What?
What?
Why?
What?
Why?
Where?
What?  
Why?  
Where?  
Who?
What?
Why?
Where?
Who?
When?
Jakub Beránek

github.com/kobzol
jakub@berankovi.net

- PhD, teaching, research @ IT4Innovations (Czech Republic supercomputing center)
Jakub Beránek

github.com/kobzol
@ jakub@berankovi.net

- PhD, teaching, research @ IT4Innovations (Czech Republic supercomputing center)
- Rust Project open-source contributor
Jakub Beránek

github.com/kobzol

jakub@berankovi.net

- PhD, teaching, research @ IT4Innovations (CZ)
- Rust Project open-source contributor
- Rust Infrastructure team (member)
Jakub Beránek

github.com/kobzol
@ jakub@berankovi.net

- PhD, teaching, research @ IT4Innovations (Cz)
- Rust Project open-source contributor
- Rust Infrastructure team (member)
- Rust Survey team (co-lead)
(disclaimer)

All opinions (and possible errors) are my own :-)

Rust

A language empowering everyone to build reliable and efficient software.
Rust timeline

2006

moz://a
Rust timeline

2006

Rust Project

2015

mozilla

Rust 1.0
Rust Foundation Platinum members

AWS

Google

HUawei

Meta

Microsoft
Introducing RustRover
A dedicated IDE for Rust Developers

Join the Preview
Why Rust?

Performance  Reliability  Productivity
<table>
<thead>
<tr>
<th>Why Rust?</th>
<th>Performance</th>
<th>Reliability</th>
<th>Productivity</th>
</tr>
</thead>
</table>

Rust programs are efficient
Rust programs are efficient **by default**.
Rust programs are efficient by default.

"Time-to-performance"
Rust is becoming a first class language in a variety of domains. At Discord, we’ve seen success with Rust on the client side and server side. For example, we use it on the client side for our video encoding pipeline for Go Live and on the server side for Elixir NIFs.
Even with **just basic optimization**, Rust was able to **outperform the hyper hand-tuned** Go version...
"Even with just basic optimization, Rust was able to outperform the hyper hand-tuned Go version... ...we were able to beat Go on every single performance metric.

Discord"
More efficient => cheaper to run
Optimizing 700 CPUs Away With Rust

Alan Ning · Follow
Published in Tenable TechBlog · 3 min read · May 6, 2021
Performance

CPU

- JS: 800 MHz
- Rust: 200 MHz

Per pod, average CPU reduced from 800m to 200m core

Memory

- JS: 70 MB
- Rust: 5 MB

Per pod, average memory reduced from 70MB to 5MB.
Rewriting the heart of our sync engine

// By Sujay Jayakar • Mar 09, 2020
“...betting on Rust was one of the best decisions we made. More than performance, its ergonomics and focus on correctness has helped us tame sync’s complexity.”

Dropbox
Why Rust?

Performance  Reliability  Productivity
Rust makes it easier to write correct software
Rust makes it easier to write **correct software** by adding **guardrails**
Confidence
“The language's expressiveness allows our developers to encode constraints that catch errors at compile time rather than in GitHub issues.”

Vercel
Which of the following statements about Rust do you feel are true?

"Rust is risky to use in production"

1.6%

Source: Rust Annual 2023 survey
Rust is memory safe by default
Chrome: 70% of all security bugs are memory safety issues

Google software engineers are looking into ways of eliminating memory management-related bugs from Chrome.
Chrome: 70% of all security bugs are memory safety issues

Google software engineers are looking into ways of eliminating memory management-related bugs from Chrome.

Microsoft: 70 percent of all security bugs are memory safety issues

Percentage of memory safety issues has been hovering at 70 percent for the past 12 years.
Chrome: 70% of all security bugs are memory safety issues

Google software engineers are looking into ways of eliminating memory management-related bugs from Chrome.

Microsoft: 70 percent of all security bugs are memory safety issues

Percentage of memory safety issues has been hovering at 70 percent for the past 12 years.

