

# 2021 Rust community annual survey

This document contains a summary of results from the 2021 annual community survey. There have been a few questions elided (see section at the end of the document). Data has been minimally processed. We expect to publish shorter reports with select data processed for a specific audience in the near future. If there's anything you're particularly interested in please get in touch with the survey working group.

Where there are small differences in the numbers between this report and previously reported numbers (e.g., in the blog post), that is likely due to how partially complete surveys were counted, or very small discrepancies in the data depending on how it is exported from the survey software.

The survey was open for responses from 8<sup>th</sup> December to 22<sup>nd</sup> December 2021. It was available in ten languages.

We'd like to thank everyone who helped with translation of the survey, and everyone who participated.

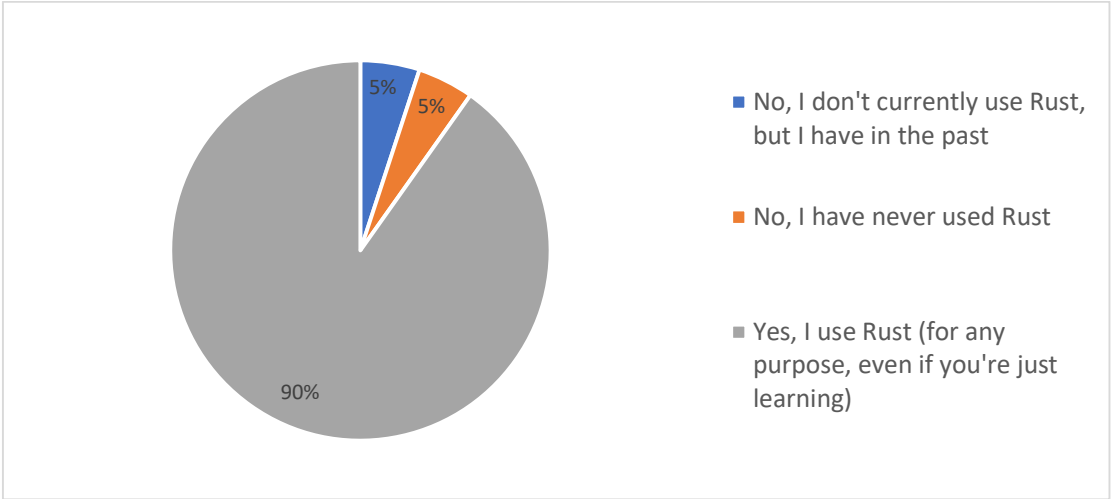
## Contents

Questions we asked everyone .....	1
Location and language .....	2
Programming experience .....	4
Work and study .....	7
Questions for respondents who write or design software .....	9
Questions for Rust users .....	11
Questions for Rust users who use Rust at work .....	56
Questions for Rust users who are taking a course or training which uses/teaches Rust .....	60
Question for former Rust users .....	62
Question for non-Rust users .....	63
Metadata .....	64
Elided questions .....	65

## Questions we asked everyone

Do you use Rust?

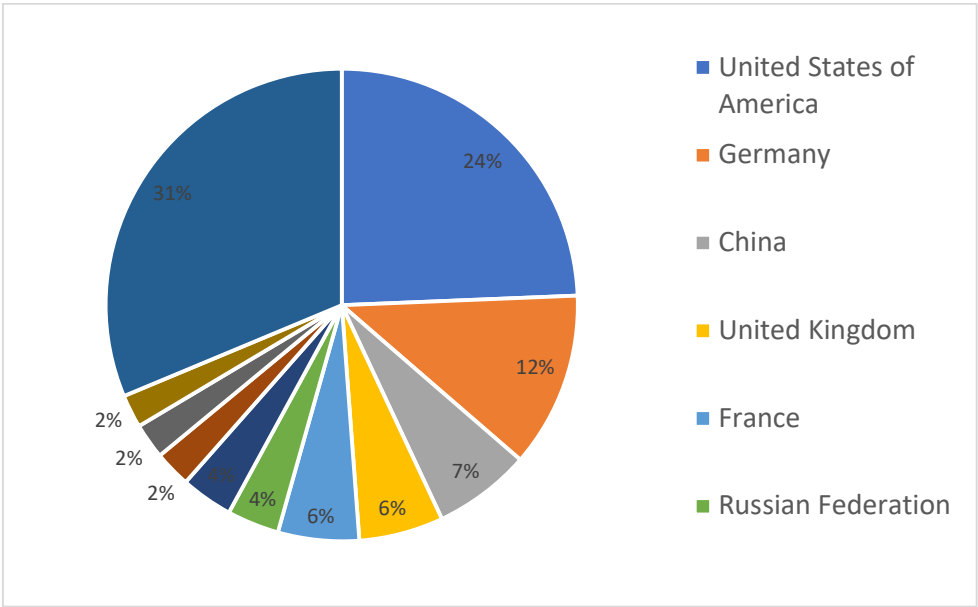
	Frequency	Percent
No, I don't currently use Rust, but I have in the past	469	5%
No, I have never used Rust	444	5%
Yes, I use Rust (for any purpose, even if you're just learning)	8395	90%



Location and language

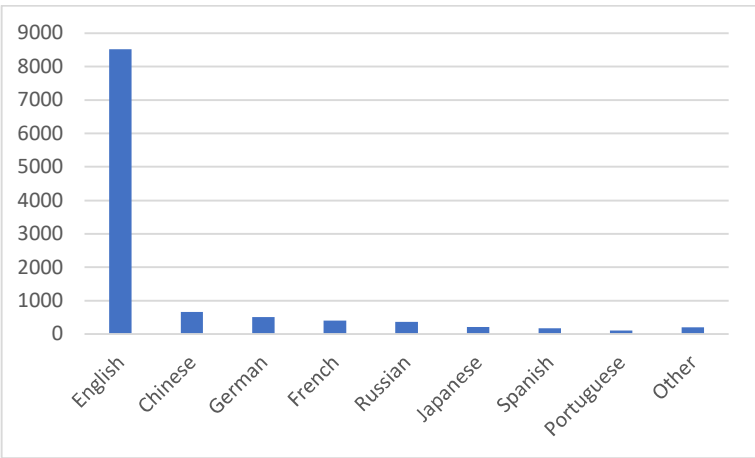
Where do you live?

United States of America	2236	24.4%
Germany	1101	12.0%
China	612	6.7%
United Kingdom	529	5.8%
France	510	5.6%
Russian Federation	328	3.6%
Canada	327	3.6%
Japan	229	2.5%
Netherlands	223	2.4%
Sweden	210	2.3%
Other Locale	2871	31.3%



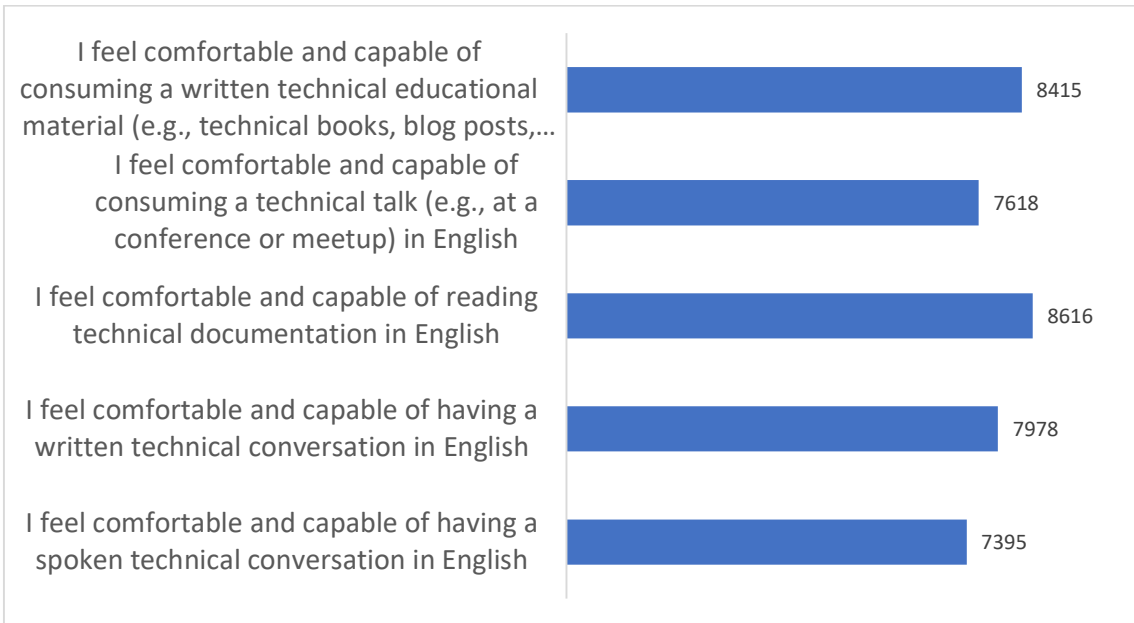
What is/are your preferred language(s) for technical communication?

English	8519
Chinese	667
German	511
French	408
Russian	369
Japanese	216
Spanish	176
Portuguese	112
Other	203



In what ways are you comfortable communicating about technical topics in English?

