



Rust For Serious Developers

Armin @mitsuhiko Ronacher

Hi, I'm Armin
... and I do Open Source

Flask
Jinja
Sentry
...



Flask

web development,
one drop at a time



SENTRY

Sentry Loves Rust

Symbolication



84.114.93.8



iPod Touch 6G
arm64



iOS
Version: 10.1.1

TAGS

device iPod Touch 6G device.family iPod level fatal os iOS 10.1.1 os.name iOS release 1.0

Thread #0 (most recent call first)

Symbolicated Unsymbolicated App Only Full Raw

id 0 name n/a was active no crashed yes

EXC_BREAKPOINT

fatal error: Index out of range
Index out of range

mach exception EXC_BREAKPOINT

SentryDemo	0x00071c7c	main (AppDelegate.swift:13)
Called from: UIKit UIApplicationMain		
SentryDemo	0x00070a4c	@objc SentryDemo.ViewController.CrashIt (protocol<>) -> () (ViewController.swift)
SentryDemo	0x00070940	SentryDemo.ViewController.CrashIt (protocol<>) -> () (ViewController.swift:25)

Source Map Expansion

EXCEPTION (most recent call first)

Original

Minified

App Only

Full

Raw

TypeError

Cannot read property 'id' of undefined

./app/views/projectDetails.jsx in **Constructor.onProjectChange** at line 90:59 [?](#) [-]

```
85.   onProjectChange(projectIds) {
86.     if (!this.state.project) return;
87.     if (!projectIds.has(this.state.project.id)) return;
88.
89.     this.setState({
90.       project: {...ProjectStore.getById(this.state.Project.id)}
91.     });
92.   },
93.
94.   identifyProject() {
95.     let {params} = this.props;
```

Called from: ~/reflux-core/lib/PublisherMethods.js in **EventEmitter.eventHandler** [?](#)

./app/stores/projectStore.jsx in **Store.loadInitialData** at line 25:9 [?](#) [+]

./app/views/organizationDetails.jsx in **Request.success** at line 100:21 [?](#) [+]

./app/api.jsx in **Object.success** at line 61:20 [?](#) [+]

Called from: ~/jquery/dist/jquery.js in **fire** [?](#)

Command Line Tool



3. mitsuhiro@herzog: ~/Development/sentry-cli (zsh) — 87X29

mitsuhiko at herzog in ~/Development/sentry-cli on git:master rust 1.15.0

\$ sentry-cli info

Sentry Server: http://localhost:8000/

Organization: default

Project: earth

Authentication Info:

Method: Auth Token

User: armin.ronacher@active-4.com (id=1)

Scopes:

- * project:read
- * project:write
- * project:delete
- * project:releases
- * team:read
- * team:write
- * team:delete
- * event:read
- * event:write
- * event:delete
- * org:read
- * org:write
- * org:delete
- * member:read
- * member:write
- * member:delete

mitsuhiko at herzog in ~/Development/sentry-cli on git:master rust 1.15.0

\$

Hello Rust

Ocaml	ADT, Pattern Matching
C++	RAI, Smart Pointers
Haskell	Typeclasses
Erlang	Message Passing, Failures
Scheme	Hygienic Macros
C#	Attributes

```
fn main() {  
    println!("Hello, World!");  
}
```

The Toolchain

is so good




```
curl https://sh.rustup.rs -sSf | sh
```

or on windows

```
https://win.rustup.rs
```

rustup

the rust toolchain manager

rustc

the rust compiler

cargo

the rust package manager

rustdoc

the rust documentation builder

Quickstart

```
$ cargo new --bin hello-rust
  Created binary (application) `hello-rust` project

$ cd hello-rust

$ cargo run
  Compiling hello-rust v0.1.0 (file:///private/tmp/hello-rust)
  Finished debug [unoptimized + debuginfo] target(s)
  Running `target/debug/hello-rust`
Hello, world!
```

