

Introduction to Slidev

Presentation slides for developers



_Kerman

Slidev 团队成员

USTC 本科 2023 级



⚡ github.com/KermanX

𝕏 x.com/KermanX_

𝕏 bsky.app/profile/KermanX.github.io

Yet Another Slides Maker?



./slides.md

```
# Welcome to Slidev!
```

```
Present|
```

Welcome to Slidev!

Present

Math?

Supports K^AT_EX, a subset of L^AT_EX.

$$\nabla \cdot \vec{E} = \frac{\rho}{\varepsilon_0}$$

$$\nabla \cdot \vec{B} = 0$$

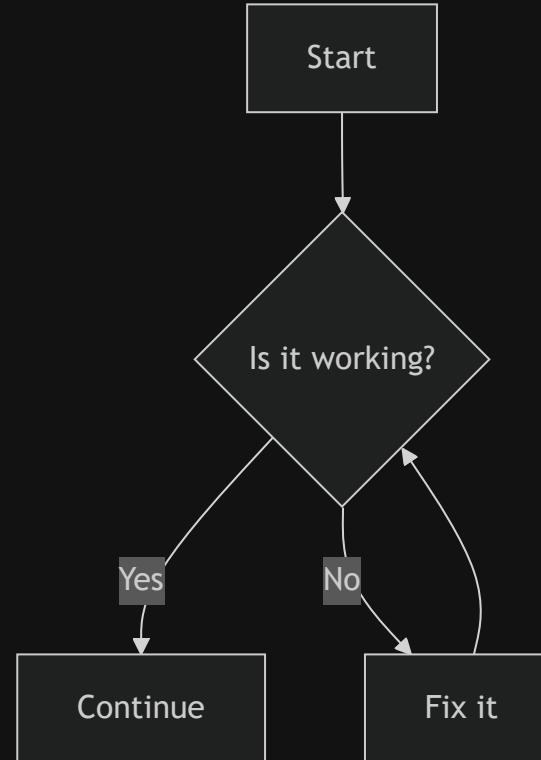
$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}$$

$$\nabla \times \vec{B} = \mu_0 \vec{J} + \mu_0 \varepsilon_0 \frac{\partial \vec{E}}{\partial t}$$

Graph?

Mermaid.js / PlantUML / Custom renderer

```
```mermaid
graph TB
 A[Start] --> B{Is it working?}
 B -- Yes --> C[Continue]
 B -- No --> D[Fix it]
 D --> B
``````
```



Code? Even editable!

"Monaco Editor"!

```
import { ref, watchEffect } from 'vue'

const a = ref(1)

watchEffect(() => {
|   console.log(a.value)
})
```

Interactivity?

$$z = x^2 + y^2 = 2x2 + 4x4 = 20$$



Animation & 3D?

Slidev

by Alvaro Saburido

Open Source & Free!

假设:

- 学生 (USTC)
- 没有/几乎没有 JavaScript / HTML / CSS 经验

目标:

- 对于学术报告, 提高制作效率:
 - 学术幻灯片只需要排版和代码展示, 基于文本的方式最为高效
 - 专注于文本内容, 自带优雅的样式
- 对于演讲, 使你的幻灯片更吸引人
 - 制作可交互的幻灯片
 - Slidev is Hackable, 只有想象力是限制

