Tacos Overview & Lab0 Intro

TA Session

TA: Peng Zijun(彭子隽)
Email: 2200012909@stu.pke.edu.cn
Github: OshinoShinobu-Chan

TACOS
Rust OS never gets rusty.

Deadlines

- Lab 0 Code will be due next Thursday 11:59 pm
- Lab 0 Design Doc will due next Sunday 11:59 pm
- **Start early, because you never know what will happen.**

What is Tacos

- ~~A taco is a traditional Mexican dish~~

- Tacos is an educational operating system developed fully in Rust for RISC-V platforms

- So, what is Rust, what is RISC-V?

TACOS

Rust OS never gets rusty.

What is Rust

- ~~Rust is the best programming l~~
- A language empowering every ent
  software.
- Fast, reliable, productive.
- Inspired by many other langua

What is Rust

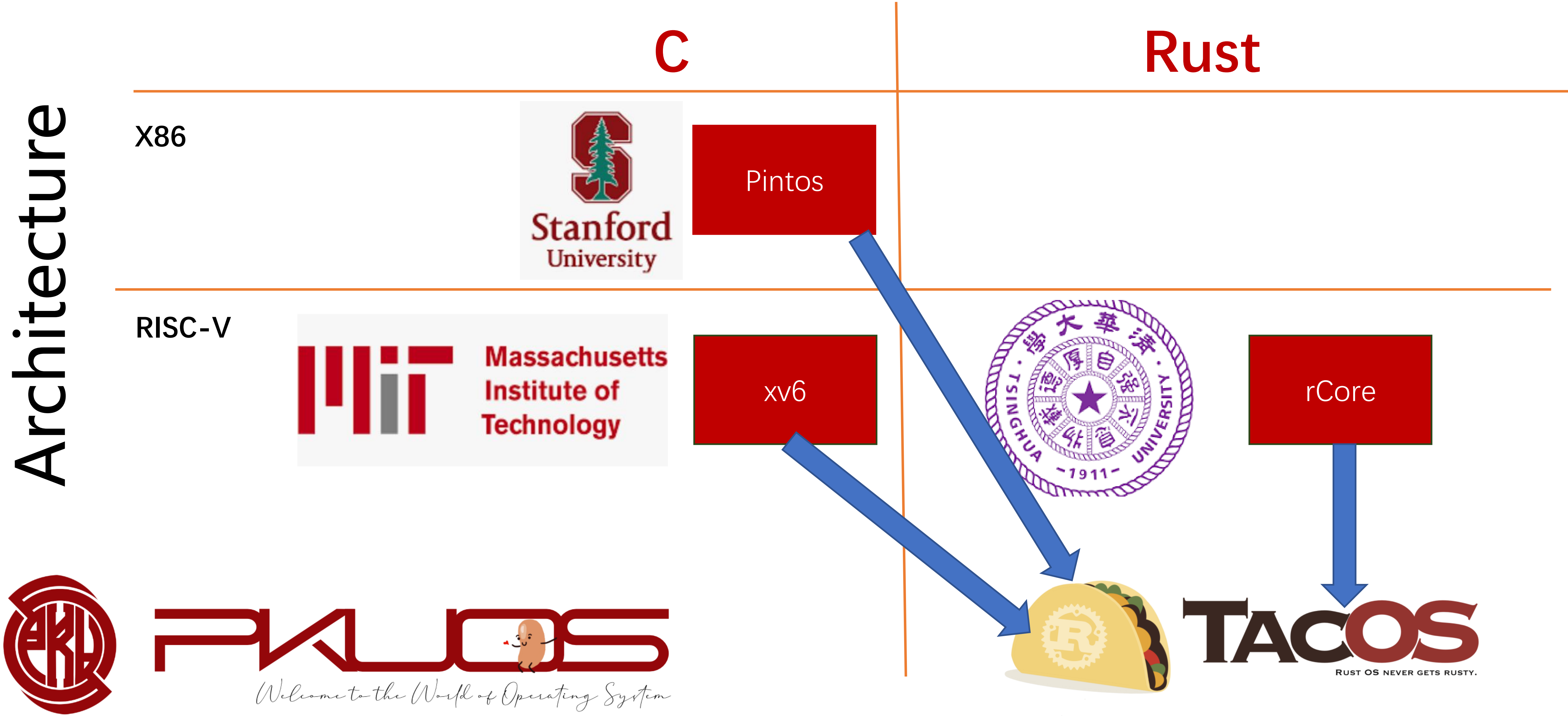
- System programming in Rust is a trend.
- Rust has been used in Linux kernel. (Rust for Linux)
- How to learn rust: *The Rust Programming Language*