Release Channels

stable

new release every 6 weeks

beta

upcoming stable release

nightly

includes unstable features

```
$ rustup override set nightly
```

```
info: using existing install for 'nightly-x86_64-apple-darwin'
```

```
info: override toolchain for '/private/tmp/hello-rust' set to  
'nightly-x86_64-apple-darwin'
```

```
    nightly-x86_64-apple-darwin unchanged - rustc 1.17.0-nightly  
(0648517fa 2017-02-27)
```

The Ecosystem

crates.io

repository of reusable rust crates

docs.rs

documentation for all crates

The Features


```
enum JsonValue {  
    Null,  
    Bool(bool),  
    Number(f64),  
    Str(String),  
    List(Box<Vec<JsonValue>>),  
    Object(Box<HashMap<String, JsonValue>>),  
}
```

```
fn main() {  
    let value = JsonValue::Str("Hello World!".to_string());  
    match value {  
        JsonValue::Str(ref string) => {  
            println!("JSON value as a string: {}", string);  
        },  
        _ => {  
            println!("JSON value was something else");  
        }  
    }  
}
```

explicit error propagation



```
use std::{fs, env, io};
```

```
fn list_current_dir() -> Result<(), io::Error> {  
    let here = env::current_dir()?;  
    for dent_rv in fs::read_dir(here)? {  
        let dent = dent_rv?;  
        let md = dent.metadata()?;  
        println!("{: <60}{: <12}{: }",  
                dent.path().display(),  
                md.len(),  
                if md.is_file() { "file" } else { "dir" });  
    }  
    Ok(())  
}
```

```
fn main() {  
    list_current_dir().unwrap();  
}
```

strong types and inference



```
use std::{fs, env, io};
use std::collections::HashMap;
```

Result Type

```
fn count_extensions() -> Result<(), io::Error> {
    let mut counts = HashMap::new();
    for dent_rv in fs::read_dir(env::current_dir().unwrap()) {
        if let Some(ext) = dent_rv?.path().extension() {
            *counts.entry(ext.to_string_lossy().into_owned()).or_insert(0) += 1;
        }
    }

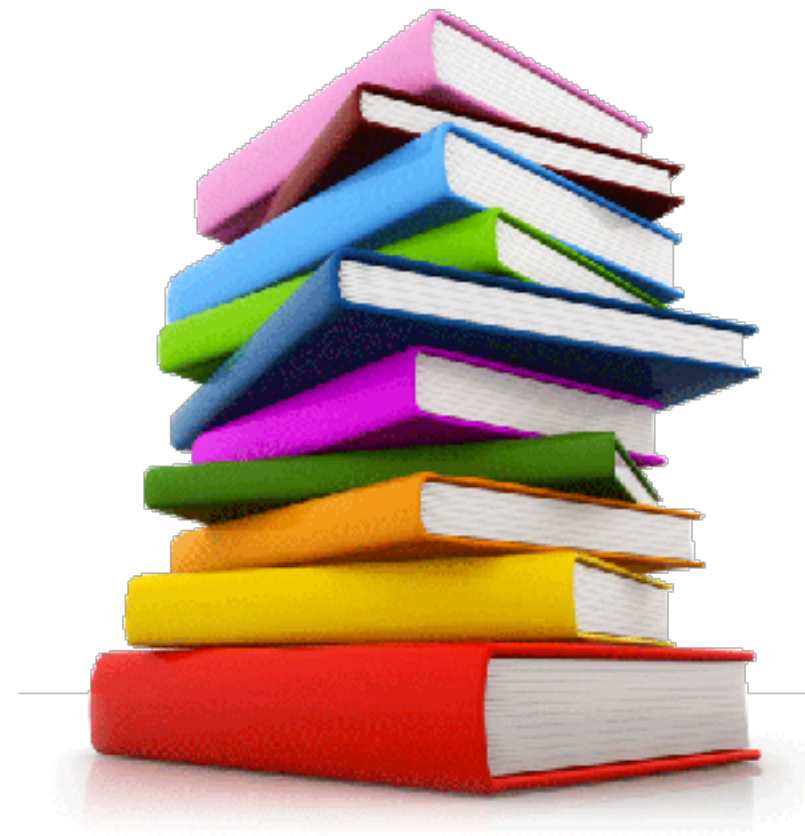
    let mut items = counts.into_iter().collect::<Vec<_>>();
    items.sort_by_key(|&(_, value)| -value);
    for (ext, count) in items {
        println!("{: >5} {}", count, ext);
    }

    Ok(())
}
```