I feel comfortable and capable of having a spoken technical conversation in English	7395	79.45%
I feel comfortable and capable of having a written technical conversation in English	7978	85.71%
I feel comfortable and capable of reading technical documentation in English	8616	92.57%
I feel comfortable and capable of consuming a technical talk (e.g., at a conference or meetup) in English	7618	81.84%
I feel comfortable and capable of consuming a written technical educational material (e.g., technical books, blog posts, etc.) in English	8415	90.41%

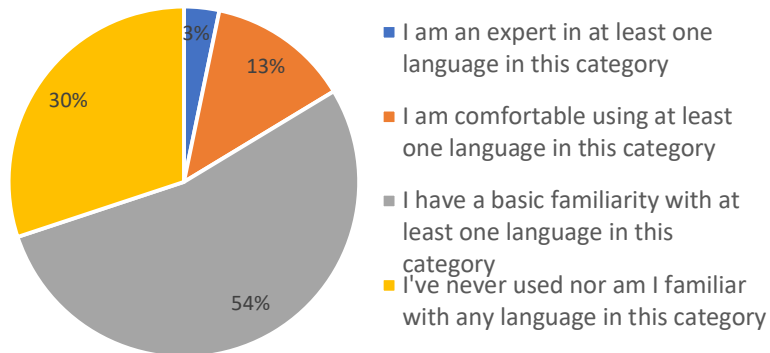


## Programming experience

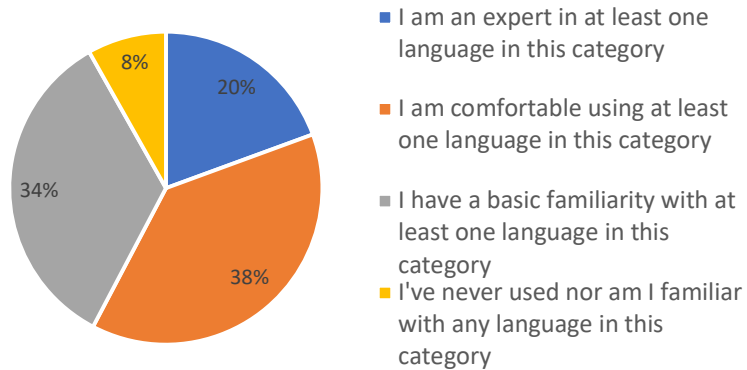
Excluding Rust, what is your experience with other kinds of programming languages?

All respondents were asked this question, but these results are for Rust users only.

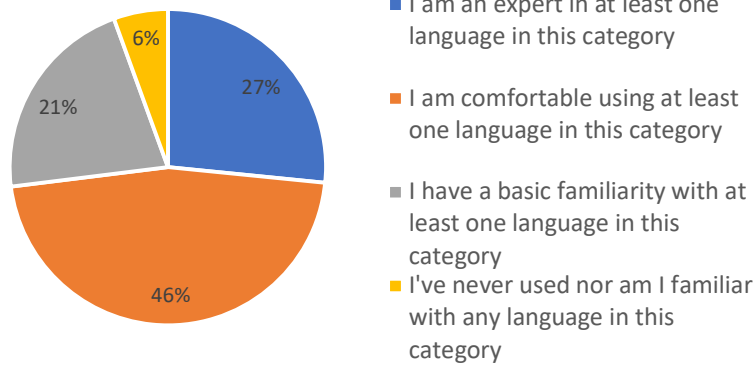
### Assembly language (of any variety)



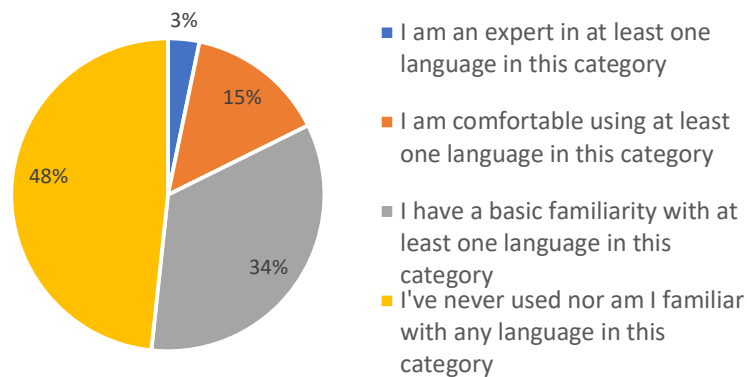
### Languages with manual memory management (e.g., C, C++, Objective-C without ARC)



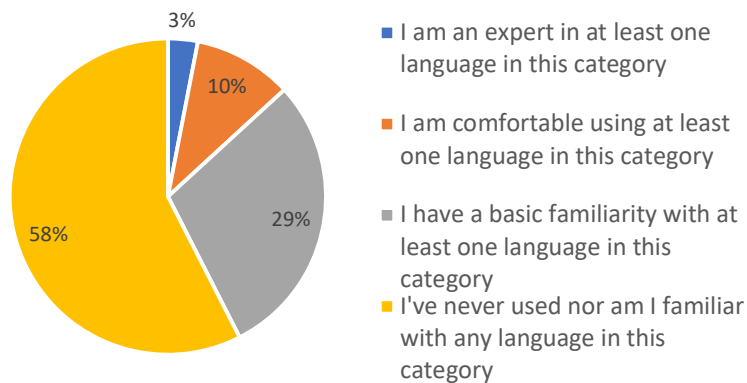
### Statically typed object oriented languages with garbage collection (e.g., Java, C#, Go)



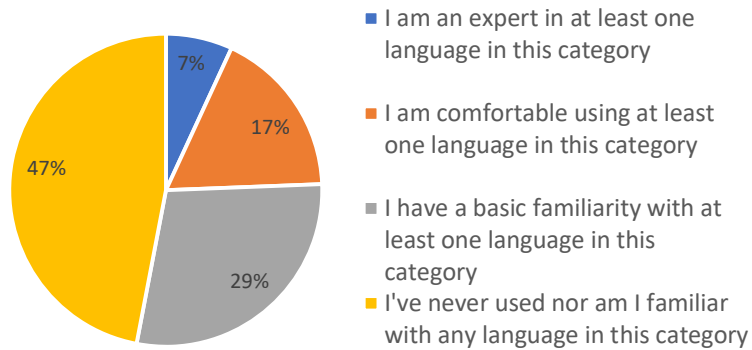
### Statically typed functional programming languages (e.g., Haskell, ML)



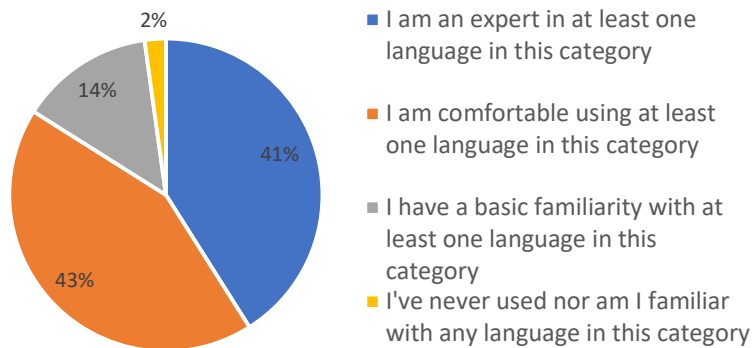
### Dynamically typed functional programming languages (e.g., Lisp, Clojure, Elixir)



### Statically typed languages with newer expressive type systems (e.g., Swift, Kotlin, Scala)

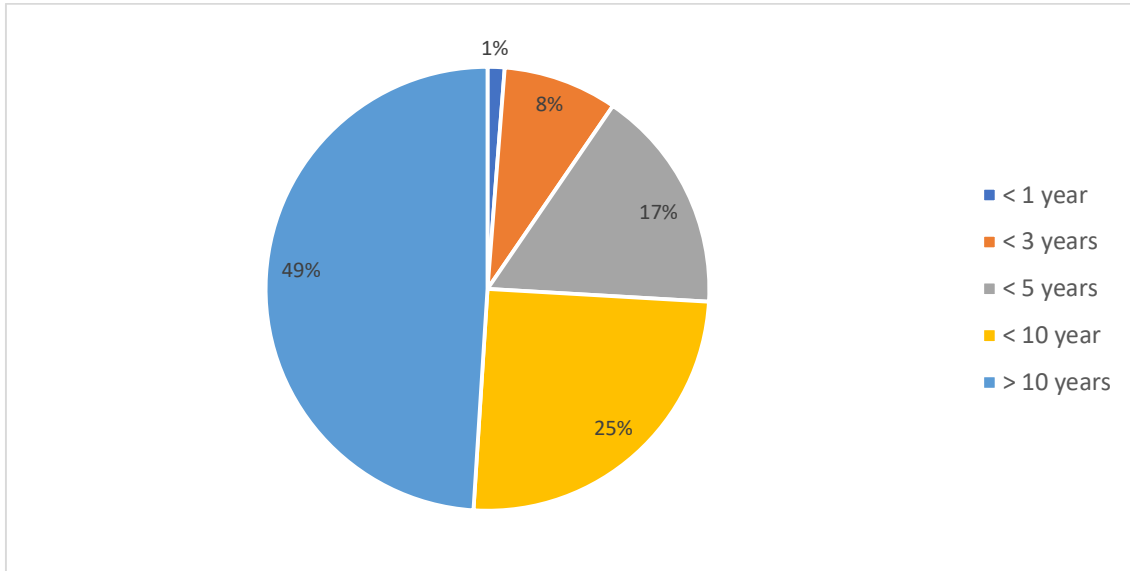


### Dynamically typed languages (e.g., Javascript, Ruby, Python, PHP, Perl)



How long have you been programming?