Next several slides is from last year TA’s slides…

RISC-V is:

- ~~A RISC ISA~~
- from UC Berkeley
- was originally designed to support computer architecture research and education
- now will also become a standard free and open architecture for industry implementations.

Educational OSs



Programming Language

Architecture

C

Rust

X86

RISC-V

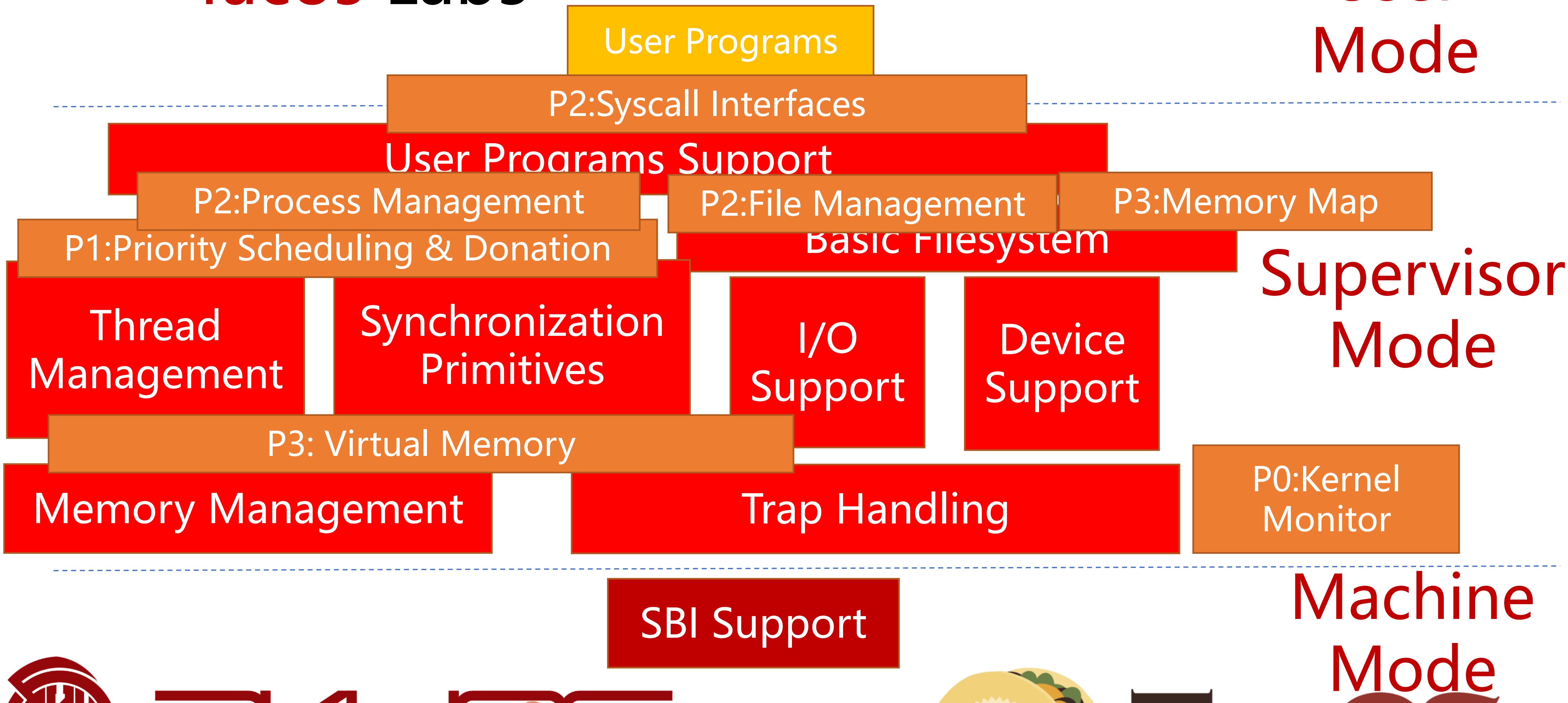
Pintos

xv6

rCore

Stanford University
Massachusetts Institute of Technology
TSINGHUA UNIVERSITY
1911
PKUOS
Welcome to the World of Operating System
TACOS
RUST OS NEVER GETS RUSTY.

Tacos Labs



User Mode

User Programs

P2:Syscall Interfaces

User Programs Support

P2:Process Management

P2:File Management

P3:Memory Map

P1:Priority Scheduling & Donation

Basic Filesystem

Supervisor Mode

Thread Management

Synchronization Primitives

I/O Support

Device Support

P3: Virtual Memory

Memory Management

Trap Handling

P0:Kernel Monitor

Machine Mode

SBI Support

Welcome to the World of Operating System

RUST OS NEVER GETS RUSTY.

Tacos is:

