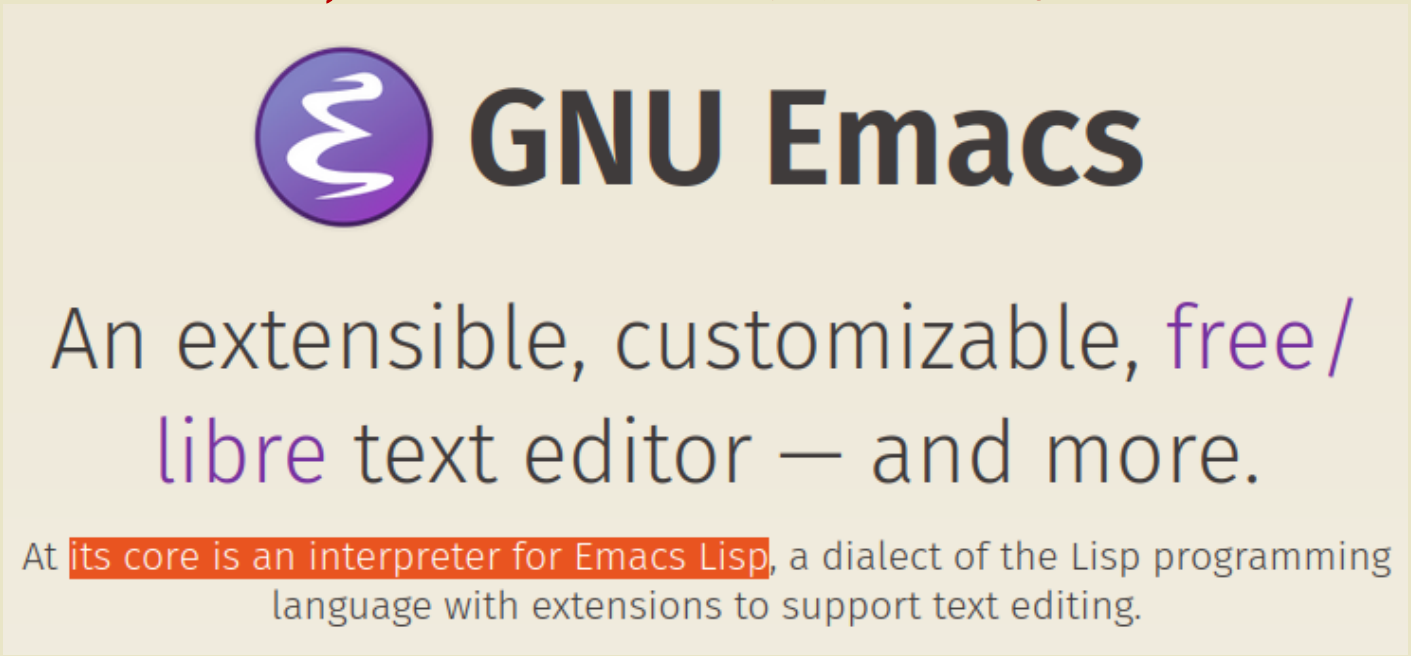
使用 “Emacs Friendly Keymap”，调整为 Emacs 键位（不过还需要结合自定义设置，go to definition 这几个选项需要手动配置）

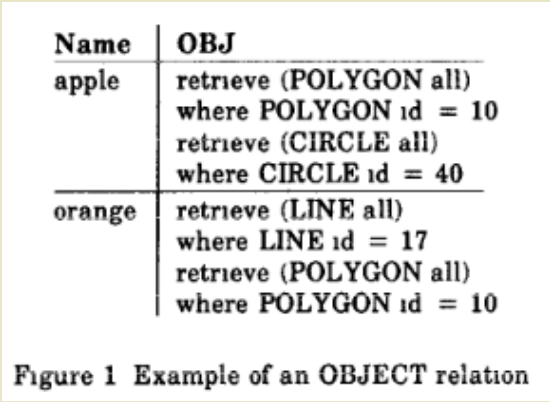
@feature:terminal，将 Terminal 的 Default Location 调整为 “Editor”。