OS Str -> String

Hash Table Entry

Vector of Tuples



data vs behavior


```
struct User {
    id: u64,
    username: String,
}

impl User {
    fn new(id: u64, username: &str) -> User {
        User {
            id: id,
            username: username.to_string(),
        }
    }

    fn username(&self) -> &str {
        &self.username
    }
}
```



it's all about traits



```
use std::fmt;
```

```
impl fmt::Debug for User {
```

```
    fn fmt(&self, f: &mut fmt::Formatter) -> fmt::Result {  
        write!(f, "<User {} (id={})>", &self.username, self.id)
```

```
    }
```

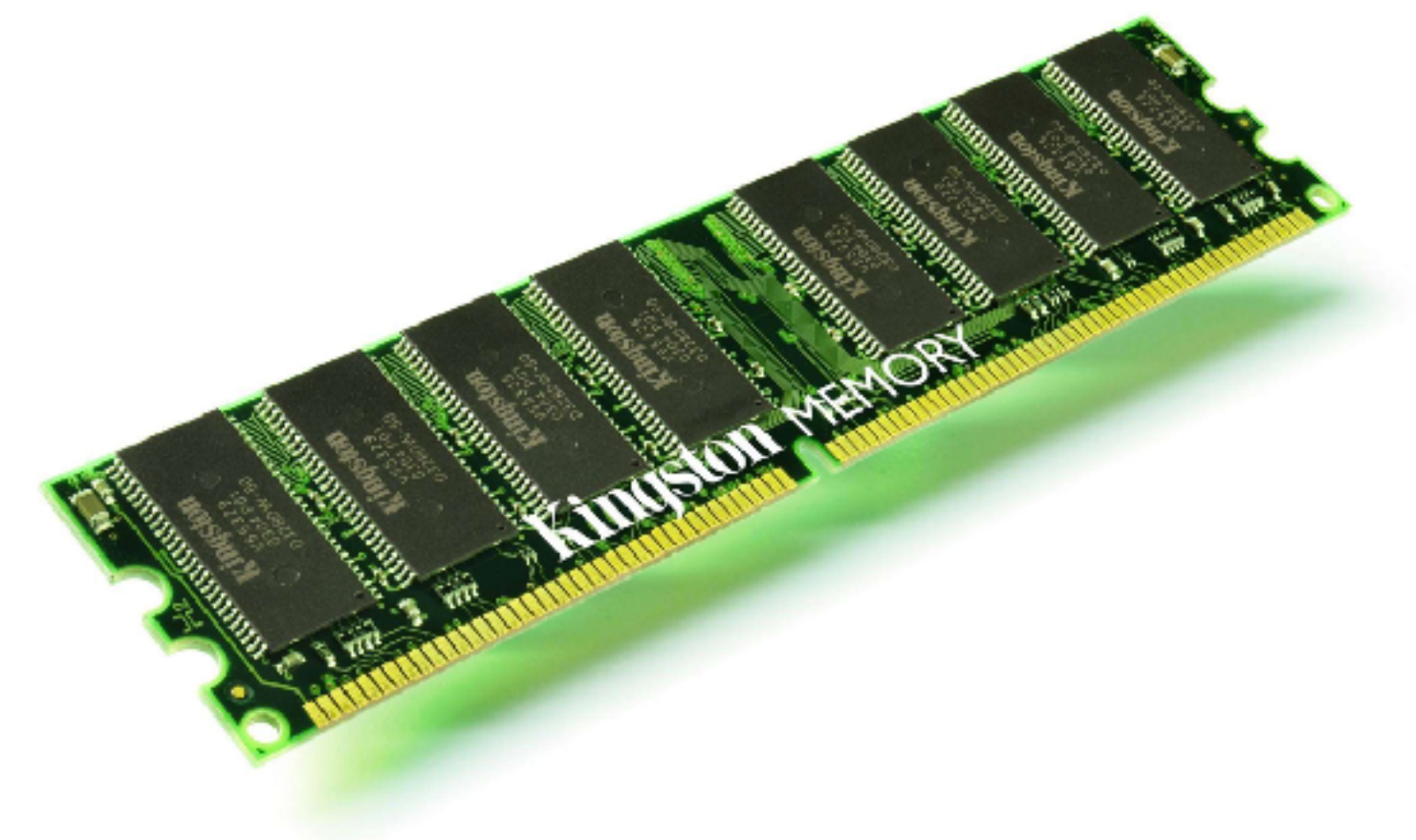
```
}
```

compiler plugins



```
#[derive(Debug, Clone, Serialize, Deserialize)]
struct User {
    id: u64,
    username: String,
}
```

memory safety







STAGE FRIGHT



so how does rust do it?

lifetimes	Reason about Memory
move semantics	Track where stuff goes
ADTs	No Null-Pointers
smart pointers	Runtime Lifetimes
bounds checking	Panic on OOB
explicit unsafe	The Escape Hatch

```
struct SdkInfo {  
    name: String,  
    version: String,  
}
```

```
struct SdkInfo<'a> {  
    name: &'a str,  
    version: &'a str,  
}
```

```
struct SdkInfo<'a> {  
    name: Cow<'a, str>,  
    version: Cow<'a, str>,  
}
```

```

fn main() {
    let items = vec![1, 2, 3];
    let sitems = items.into_iter().map(|x| x.to_string()).collect::<Vec<_>>();
    println!("Converted {} items to strings", items.len());
}

```

```

error[E0382]: use of moved value: `items`

```

```

--> fail.rs:4:46

```

```

3 |     let sitems = items.into_iter().map(|x| x.to_string()).collect::<Vec<_>>();

```

```