- an operating system developed fully in Rust for RISC-V platforms
- is designed to be modular
- has 4 challenging labs
- in active development! Welcome to join!

PKUOS

Welcome to the World of Operating System

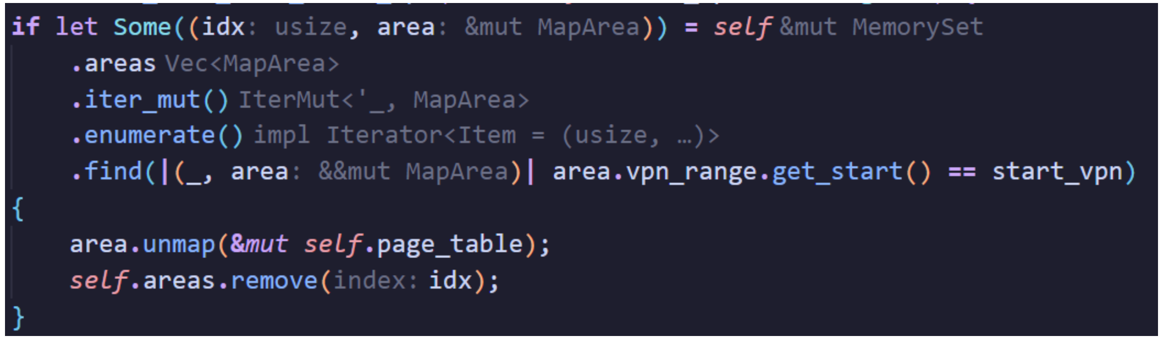
TACOS

RUST OS NEVER GETS RUSTY.

Tacos or Pintos

Vote for Tacos

- Rust is a beautiful language.
  - It has many useful modern features, like trait, Result, Option……
  - It has functional programming support. So you can use things like closure.
  - Rules like ownership and lifecycle helps you avoid bugs related to pointers.
    - However, it **won’t** reduce the complexity. It just forces you to make better design before getting start.



```
if let Some((idx: usize, area: &mut MapArea)) = self &mut MemorySet
    .areas Vec<MapArea>
    .iter_mut() IterMut<'_, MapArea>
    .enumerate() impl Iterator<Item = (usize, …)>
    .find(|(_, area: &&mut MapArea)| area.vpn_range.get_start() == start_vpn)
{
    area.unmap(&mut self.page_table);
    self.areas.remove(index: idx);
}
```

TACOS

Rust OS never gets rusty.