这样你的 VSCODE 就可以得到类似于 Emacs 的编

Emacs 对于其它方面软件开发的理解

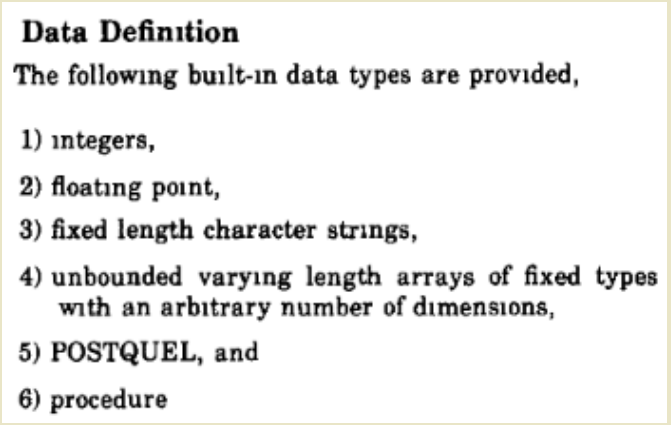
除了编辑器本身的理解之外，Emacs 的拓展机制原理也促进了我理解其它软件的拓展机制，乃至于一些经典的文章。





Name	OBJ
apple	retrieve (POLYGON all) where POLYGON id = 10 retrieve (CIRCLE all) where CIRCLE id = 40
orange	retrieve (LINE all) where LINE id = 17 retrieve (POLYGON all) where POLYGON id = 10

Figure 1 Example of an OBJECT relation



Data Definition

The following built-in data types are provided,

1) integers,
2) floating point,
3) fixed length character strings,
4) unbounded varying length arrays of fixed types with an arbitrary number of dimensions,
5) POSTQUEL, and
6) procedure

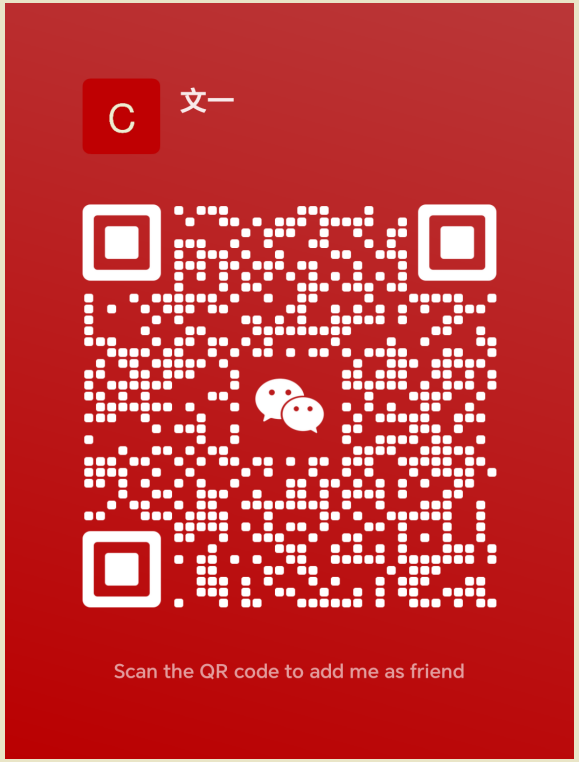
写在最后

感谢我的本科生导师，袁国铭博士，中国 PostgreSQL 分会的魏波老师、王其达老师，开放原子开源基金会的张凯老师，IvorySQL 社区的任娇老师、牛世继老师、王守波老师，OpenTenBase 社区的符芬菊老师、单致豪老师，EmacsTalk 的刘家财老师，Emacs 社区的王勇老师，青学会的吴洋老师，PingCAP 的陈小伟老师，TiDB 版主严少安老师，云和恩墨的彭冲老师，Kiwi 社区的于雨老师、刘月财老师，华中科技大学开源俱乐部的王振辰同志，OERV 团队的邱文键同志，@Makio，@红发哥，@可可，@轩轩，@进化哥，@大炮，@M佬，黄宏亮老师。

我们将继续前进，为建成一个更为开放繁荣的开源数据库生态而作出更多的贡献。

谢谢你们，祝各位工作顺利。

祝各位工作顺利，谢谢你们



欢迎加我微信

EmacsTalk
社区公众号

Thanks