Figure 2: Memory safety bugs contribution to Android vulnerabilities
Press Release: Future Software Should Be Memory Safe
Memory Safe Languages in Android 13

December 1, 2022

New Native Code

Android release

%
A Memory Safe Implementation of the Network Time Protocol

Folkert de Vries
Oct 11, 2022

Bringing Memory Safety to sudo and su

Josh Aas
Apr 26, 2023
Stability
Stability

Rust 1.0 released in 2015 (9 years ago)
Stability
Rust 1.0 released in 2015 (9 years ago)

...without stagnation
Rust 1.78 released last week
Do you agree with the following statements on Rust stability?

"Upgrading to a new stable compiler version requires either no changes or extremely small & easy changes to my code"

97.72%

Source: Rust Annual 2023 survey
Why Rust?

Performance  Reliability  Productivity
Unified tooling
Unified tooling

• Building
Unified tooling

- Building
- Testing
Unified tooling

- Building
- Testing
- Dependency management
Unified tooling

- Building
- Testing
- Dependency management
- Documentation
Unified tooling

- Building
- Testing
- Dependency management
- Documentation
- Deployment
Unified tooling

- Building
- Testing
- Dependency management
- Documentation
- Deployment
- ...

Productivity
Unified tooling

- Building
- Testing
- Dependency management
- Documentation
- Deployment
- ...

All with a **single tool** (Cargo)
Package ecosystem

144,373
Crate in stock
Rust has absolutely **stunning dependency management**. NPM
Rust is great for efficient, production-ready software
Rust is great for efficient, production-ready software

Not all software is like that
Which of the following statements are reasons why you use Rust at work?

"We find it easy to prototype with"

14%

Source: Rust Annual 2023 survey
Longer initial ramp-up time
Trade-offs

Longer initial ramp-up time

Less bugs in production
Which of the following statements apply to your experience using Rust at work?

"Overall, adopting Rust has slowed down our team"

8.2%

Source: Rust Annual 2023 survey
Onboarding
Confidence Contributing to Rust Codebase

- Still not productive: 8.1%
- More than 4 months: 8.5%
- 3 - 4 months: 14.7%
- 2 - 3 weeks: 25.6%
- 1 - 2 months: 37.6%
- 1 week or less: 5.4%

More than 2/3 of respondents are confident contributing to Rust codebase within 2 months or less when learning Rust.

Q14 - While learning Rust, how long did it take you to become sufficiently productive to contribute to your Rust codebase without worrying about the language? (choose one)

Source: Lars Bergstrom - Beyond Safety and Speed: How Rust Fuels Team Productivity
Ecosystem maturity
Ecosystem maturity

Many libraries, less frameworks
Rust has a Lego-like package ecosystem.

Luca Palmieri
2023 Stack Overflow Survey: Rust is the most admired programming language, making it the most loved language for 8 years in a row

survey.stackoverflow.co/2023/
Where is Rust being used today?
Where do you live?
(total responses = 9572)

Source: Rust Annual 2023 survey
In what technology domain(s) is Rust used at your organisation?
(total responses = 4139, multiple answers)

Not all answers are displayed

Source: Rust Annual 2023 survey
In what technology domain(s) is Rust used at your organisation?
(total responses = 4139, multiple answers)

Percent out of all responses (%)

- 52.7%

Source: Rust Annual 2023 survey

Not all answers are displayed
GitHub built a new search engine for code 'from scratch' in Rust

GitHub built a new code-focused search engine in Rust because popular text search engines couldn't scale enough.

Written by Liam Tung, Contributing Writer
Feb. 9, 2023 at 3:24 a.m. PT
In what technology domain(s) is Rust used at your organisation?
(total responses = 4139, multiple answers)

Source: Rust Annual 2023 survey
In what technology domain(s) is Rust used at your organisation?
(total responses = 4139, multiple answers)

Serverless Rust with Cloudflare Workers
10/16/2018

Steven Pack

Not all answers are displayed

Source: Rust Annual 2023 survey
In what technology domain(s) is Rust used at your organisation?
(totals responses = 4139, multiple answers)

- Server-side or "backend" applications: 52.7%
- Cloud computing applications: 27.2%
- Operating systems: 17.5%
- Embedded devices (bare metal): 12.6%

Source: Rust Annual 2023 survey
Ferrocene is an open source qualified Rust compiler toolchain. With this, Ferrous Systems invested its decades of Rust experience to make Rust a first-class language for mission-critical and functional safety systems.