Tacos or Pintos

Vote for Tacos

- **RISC-V** is a modern ISA
  - Do not need to deal with legacies

- **Tacos** is born at PKU
  - You will have a detailed walkthrough of how Tacos is written from scratch.
  - If you have better designs, and want to change/rewrite components in **Tacos**, you will receive immediate help!

- rCore is also Rust+RISC-V, you can take look at it.

TACOS

Rust OS never gets rusty.

Tacos or Pintos

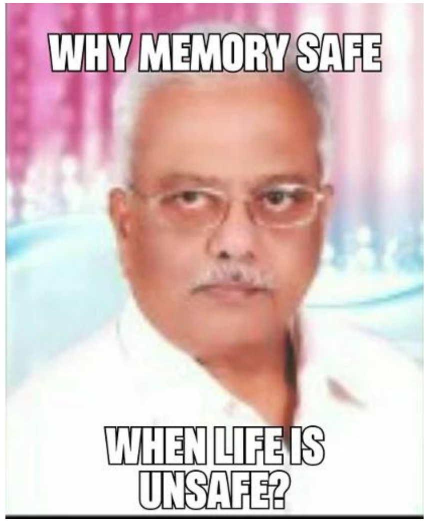
Vote against Tacos

- RISC-V and Rust is too modern
    - Potential bugs in compiler/emulator/toolchain.
        - I've been experienced a funny bug in Rust Analyzer.

- You have learnt x86 in ICS, but RISC-V is a new ISA for you.
    - There's totally different instructions.
    - You have to take some time to be familiar with RISC-V

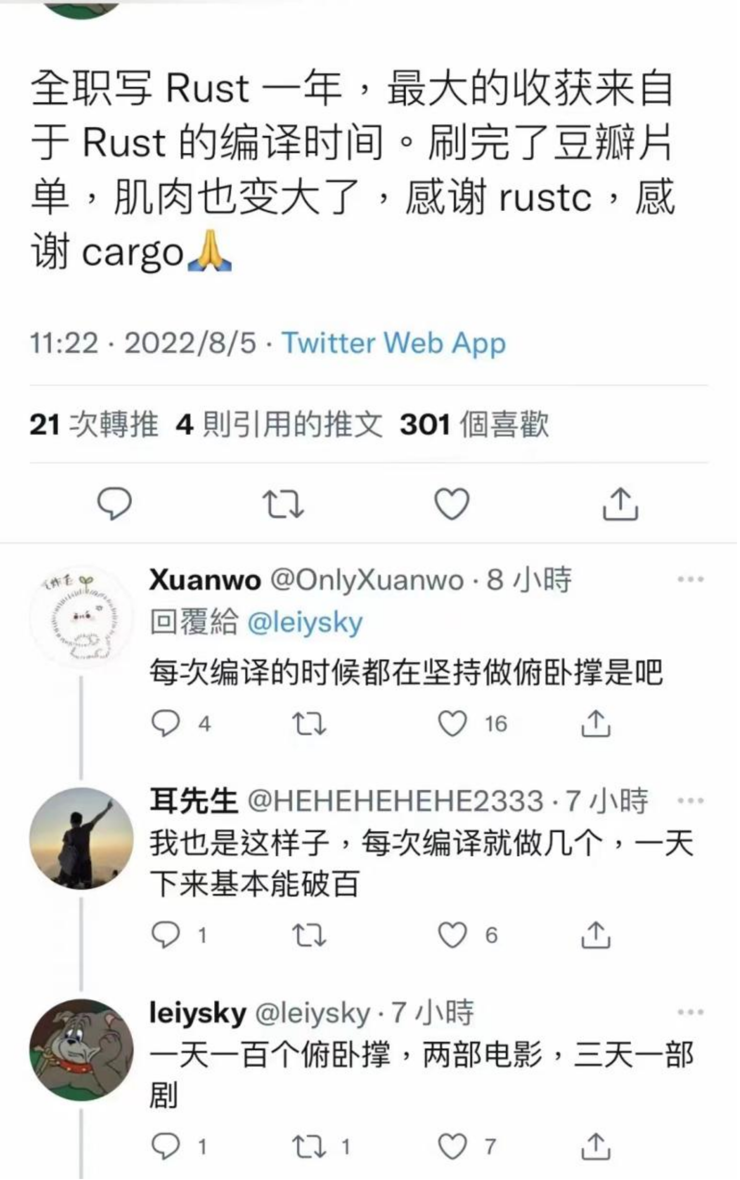
Rust OS never gets rusty.