	Frequency	Percent
< 1 year	114	1.2%
< 3 years	771	8.3%
< 5 years	1511	16.3%
< 10 year	2321	25.1%
> 10 years	4528	49.0%



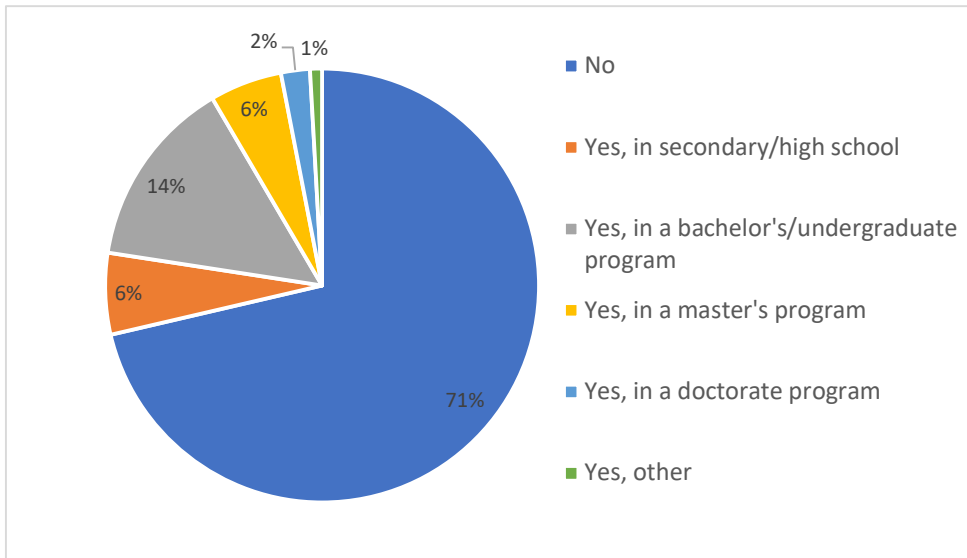
## Work and study

All respondents were asked the next two questions

Are you a full- or part-time student?

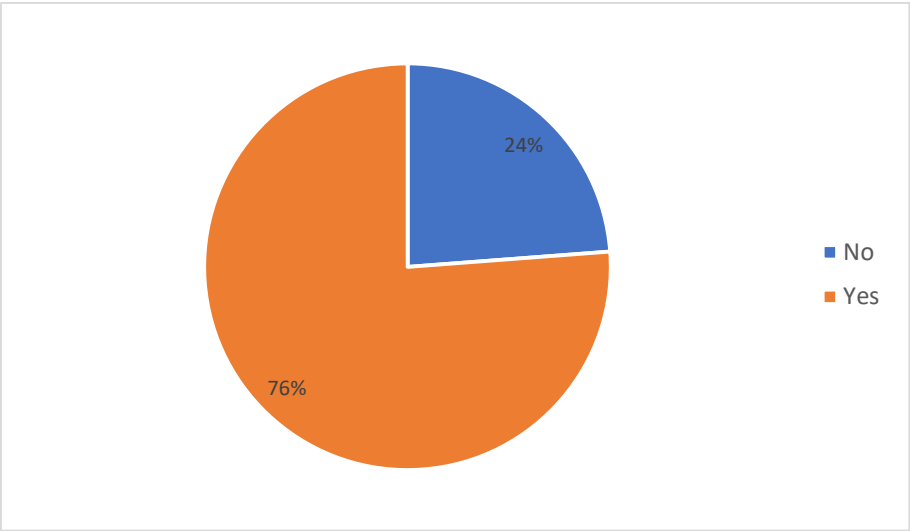
	Frequency	Percent
No	6561	71.4%
Yes, in secondary/high school	558	6.1%
Yes, in a bachelor's/undergraduate program	1299	14.1%
Yes, in a master's program	494	5.4%
Yes, in a doctorate program	198	2.2%
Yes, other*	83	0.9%

\* other includes 'yes, in a vocational program'.



Are you employed full- or part-time (including paid internships)?

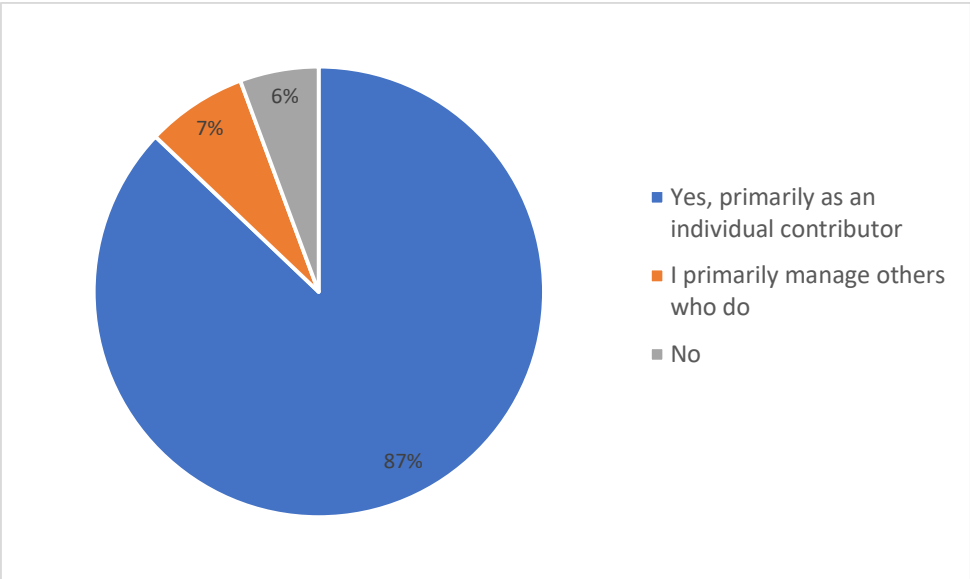
	Frequency	Percent
No	2197	23.8%
Yes	7028	76.2%



Do you write or design software in your work?

Only asked if the respondent answered 'yes' to the previous question.

	Frequency	Percent
Yes, primarily as an individual contributor	6156	87.1%
I primarily manage others who do	512	7.2%
No	401	5.7%

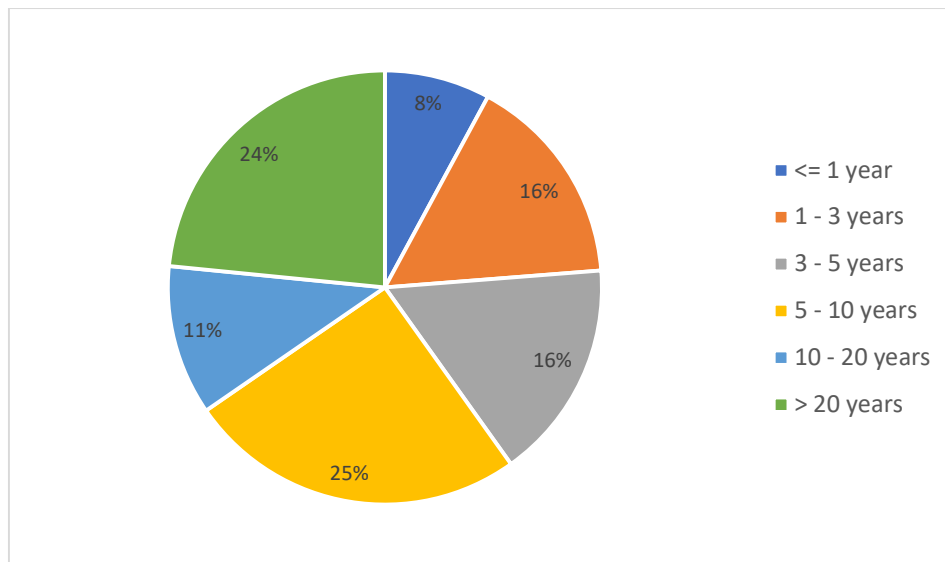




## Questions for respondents who write or design software

How long have you worked in software professionally?