Your First Slidēv

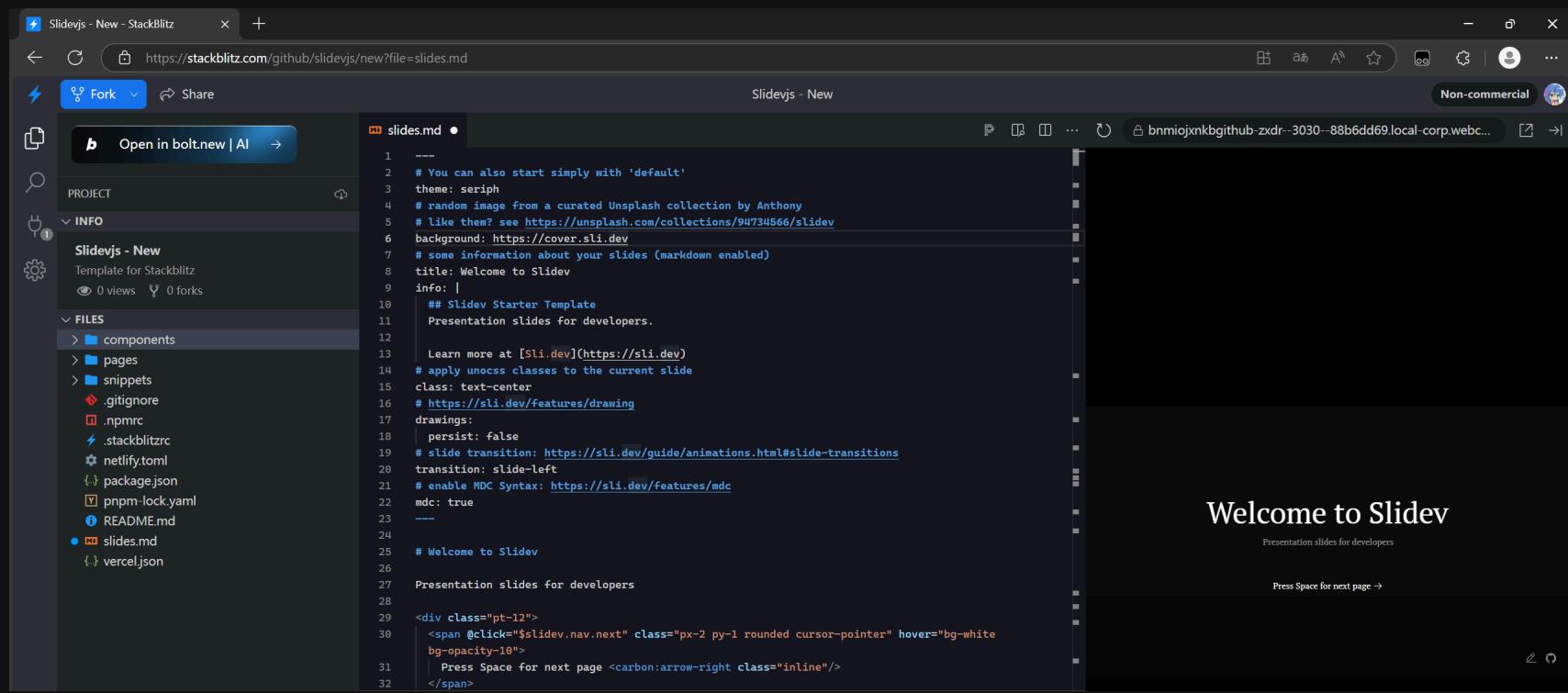
Option 1:

Stackblitz 在线环境

优点：无需任何配置，只需要浏览器

缺点：编辑体验不如本地环境

<https://sli.dev/new>



The screenshot shows the Stackblitz editor interface. On the left, there's a sidebar with project settings like 'Open in bolt.new | AI', 'PROJECT', 'INFO' (containing 'Slidevjs - New'), and 'FILES' (listing 'components', 'pages', 'snippets', '.gitignore', '.npmrc', '.stackblitzrc', 'netlify.toml', 'package.json', 'pnpm-lock.yaml', 'README.md', 'slides.md', and 'vercel.json'). The main area has tabs for 'slides.md' and 'Non-commercial'. The code editor shows the content of 'slides.md', which includes a YAML front matter section and some HTML. The preview pane on the right shows a slide with the title 'Welcome to Slidev' and the subtext 'Presentation slides for developers'. At the bottom, there are navigation controls and a message 'Press Space for next page →'.

```
---  
# You can also start simply with 'default'  
theme: serif  
# random image from a curated Unsplash collection by Anthony  
# like them? see https://unsplash.com/collections/94734566/slides  
background: https://cover.sli.dev  
# some information about your slides (markdown enabled)  
title: Welcome to Slidev  
info: |  
  ## Sliddev Starter Template  
  Presentation slides for developers.  
  
Learn more at [Sli.dev](https://sli.dev)  
# apply unocss classes to the current slide  
class: text-center  
# https://sli.dev/features/drawing  
drawings:  
  persist: false  
# slide transition: https://sli.dev/guide/animations.html#slide-transitions  
transition: slide-left  
# enable MDC Syntax: https://sli.dev/features/mdc  
mdc: true  
---  
# Welcome to Slidev  
Presentation slides for developers  
  
<div class="pt-12">  
  <span @click="$slide.nav.next" class="px-2 py-1 rounded cursor-pointer" hover="bg-white  
  bg-opacity-10">  
    Press Space for next page <carbon:arrow-right class="inline"/>  
  </span>
```