Tacos or Pintos



- Rust compiler:
  - Compiles slowly.
  - Some rules are too strict.
    - `unsafe` and raw pointers may become your friends
- Rust is hard to learn.
- **Tacos** is in developing
  - The document is not as elaborate as Pintos
  - Potential bugs

Vote against Tacos



全职写 Rust 一年，最大的收获来自于 Rust 的编译时间。刷完了豆瓣片单，肌肉也变大了，感谢 rustc，感谢 cargo🙏

11:22 · 2022/8/5 · Twitter Web App

21 次轉推 **4** 則引用的推文 **301** 個喜歡

Xuanwo @OnlyXuanwo · 8 小時
回覆給 @leiysky
每次编译的时候都在坚持做俯卧撑是吧
4 16

耳先生 @HEHEHEHEHE2333 · 7 小時
我也是这样子，每次编译就做几个，一天下来基本能破百
1 6

leiysky @leiysky · 7 小時
一天一百个俯卧撑，两部电影，三天一部剧
1 1 7

Setting up environment

- Following the document.

- The best way is using docker and dev container plugin.

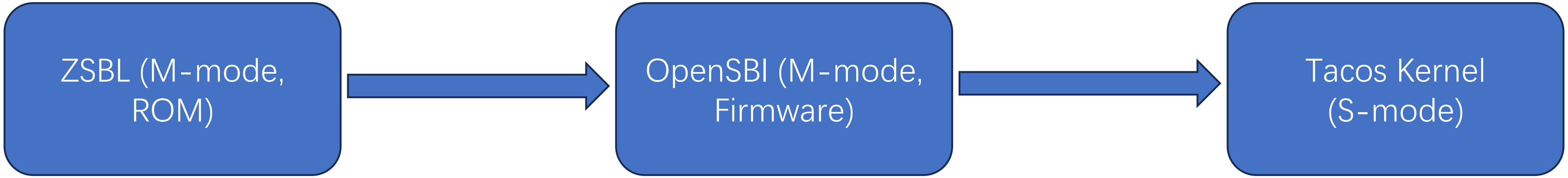
- If you’re computer is not x86, like Apple M-series chips, you may find Tacos is very slow. In this case, using clab will be a good idea.

Setting up environment

- You may meet some problem pulling docker image due to some reasons.
  - If you are using clab,
    - Install docker on your PC and clab. (STFW)
    - Pull the docker image (chromi2eugen/tacos:latest) on your PC first.
    - Use ``` `docker save -o tacos.tar` ``` to save the image
    - Use scp or other methods to upload `tacos.tar` to your server
    - Use ``` `docker load -I tacos.tar` ``` to load the image to your docker in clab.

Lab0 Intro

- GDB
- Tacos booting process



ZSBL (M-mode, ROM)

OpenSBI (M-mode, Firmware)

Tacos Kernel (S-mode)

Other suggestions

- When you use git, you can also create a **private repository** on github.
  - If something goes wrong on your computer or clab, you can have a backup.
  - Do not use rm –rf

- AI may be your friend.
  - But don't rely on copilot to write your code. They are almost always wrong when implementing **Tacos**.

- Read code in **Tacos**.

RUST OS NEVER GETS RUSTY.