	Frequency	Percent
<= 1 year	525	7.9%
1 - 3 years	1057	15.9%
3 - 5 years	1091	16.4%
5 - 10 years	1684	25.3%
10 - 20 years	745	11.2%
> 20 years	1560	23.4%

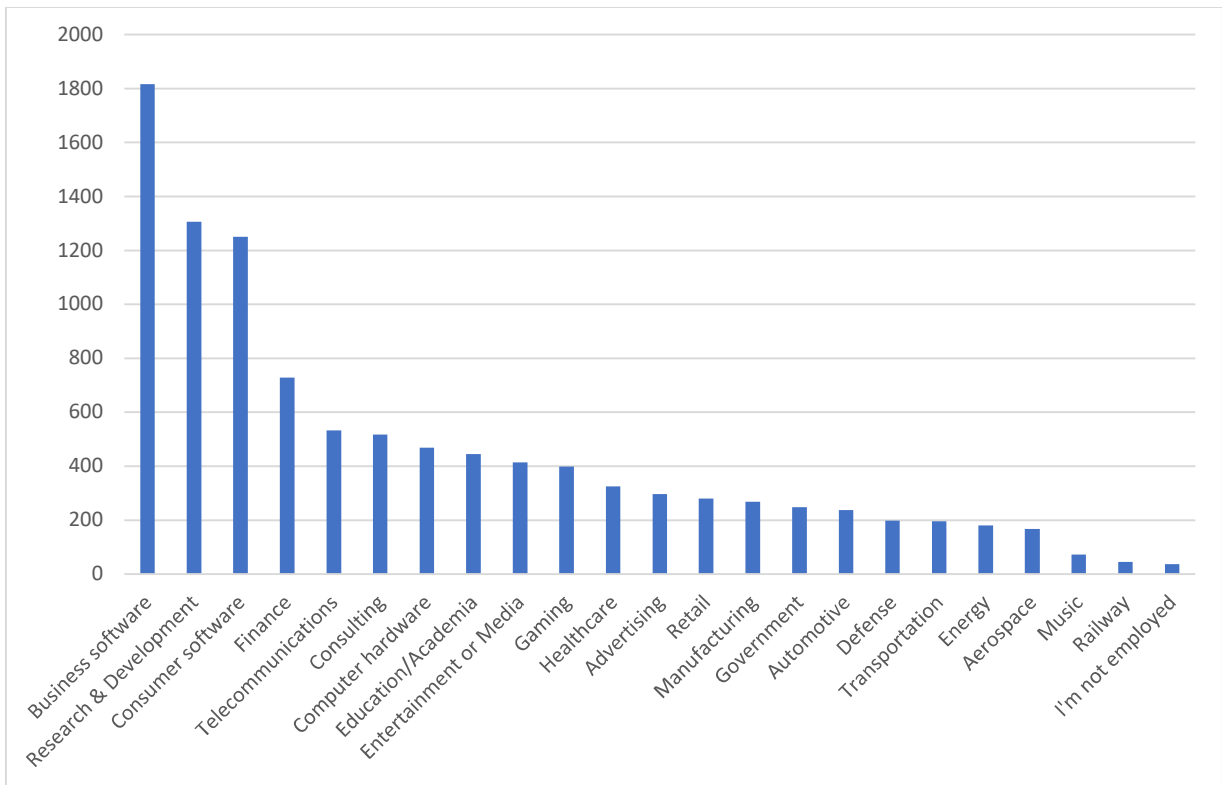


What category best describes your current employer's industry?

All respondents were asked this question, but these results are for Rust users only.

Business software	1817
Research & Development	1306
Consumer software	1250
Finance	728
Telecommunications	533
Consulting	517
Computer hardware	468
Education/Academia	445
Entertainment or Media	414
Gaming	398
Healthcare	326
Advertising	297
Retail	280
Manufacturing	269
Government	247
Automotive	238

Defense	198
Transportation	196
Energy	181
Aerospace	167
Music	72
Railway	45
I'm not employed	37

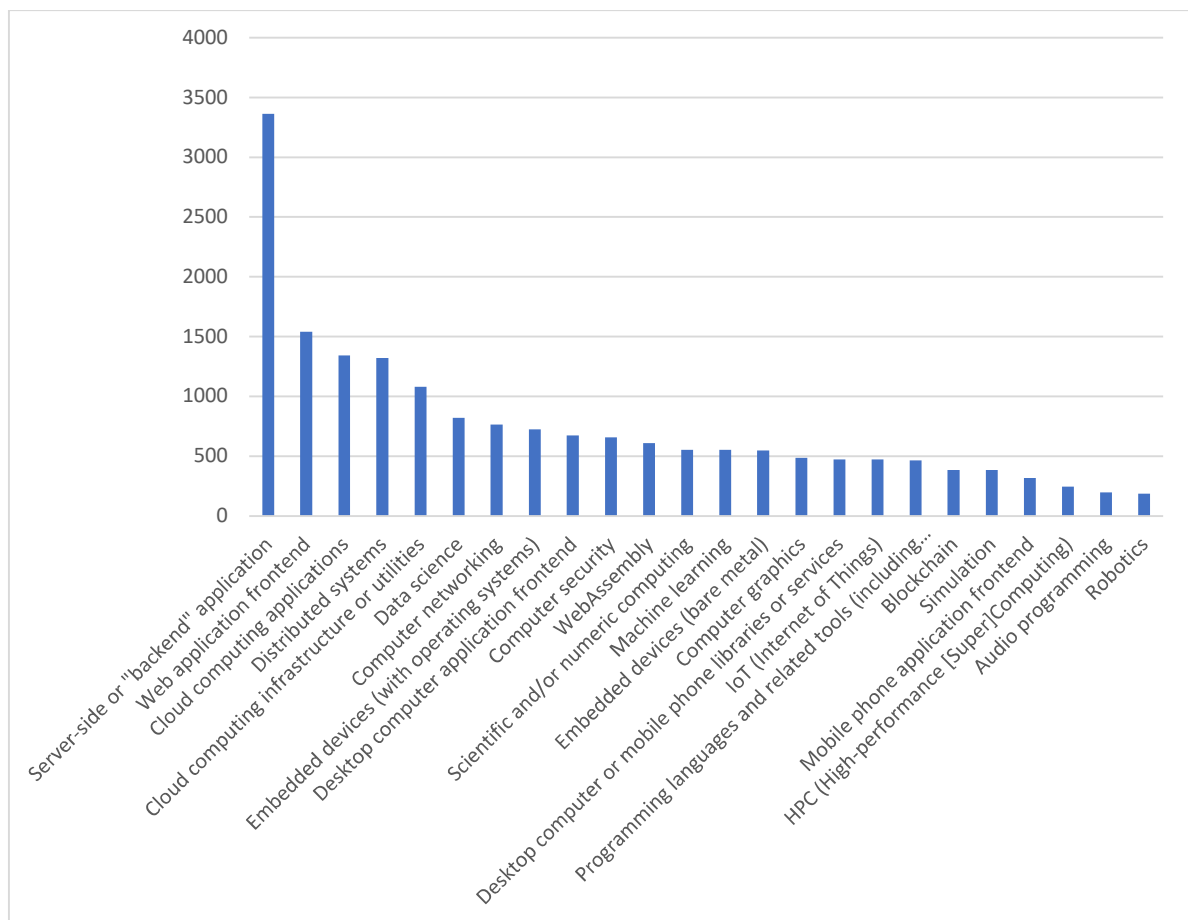


Which categories best describe the tech domain(s) you currently write or design software in?

All respondents were asked this question, but these results are for Rust users only.

Server-side or "backend" application	3365
Web application frontend	1540
Cloud computing applications	1345
Distributed systems	1321
Cloud computing infrastructure or utilities	1080
Data science	820
Computer networking	765
Embedded devices (with operating systems)	724
Desktop computer application frontend	675
Computer security	657
WebAssembly	610

Scientific and/or numeric computing	555
Machine learning	553
Embedded devices (bare metal)	549
Computer graphics	486
Desktop computer or mobile phone libraries or services	473
IoT (Internet of Things)	473
Programming languages and related tools (including compilers, IDEs, standard libraries, etc.)	465
Blockchain	386
Simulation	386
Mobile phone application frontend	318
HPC (High-performance [Super]Computing)	244
Audio programming	197
Robotics	188

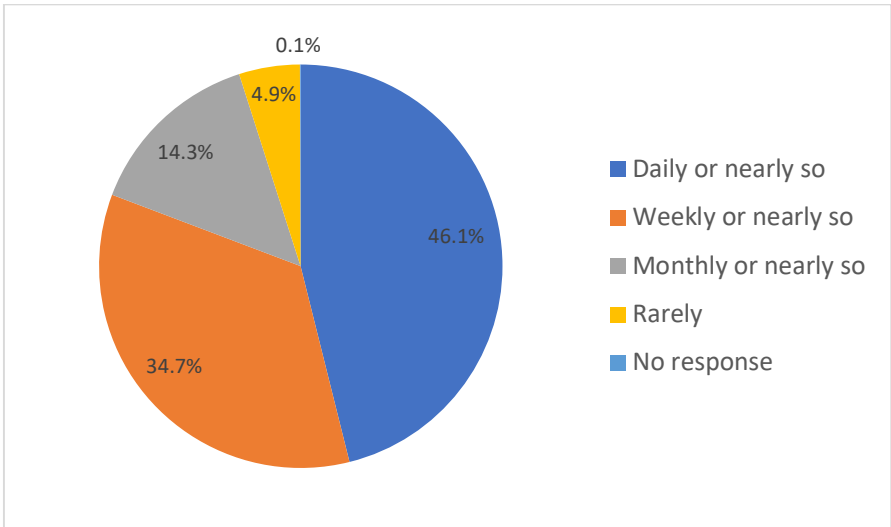


## Questions for Rust users

The following questions were asked of respondents who answered 'yes' to the first question.

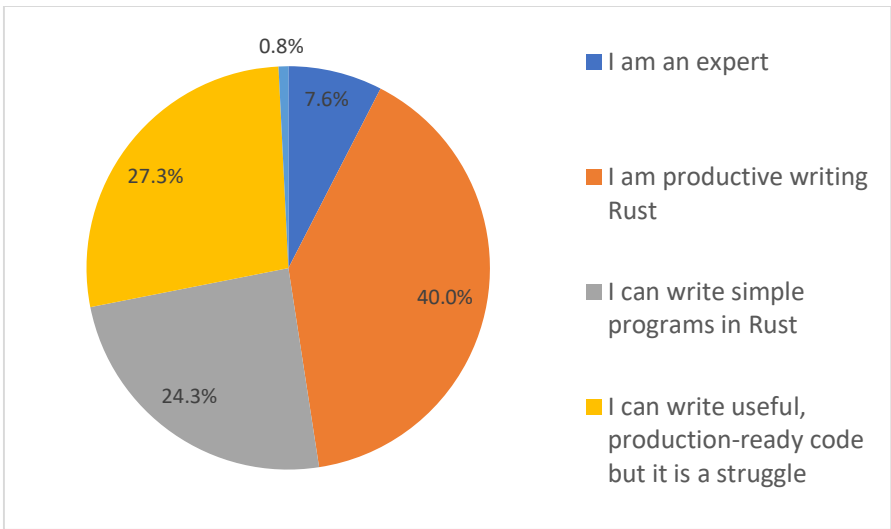
On average, how often do you use Rust?