Option 2:

本地环境

1. 安装 Node.js

- 打开 <https://nodejs.org/>
- 点击下载按钮并安装

2. 换源 (可选)

打开终端，运行以下命令

```
npm config set registry https://registry.npmmirror.com
```

本地环境

创建 Slidev 项目

- 打开终端（桌面/文件夹中右键，选择“在终端中打开”）
- 创建 Slidev 项目

```
npm create slidev@latest
```

- 进入项目目录

```
cd my-slidev
```

- 打开 VSCode

```
code .
```

本地环境

运行 Slidenv

打开终端（推荐使用 VSCode 内置终端），执行以下命令：

```
npm run dev
```

会自动打开一个浏览器窗口，显示你的幻灯片

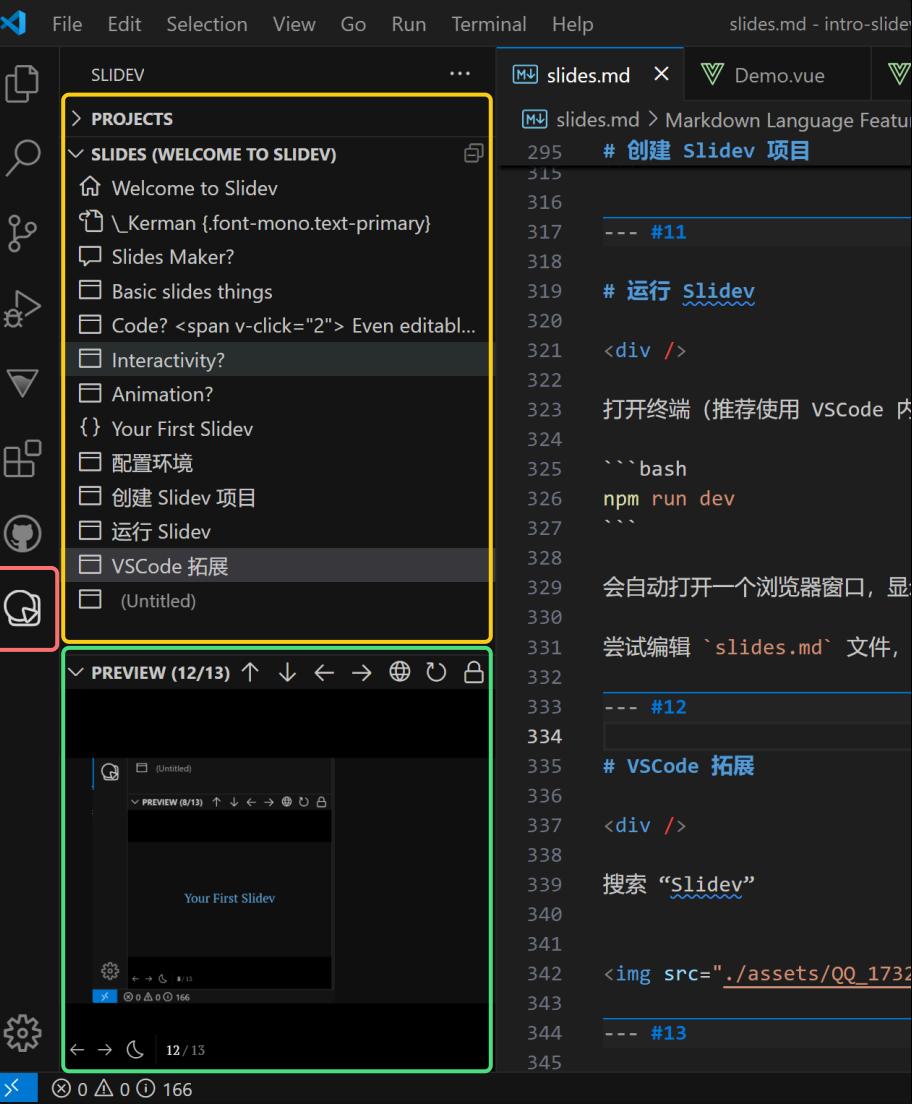
尝试编辑 `slides.md` 文件，保存后浏览器会自动更新内容

本地环境

VSCode 拓展

搜索“Slidev”，安装拓展

- 点击 Slidev 图标
- 幻灯片导航
- 幻灯片预览



用户界面

导航栏

-  全屏播放
-  上一步/下一步
-  幻灯片列表
-  打开摄像头
-  录制演讲
-  绘图
-  编辑幻灯片

快捷键

-   SPACE 上一步/下一步
-   上一张/下一张 (跳过动画)
-  F 全屏



语法介绍

Markdown

幻灯片的内容使用 Markdown 语法编写

```
# 标题
```

- 列表项 1
- 列表项 2

粗体 *斜体* ~~删除线~~

[链接](https://sli.dev)

标题

- 列表项 1
- 列表项 2

粗体 *斜体* ~~删除线~~

链接

HTML & Vue

HTML 和 Vue 提供更强大的灵活性

```
<div style="color: red">  
  Some HTML  
</div>  
<div v-for="i in 3">{{ 2**i }}</div>
```

Some HTML

2

4

8



UnoCSS

CSS 和 UnoCSS 提供更多的样式选择

需要一些学习。可能比 LaTeX 更简单和直观

```
<div text-blue> Text color </div>
```

Text color

```
<div text-2xl> Text size </div>
```

Text size

```
<div border="1 blue"> Border </div>
```

Border

```
<div bg="blue op-40"> Background </div>
```

Background

```
<div pl-8> Padding </div>
```

Padding

声明式 CSS

当然，我们也可以使用普通的 CSS

- 全局 CSS: `./style.css`
- 每个幻灯片的专属 CSS: `style` 标签:

```
# My slide

<style>
h1 {
  color: red;
  // Or: @apply text-red;
}
</style>
```

```
# This won't be red
```

常用布局

基于 Web 的幻灯片有更多的布局选择

- Grid: 静态分配空间

```
<div grid grid-cols-2 gap-4>
  <div border p-2> Left </div>
  <div border p-2> Right </div>
</div>
```

Left

