

- ASÍNCIONISMO / CONCURRENCIA
- bib. standar Async / await  
Future trait
- async\_std crate  
futures crate

Thread  
hilo real  
sin gestión



async  
Await ~ esperar  
hilo virtual

gestión.

event loop



intuición

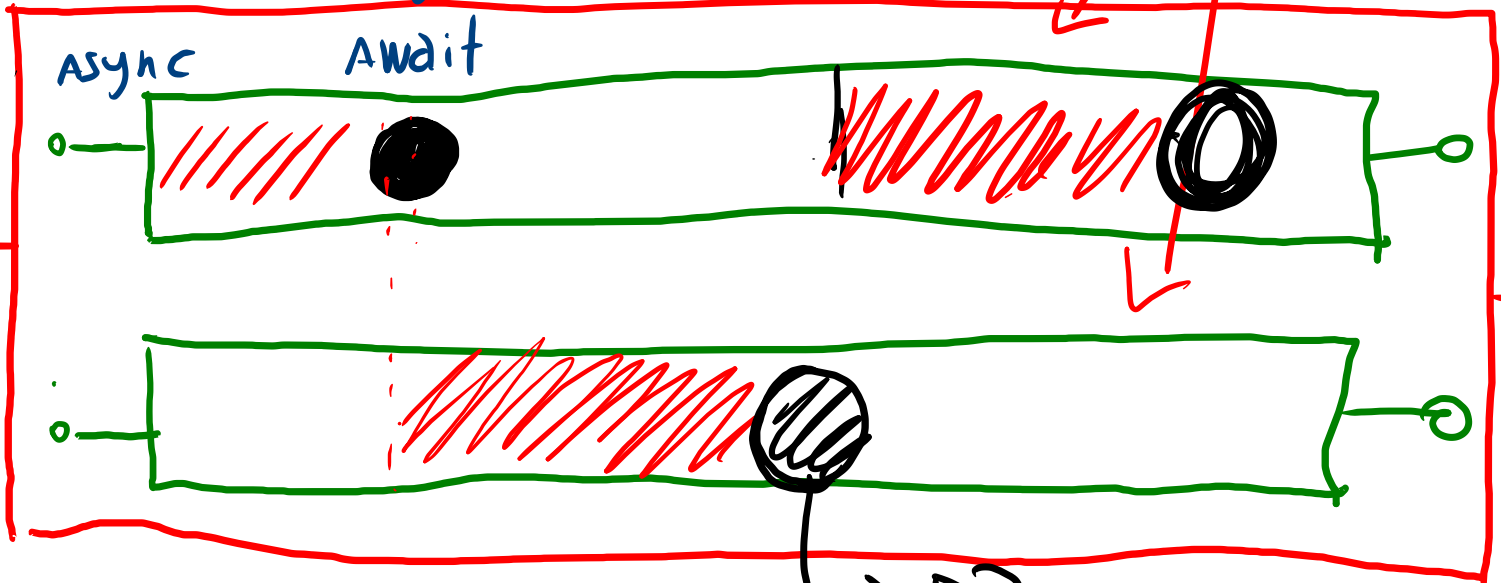
1 rewrso comp. runtime.

REST API



Async

Await



DB

JS  $\mapsto$  Promise  $\rightarrow$  async/await

Python  $\mapsto$  coroutines  
(function)  $\mapsto$  tasks  
(Futures)

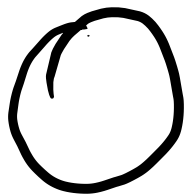
continuation  
Rust  $\mapsto$  async fn  $\mapsto$  future  
(task)  $\rightarrow$  async/await  
Future

# Rust con async :

1º def funciones + trait

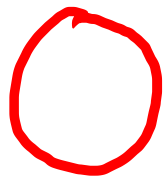
2º trait Future

3º lógica operación

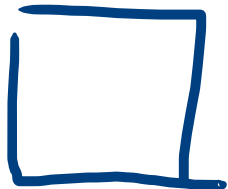
  
fn



  
run







future





async  
fn

run.  


Revision web /

Crate "async\_std"

↳ Rust + async

File  
I/O  
:  
:  
:  
}

ASync



• await

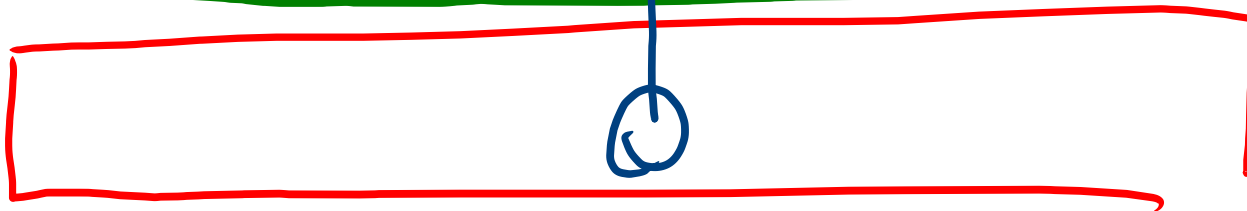


tonic

logger (logging) / tonic . . . /

Async

+ Await



Rust