	Frequency	Percent
Daily or nearly so	3871	46.1%
Weekly or nearly so	2910	34.7%
Monthly or nearly so	1200	14.3%
Rarely	409	4.9%



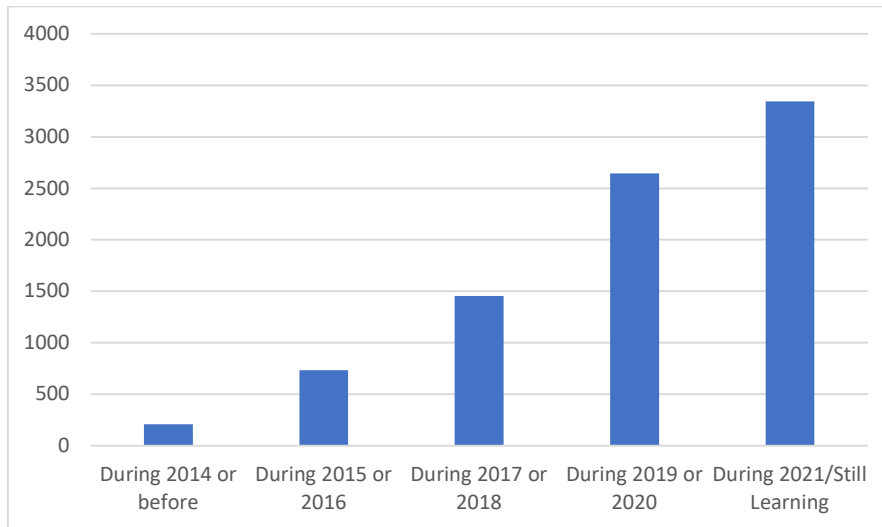
How would you rate your Rust expertise?

	Frequency	Percent
I am an expert	634	7.6%
I am productive writing Rust	3356	40.0%
I can write simple programs in Rust	2041	24.3%
I can write useful, production-ready code but it is a struggle	2293	27.3%
I can't read or write Rust	65	0.8%



When did you learn to program in Rust?

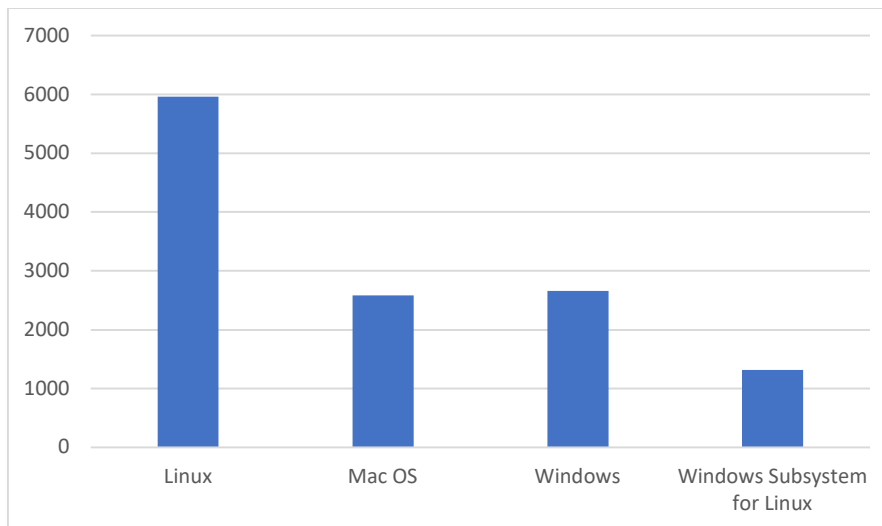
During 2014 or before	208	2.5%
During 2015 or 2016	733	8.7%
During 2017 or 2018	1455	17.4%
During 2019 or 2020	2646	31.6%
During 2021/Still Learning	3344	39.9%



Which operating systems do you use regularly for Rust development?

Linux	5962	64%
Mac OS	2582	28%
Windows	2655	29%
Windows Subsystem for Linux	1316	14%
BSD (all flavours)*	59	1%

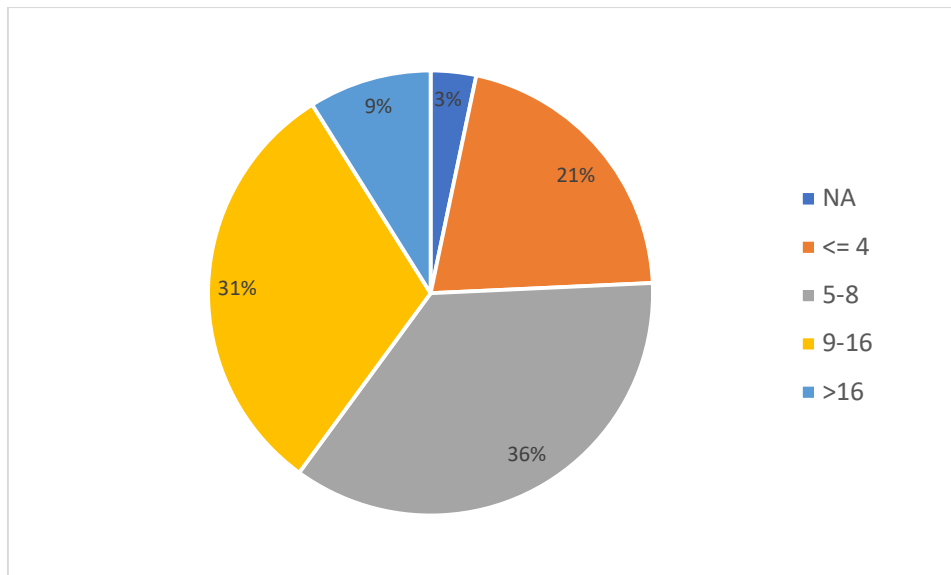
\* BSD was not a supplied answer but was the most common 'other' answer



On the primary machine you compile Rust code on, how many CPUs are there?

This was a free-form question. There was an obvious bias towards 'round numbers' (4, 8, 16, ...). The largest number of cores reported was 256 and the smallest was 1

	Frequency	Percent
NA	278	3.30%
<= 4	1763	21.00%
5-8	3006	35.80%
9-16	2607	31.00%
>16	751	8.60%

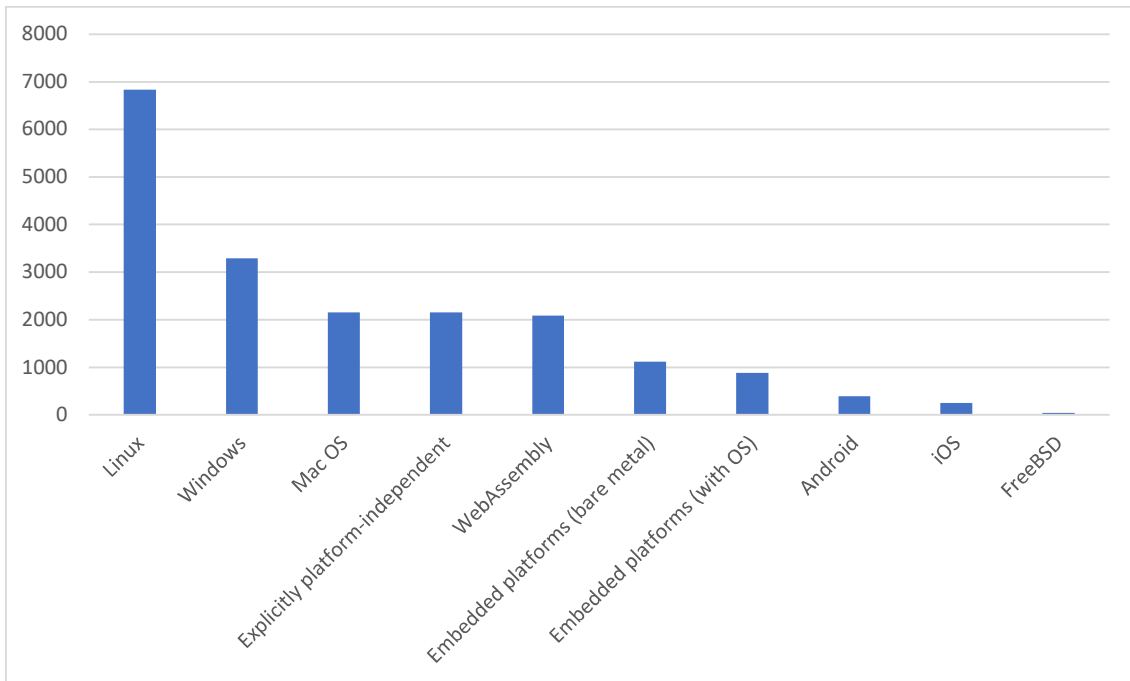


Which operating systems or runtimes do you develop Rust software for?