Right

- Flex: 动态分配空间

```
<div flex gap-4>
  <div border p-2> Item </div>
  <div border p-2> Large Item </div>
  <div border p-2 flex-grow> Rest </div>
</div>
```

Item

Large Item

Rest

分隔幻灯片

以上是单张幻灯片的编写方式，如何编写多张幻灯片？

```
# 第一页
```

第一页的内容

```
# 第二页
```

第二页的内容

注：VSCode 拓展会使用蓝色线条突出分隔符

配置幻灯片

为幻灯片添加属性！

```
---  
theme: serif  
---
```

第一张幻灯片前的配置是 Headmatter

用于配置整个文稿的属性

```
# 第一张幻灯片
```

```
---  
layout: intro  
---
```

其他每页前的配置是 Frontmatter

用于配置单页的属性

```
# 第二页的布局是 intro!
```

```
---
```

Headmatter 和 Frontmatter 都是可选的

```
# 第三页没有任何配置
```

数学公式

行内公式: $\$1 + 1 = \text{textcircled}{9}\$$

行内公式: $1 + 1 = \textcircled{9}$

多行公式：

```

$$ \{all|1|2\}
\begin{aligned}
a\ x^2 + b\ x + c &= 0 \\
x &\leq \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}
\end{aligned}
$$

```

多行公式：

$$ax^2 + bx + c = 0$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

代码块

朴实无华的代码块，语法和 Markdown 相同

```
```js
console.log('Hello, Sliddev!')
```
```

```
console.log('Hello, Sliddev!')
```

带有动态突出显示的代码块

```
```js {1-2|3|4}
const a = 2;
const b = 3;
const c = a ** b;
console.log(c);
```
```

```
const a = 2;
const b = 3;
const c = a ** b;
console.log(c);
```

代码运行器

内置 JavaScript & TypeScript, Python 可安装插件

```
---  
addons:  
  - python-runner  
---
```

```
<!-- ... -->
```

```
```py {monaco-run}  
import pandas as pd
from termcolor import colored

print(colored("Sliddev", "blue"))
df = pd.DataFrame({
 'a': [1, 2],
 'b': [3, 4]
})
print(df)
```
```

```
import pandas as pd  
from termcolor import colored  
  
print(colored("Sliddev", "blue"))  
df = pd.DataFrame({  
    'a': [1, 2],  
    'b': [3, 4]  
})  
print(df)
```

Running...