      ----- value moved here

```

```

4 |     println!("Converted {} items to strings", items.len());

```

```

      ^^^^^ value used here after move

```

```

= note: move occurs because `items` has type `std::vec::Vec<i32>`,
       which does not implement the `Copy` trait

```

```

error: aborting due to previous error

```

free parallelism



```
extern crate rayon;
```

Values that order & can be sent to threads

```
fn quick_sort<T: PartialOrd + Send>(v: &mut [T]) {  
    if v.len() > 1 {  
        let mid = partition(v);  
        let (low, high) = v.split_at_mut(mid);  
        rayon::join(|| quick_sort(low), || quick_sort(high));  
    }  
}
```

split mutable slice into two

Spawn two threads and join

```
fn partition<T: PartialOrd + Send>(xs: &mut [T]) -> usize {  
    let pivot = xs.len() - 1;  
    let mut i = 0;  
    for j in 0..pivot {  
        if xs[j] <= xs[pivot] { xs.swap(i, j); i += 1; }  
    }  
    xs.swap(i, pivot);  
    i  
}
```

```
struct Server {  
    ctx: Arc<Context>,  
}
```

```
impl Server {  
    pub fn run(&self) -> Result<(), > {  
        let ctx = self.ctx.clone();  
        Server::http("127.0.0.1:3000")?  
            .handle(move |req: Request, resp: Response|  
                {  
                    let local_ctx = ctx.clone();  
                    ...  
                })  
    }  
}
```

increments refcount and returns new Arc



Should you use it?

maybe!

Python Modules

Redis/Nginx Modules

Command Line Tools

Native Libraries

One-Trick-Pony API Servers

Our Experience

code is very readable

compile times suck

crate ecosystem is great

highly maintainable

Q&A