Linux (desktop or server)	6833	73%
Windows	3290	35%
Mac OS	2156	23%
Explicitly platform-independent	2154	23%
WebAssembly	2089	22%
Embedded platforms (bare metal)	1117	12%
Embedded platforms (with an operating system)	886	10%
Android	389	4%
iOS	252	3%
FreeBSD*	42	0%
All Platforms**	1039	11%

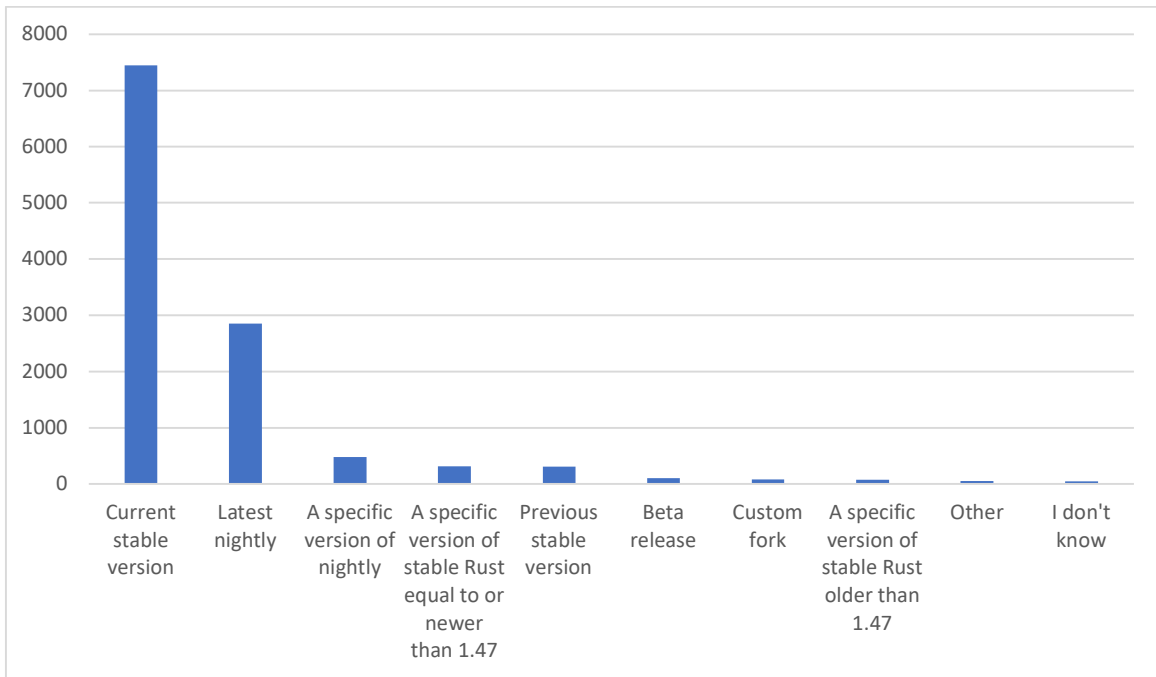
\* FreeBSD was not a supplied answer but was the most common 'other' answer

\*\* All platforms is the count of respondents who selected all answers.



Which version(s) of Rust do you use for local development?

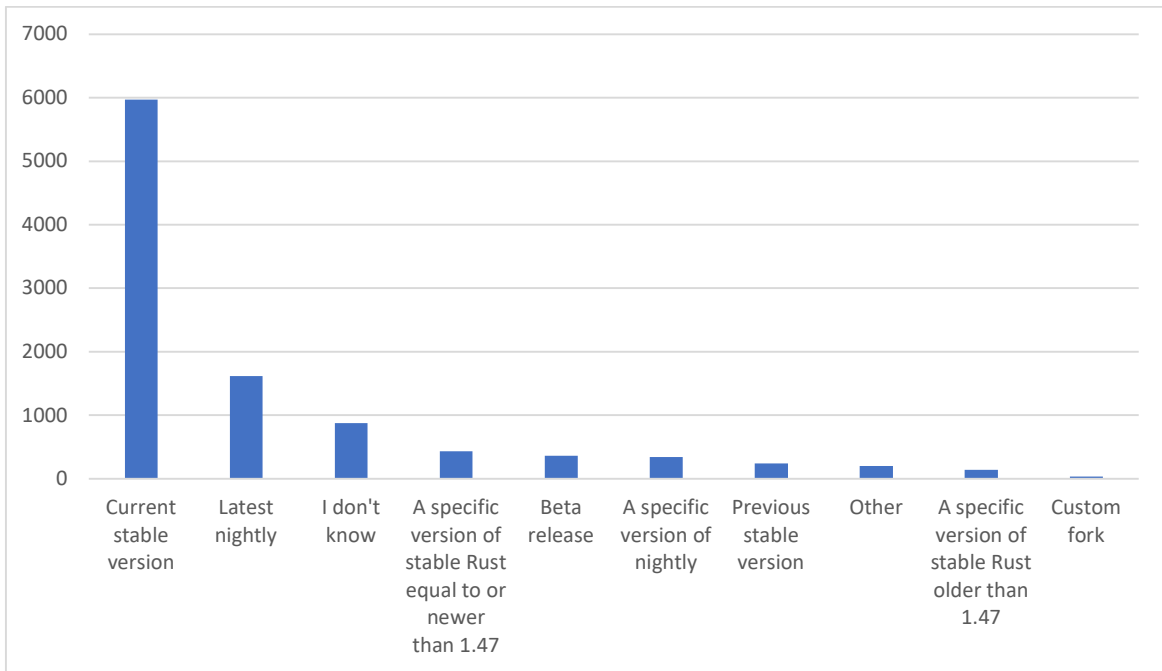
Current stable version [i.e., 1.57]	7449
Latest nightly	2852
A specific version of nightly	481
A specific version of stable Rust equal to or newer than 1.47	313
Previous stable version [i.e., 1.56 or earlier]	305
Beta release	106
Custom fork	77
A specific version of stable Rust older than 1.47	73
Other	50
I don't know	46



Which version(s) of Rust do you use in automated testing (e.g., CI)?

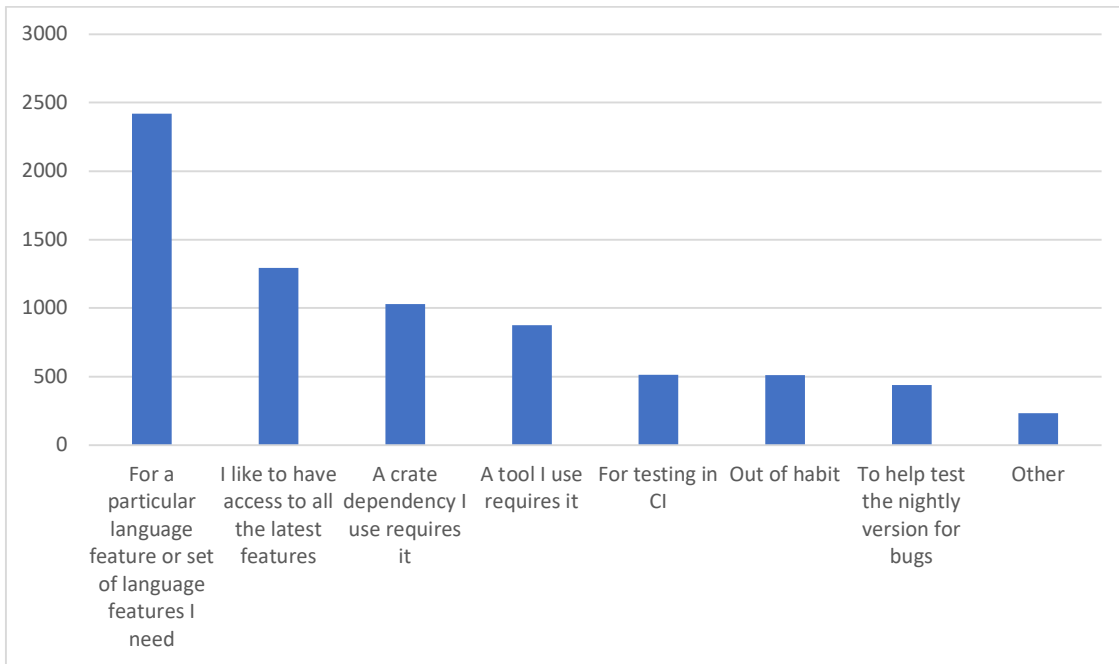
Current stable version [i.e., 1.57]	5970
Latest nightly	1615
I don't know	875
A specific version of stable Rust equal to or newer than 1.47	431
Beta release	360
A specific version of nightly	345
Previous stable version [i.e., 1.56 or earlier]	244
Other	199
A specific version of stable Rust older than 1.47	139
Custom fork	36





If you use nightly, why?