点击动画

我也不知道为什么叫“点击动画”，其实就是一步一步的动画意思

```
<div> 1 + 1 = ? </div>
<div v-click> ⑨ ? </div>
<div v-click> No! </div>
<div v-after> It's 2. </div>
```

1 + 1 = ?

⑨ ?

No!

It's 2.

- `v-click` : 下一步才显示
- `v-after` : 和上一个 `v-click` 或 `v-after` 同时显示

列表动画

对于列表和表格，写一堆 `v-click` 也太麻烦了

```
<v-clicks>
```

- Option 1
- Option 2
- Option 3

```
</v-clicks>
```

```
<v-clicks>
```

ID	Data
--	----
1	A
2	B

```
</v-clicks>
```

- Option 1
- Option 2
- Option 3

ID	Data
1	A
2	B

不按顺序的动画

这也是一种常见的情况

```
<div v-click="2"> Third </div>
<div v-click="0"> First </div>
<div v-click="1"> Second </div>
```

Third
First
Second

编程式的动画

```
<div :class="{'text-red': $clicks > 2}">
  will change color
</div>
```

will change color

在 frontmatter 使用 `clicks: 3` 来控制动画步骤总数

其他很强大的动画类型

- Slide transition
- CSS view transition
- CSS transition property
- Motion

View Transition

很酷，对吧

```
---  
transition: view-transition  
---  
  
# 其他很强大的动画类型  
  
- <span view-transition-title> CSS view transition </span>  
  
---  
transition: view-transition  
---  
  
# <span view-transition-title> View Transition </span>
```

Icons

```
<carbon-logo-github />  
<i i-carbon-chart-column />
```



<https://icones.js.org/>

Icônes

Search category...

Material

- Material Symbols**
Google Apache 2.0
13287 icons
- Material Symbols Light**
Google Apache 2.0
13354 icons
- Google Material Icons**
Material Design Authors Apache 2.0
10955 icons
- Material Design Icons**
Pictogrammers Apache 2.0
7447 icons
- Material Design Light**
Pictogrammers Open Font License
284 icons
- Material Line Icons**
Vjacheslav Trushkin MIT
1085 icons

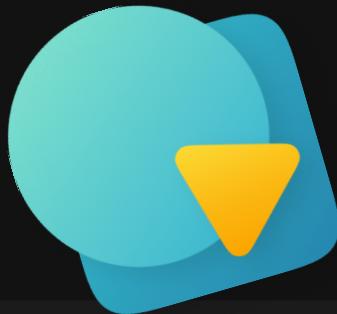
UI 24px

- Solar**
480 Design CC BY 4.0
7401 icons
- Tabler Icons**
Pawel Kuna MIT
5664 icons
- Huge Icons**
Hugeicons MIT
3929 icons
- MingCute Icon**
MingCute Design Apache 2.0
3040 icons
- Remix Icon**
Remix Design Apache 2.0
2892 icons
- Myna UI Icons**
Praveen Juge MIT
2378 icons

- IconaMoon**
Dariush Habibpour CC 4.0
- Iconoir**
Luca Burgio MIT
- Lucide**
Lucide Contributors ISC
- Lucide Lab**
Lucide Contributors ISC
- Unicons**
Iconscout Apache 2.0
- TDesign Icons**
TDesign MIT

Draggable Elements

Double-click on the draggable elements to edit their positions.



Double-click me!