Infineon expands Rust ecosystem for AURIX™ with HighTec’s ISO 26262 ASIL D qualified Rust compiler and other solutions

Source: Rust Annual 2023 survey
In what technology domain(s) is Rust used at your organisation?
(total responses = 4139, multiple answers)

- Server-side applications or "backend": 52.7%
- Cloud computing: 27.2%
- Operating systems: 17.5%
- Devices (with metal): 12.6%
- Application frontend: 12.0%
- Scientific computing: 10.4%
- IoT (Internet of Things): 9.5%

Not all answers are displayed

Source: Rust Annual 2023 survey
In what technology domain(s) is Rust used at your organisation?
(totals responses = 4139, multiple answers)

Code at the speed of thought

Zed is a high-performance, multiplayer code editor from the creators of Atom and Tree-sitter. It's also open source.

Source: Rust Annual 2023 survey
Microsoft posts ‘early stages’ code for developing Windows drivers in Rust

By Tim Anderson - September 25, 2023

If you’re on the Win11 Insider ring, you’re getting the first taste of Rust in the Windows kernel!

_directory of C:\Windows\System32_

```
C:\Windows\System32>dir win32k*
Volume in drive C has no label.
Volume Serial Number is E60B-9A9E

Directory of C:\Windows\System32

04/15/2023  09:50 PM          708,608  win32k.sys
04/15/2023  09:49 PM          110,592  win32kbase.sys
04/15/2023  09:50 PM          4,194,304  win32kfull.sys
04/15/2023  09:49 PM          40,960  win32kfull.rs.sys
04/15/2023  09:49 PM          69,632  win32k.sys
04/15/2023  09:49 PM          98,304  win32ksgd.sys

7 File(s)          8,646,656 bytes
0 Dir(s)         116,366,049,280 bytes free
```

/rs = Rust!
Rust Programming Language To Land in Linux Kernel 6.1

By Ian Evenden published October 06, 2022

Linux will support the Rust programming language in its kernel from version 6.1.
Rust Programming Language To Land in Linux Kernel

By Ian Evenden published

Linux will support the Rust language in its kernel from version 6.1.

“…on the whole, I don't hate it.”

Linus Torvalds
Make Ship Happen

Turbo is an incremental bundler and build system optimized for JavaScript and TypeScript, written in Rust.

TURBOREPO
High-performance build system for JavaScript and TypeScript codebases.

TURBOPACK
Introducing the Rust-powered successor to Webpack.
RUFF

Lint at lightspeed

An extremely fast Python linter, written in Rust.
Rust anywhere
Rust anywhere

- Linux, Windows, macOS, …
Rust anywhere

- Linux, Windows, macOS, ...
- Cloud, backend, frontend, desktop, mobile, ...
Rust anywhere

- Linux, Windows, macOS, …
- Cloud, backend, frontend, desktop, mobile, …
- From embedded devices to supercomputers
Rust anywhere

- Linux, Windows, macOS, ...
- Cloud, backend, frontend, desktop, mobile, ...
- From embedded devices to supercomputers
- Code reuse
Rust anywhere

• Linux, Windows, macOS, …
• Cloud, backend, frontend, desktop, mobile, …
• From embedded devices to supercomputers
• Code reuse
• Interoperable with C, C++, Python, WebAssembly, …
Adoption of Rust
Which programming, scripting, and markup languages have you used in the last 12 months?

Source: The State of Developer Ecosystem in 2023 (JetBrains)
Package ecosystem size

Source: crates.io.pypistats.org.all-the-package-names
Package ecosystem size

- Rust
  - ~140k

Source: crates.io.pypistats.org.all-the-package-names
Package ecosystem size

- Rust  ~140k
- Python ~520k

Source: crates.io, pypistats.org, all-the-package-names
Package ecosystem size

- Rust  ~140k
- Python ~520k
- JavaScript ~2,800k

Source: crates.io, pypistats.org, all-the-package-names
Number of Rust packages

Source: crates.io
Rust package downloads

Rust is no longer **niche**...

Source: crates.io
...but it is not yet fully **mainstream**.
How Rust is Tilde’s Competitive Advantage

Source: Rust Case Study: How Rust is Tilde's Competitive Advantage
Thank you for your attention!

Rust Annual Survey 2023:

Slides were created using github.com/spirali/elsie