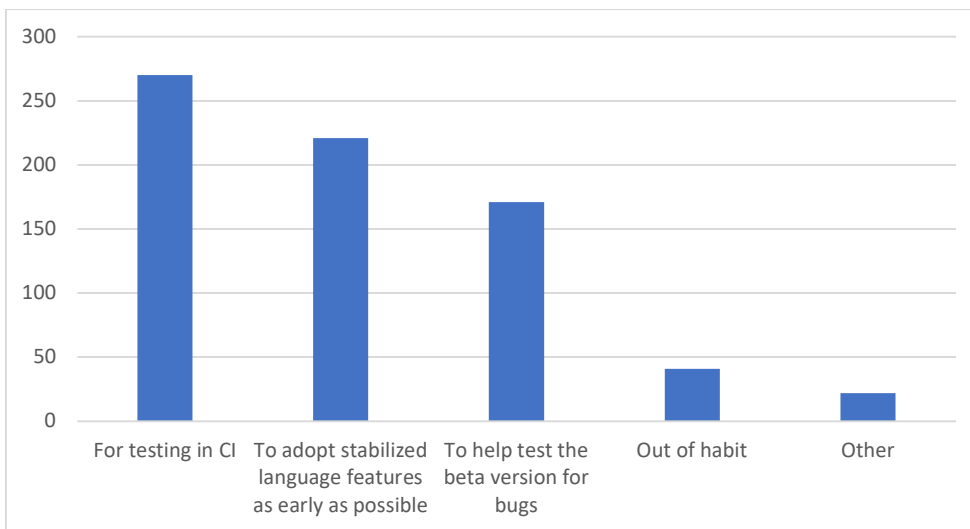
I don't use nightly	3828
For a particular language feature or set of language features I need	2420
I like to have access to all the latest features	1293
A crate dependency I use requires it	1028
A tool I use requires it	876
For testing in CI	513
Out of habit	511
To help test the nightly version for bugs	439
Other	231



Note that graph excludes non-nightly users.

If you use beta, why?

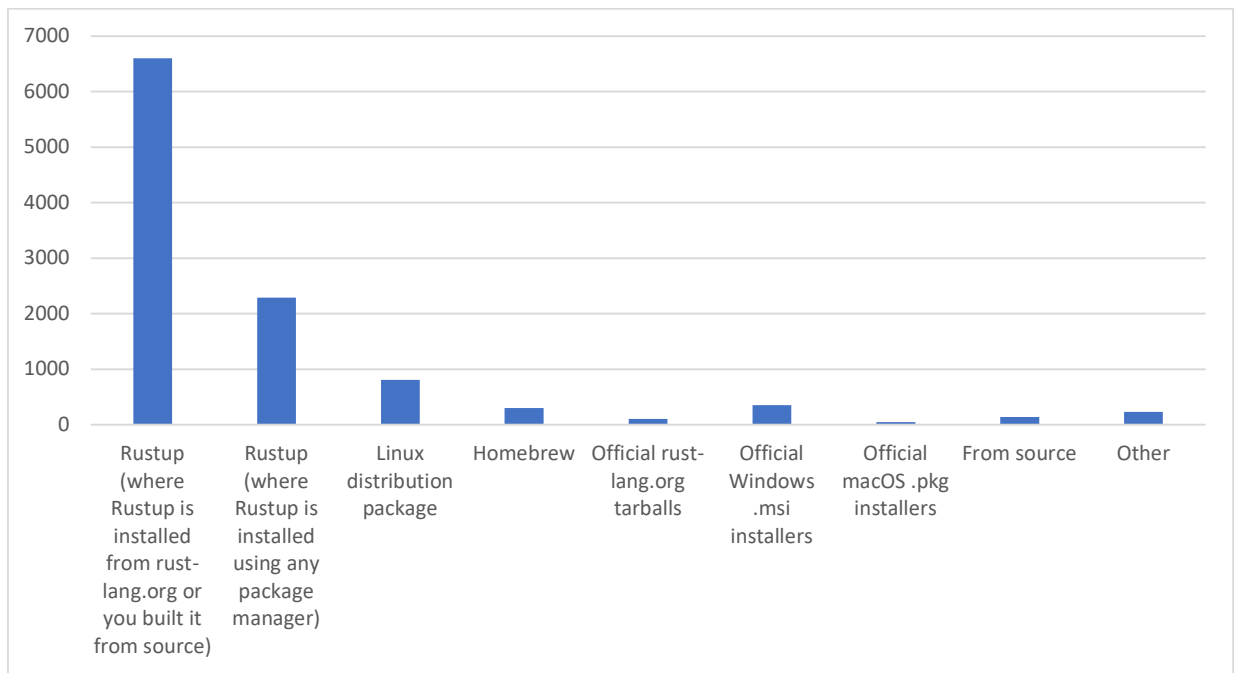
I don't use beta	6957
For testing in CI	270
To adopt stabilized language features as early as possible	221
To help test the beta version for bugs	171
Out of habit	41
Other	22



Note that graph excludes non-beta users.

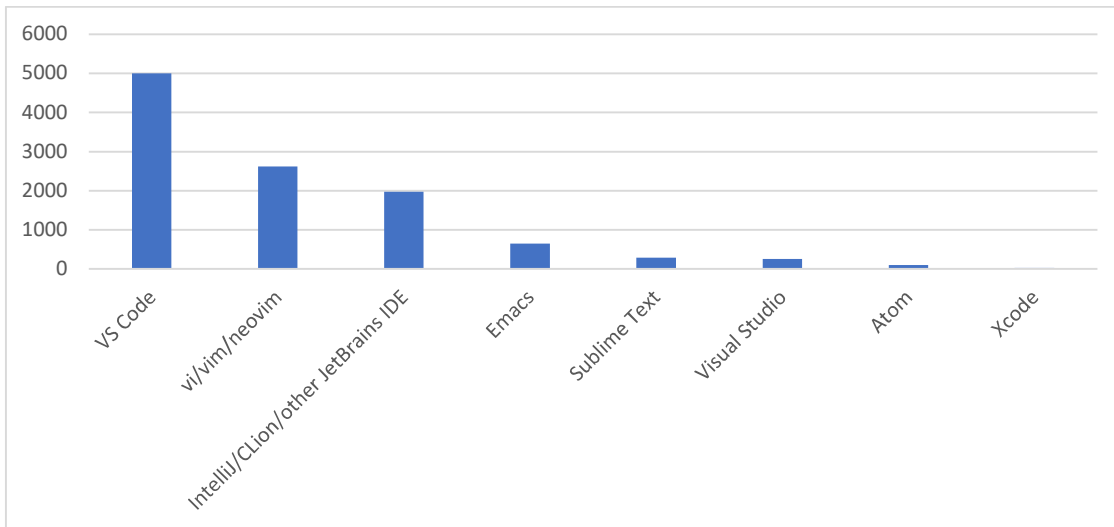
In which ways do you install Rust?

Rustup (where Rustup is installed from rust-lang.org or you built it from source)	6600
Rustup (where Rustup is installed using any package manager)	2288
Linux distribution package	807
Homebrew	300
Official rust-lang.org tarballs	105
Official Windows .msi installers	354
Official macOS .pkg installers	48
From source	137
Other	229



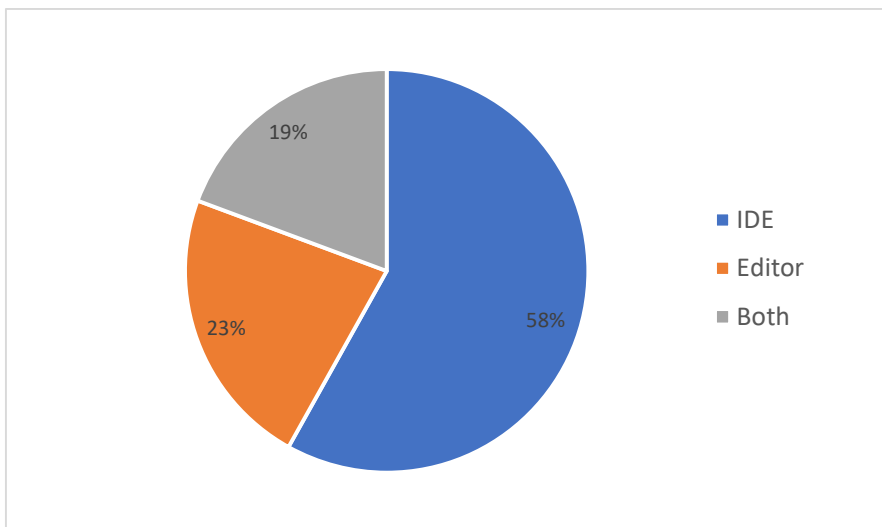
Which editor or IDE setup do you use with Rust code on a regular basis?