DIRECTIVE USAGE

```

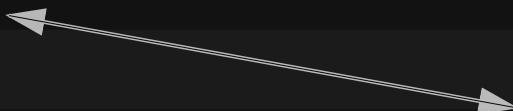
```

COMPONENT USAGE

```
<v-drag text-3xl>
  <div class="i-carbon:arrow-up" />
  Use the `v-drag` component to have a draggable container!
</v-drag>
```

DRAGGABLE ARROW

```
<v-drag-arrow two-way />
```



一些技巧

演示

- Option 1: **投屏 / HDMI**
 - 我的首选
- Option 2: **局域网**
 - 你的电脑启动 `npm run dev --remote`， 目标设备访问该地址
 - 不同设备，演讲者可以看演讲者视图，进度和绘图同步
 - 科大好像不支持
- Option 3: **导出到 PPTX / PDF**
 - 丧失交互性
 - 不得已的选择
- Option 4: **部署到 GitHub Pages 等**
 - 任何时候都可以访问
 - 无法同步进度和绘图

导出

```
npm run export --format pdf  
npm run export --format pptx  
npm run export --format png
```

会有提示要求你安装 Playwright，可能面临网络问题

PPTX 是一堆图片

导出有很多有用的选项，参见我们的文档

即将推出更易用、更稳定的导出方式，敬请期待！

(前几天才想到思路，来不及写完了 😊)

部署

编译成静态网站：

```
npm run build  
npm run build --base /folder/
```

(生成的代码在 `dist` 目录下)

或者，使用托管服务，解放双手：

- GitHub Pages / Netlify / Vercel
- Sliderv 官网有指南，非常方便

AI

- Bolt.new (有专门的 Slidev 选项)
- GitHub Copilot / Cursor / ...

```
799 --- #30
800
801 # AI
802
803 <div />
804
805 - Bolt.new
806
807 - GitHub Copilot You, 1 second
808
```

网络问题

- 代理
- 换源
- PNPM
- GitHub Codespaces

一些提示

- PPT 和 Slidev 的适用场景不同，有的时候 PPT 更合适
- 使用 Slidev 时，先确定是追求效率还是质量
 - 追求效率：Slidev 里只需要写文本，就可以获得不错的排版
 - 写好了文章，需要做汇报的情况
 - 追求质量：编写 CSS 和 Vue，可以做出完美效果，但需要更多时间
 - 专业演讲，需要更好的效果
- Slidev 中的高级功能需要一些 JavaScript / CSS 知识，如果没有基础，琢磨起来比较耗时可以在社区里（群里）提问

更多功能，参见文档 <https://cn.sli.dev>

The screenshot shows the Sliderv documentation website with a dark theme. At the top, there is a navigation bar with the Sliderv logo, a search bar, and links for '指南' (Guide), '功能' (Features), '参考' (Reference), '资源' (Resources), and more.

功能列表

本页是 Sliderv 提供的相对独立的功能列表。每一个功能都可以在有需要的时候单独使用，使你的幻灯片更加精彩。

如果你想按主题了解这些功能，也可以阅读 [index](#)。

Q、查找功能...

代码块形式的 frontmatter 使用单独的 YAML 代码块作为 frontmatter 语法	在构建时生成 PDF 在构建时生成可供下载的 PDF。 导出 构建	打包远程资源 在构建时下载和打包所有远程资源。 构建	幻灯片大小 设置所有幻灯片的大小。 布局
笔记的动画标记 根据点击动画高亮演讲者笔记。	Line Numbers 显示代码块中的行号。你可以选择为所有或者特定的代码块启用此特性。	代码块最大高度 为一个代码块设定最大高度并启用滚动。	基于导航方向的样式 根据幻灯片放映方向应用不同的样式和动画。

加入我们的社区

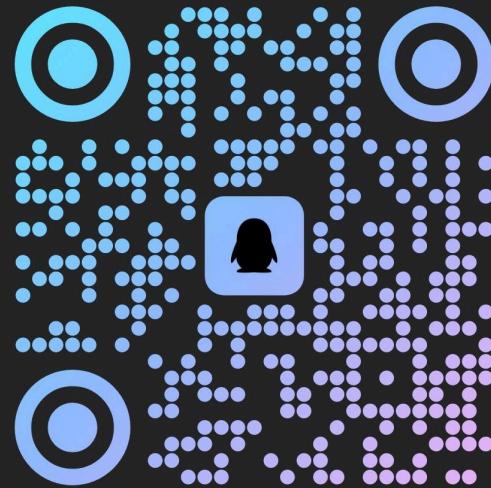
欢迎提 issue 和问题，也欢迎 PR

- 🌐 Website: <https://sli.dev>
- 🐳 GitHub: <https://github.com/slidevjs/slidev>
- 💬 Discord: <https://chat.sli.dev>
- 💬 QQ Group: 978643067



Slidev.js

群号: 978643067



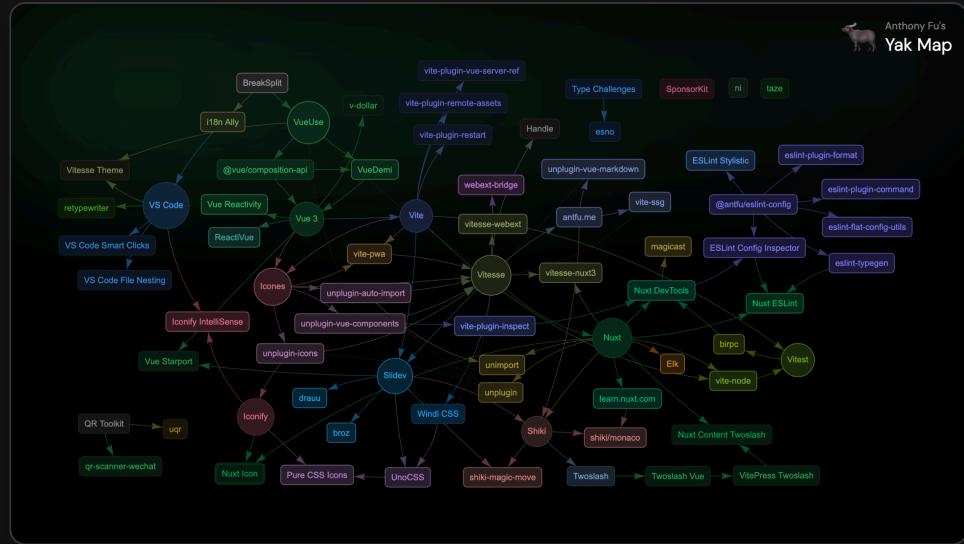
题外话

TypeScript 开源社区

什么是开源

- 字面意思：开放源代码
- 做慈善？ / 另一种商业模式？ / 玩？
- **开源是一种礼物**
互惠不是目的

<https://talks.antfu.me/2024/vueconf-shenzhen/3>
Yak Shaving by Anthony Fu



开源世界里不同的社区

- Linux 为代表的社区和 TypeScript 社区
 - 有相似之处
 - 成员上几乎没有交集
 - 有很多不同
- TypeScript / JavaScript 的特性
 - 脚本语言，更加难闭源
 - 大多数代码用于编写 UI，没必要隐藏代码
 - Web 世界开放共享的文化
 - “顺手就开源了” / “开源了没坏处”（当然，公开代码 ≠ 维护项目）

我参与开源的一年

- 第一次使用 Slidev 是在初中，那时 Slidev 刚发布不久，非常火爆
- 今年二月提交了第一个 PR，随后加入了 Slidev 团队
- 修复 bug、添加功能、重构代码、回答社区里的问题
- 甚至修复了一个我初中就碰到的 bug（很有趣的经历）
- 为 Slidev 编写 VSCode 拓展时，想到去开发 reactive-vscode 这个库
- 以此为契机，结识了很多朋友，包括 Slidev 的作者 Anthony Fu
- 参加了 VueConf 2024，Rolldown 团队成员告诉我“Rust 不难”
- 于是第三次入门 Rust，终于入门成功

一些信息

1. 开源是一种礼物。不要功利地看待，但坚信它对你的价值
2. 如果你想参与开源，JavaScript / TypeScript 社区是一个很好的起点
 - 门槛不高
 - 社区活跃
 - 华人参与度不错
3. 参与开源可能需要一个契机。从你的需求出发，找到一个感兴趣的项目

其他感想

1. GitHub 上的星星数只能作相对比较，不能作绝对比较
2. 那面绿墙只能说明这位用户是否还在使用 GitHub
3. 使用英文和 minimal reproduction

谢谢大家！