VS Code	4997	53.69%
vi/vim/neovim	2620	28.15%
IntelliJ/CLion/other JetBrains IDE	1977	21.24%
Emacs	648	6.96%
Sublime Text	288	3.09%
Visual Studio	251	2.70%
Atom	101	1.09%
Xcode	24	0.26%



Grouped answers:

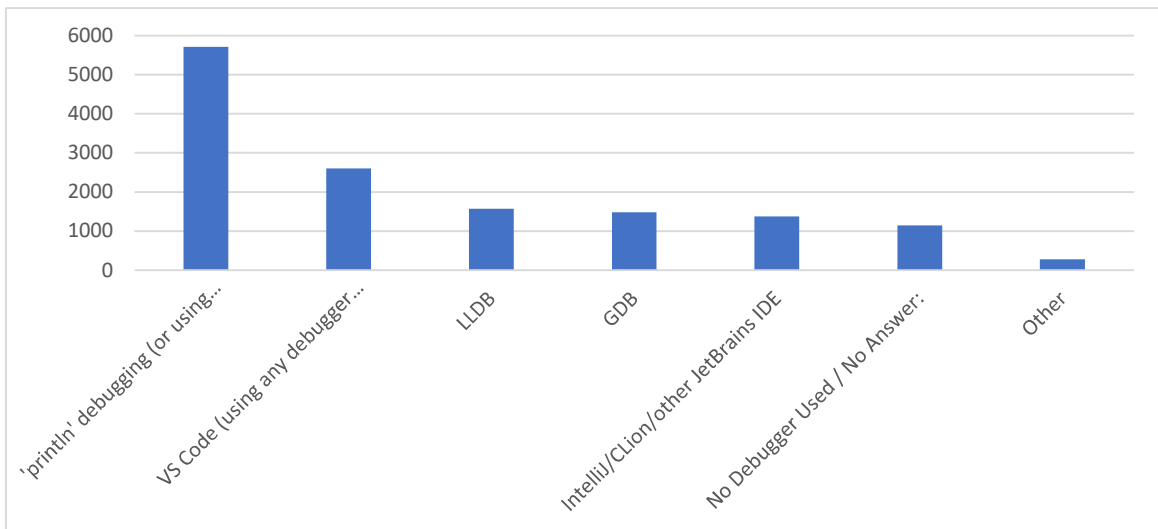
Any IDE (VS code, IntelliJ, Visual Studio, XCode)	4791	58%
Any Editor (VI, Emacs, Sublime Text, Atom)	1862	23%
Both	1595	19%



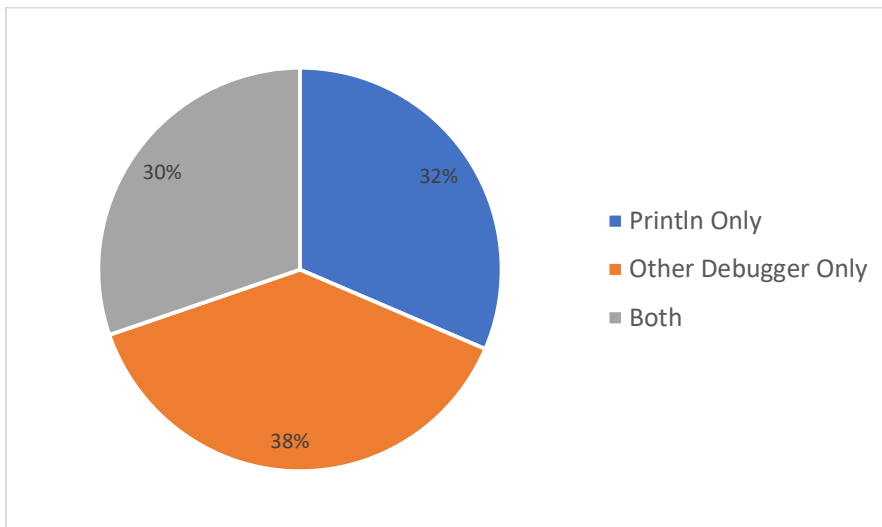
Which debuggers do you use for debugging Rust programs on a regular basis?

'println' debugging (or using any logging or tracing crate, etc.)	5713	61.4%
VS Code (using any debugger extension)	2599	27.9%
LLDB	1568	16.8%
GDB	1490	16.0%
IntelliJ/CLion/other JetBrains IDE	1379	14.8%
No Debugger Used / No Answer:	1152	12.4%
Other*	283	3.0%

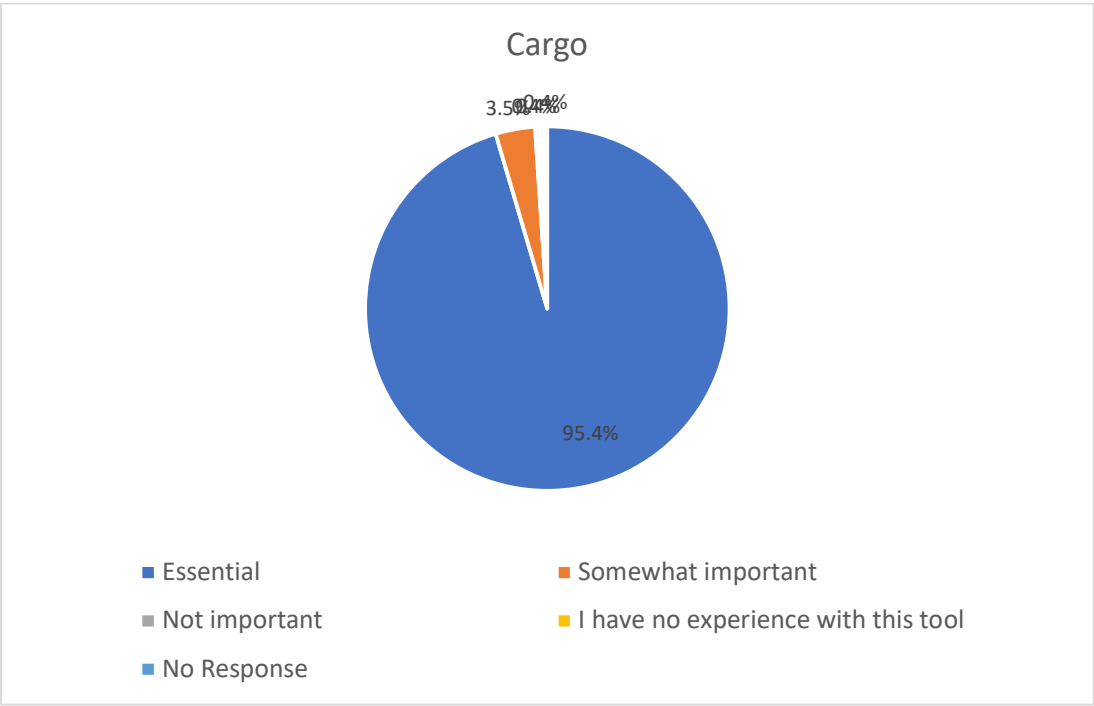
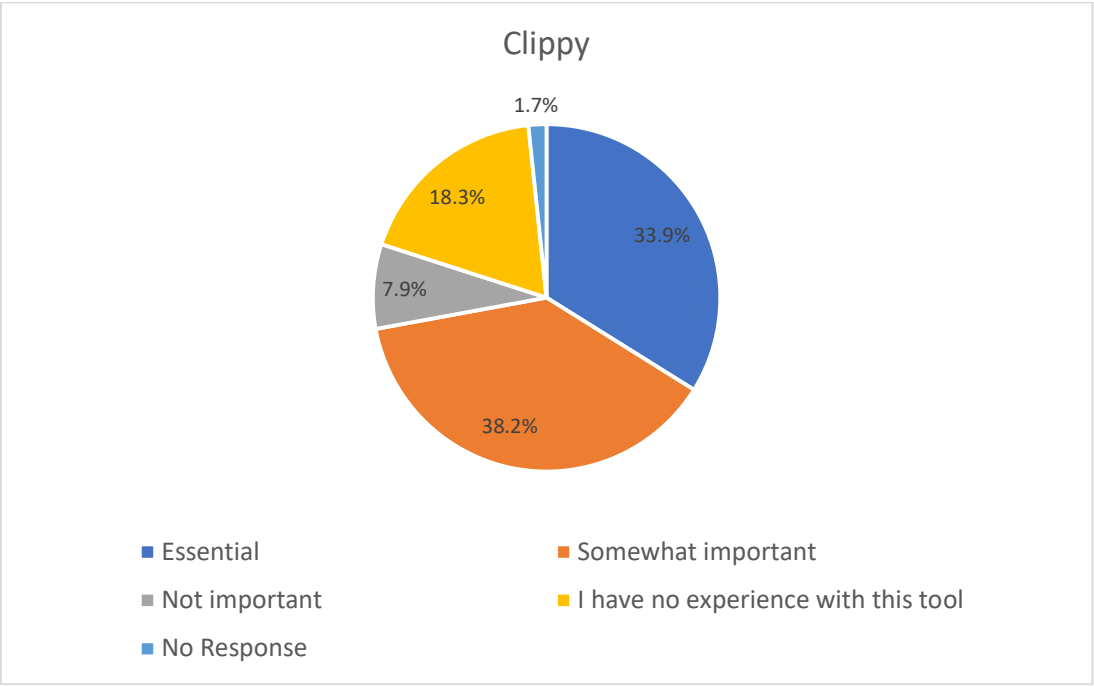
\*Other is a grouping of Visual Studio, RR/Pernosco, WinDBG, and free-form other answers.



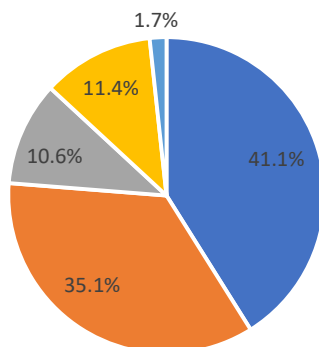
Summary of 'println' debugging vs using a debugger:



Please indicate how vital to your workflow each of the following tools are when programming with Rust

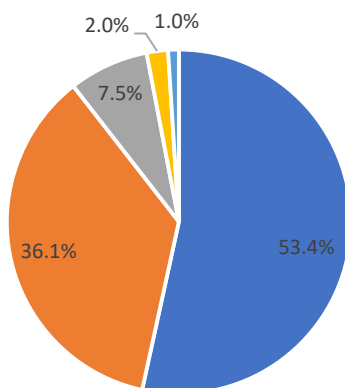


Rustdoc/cargo doc



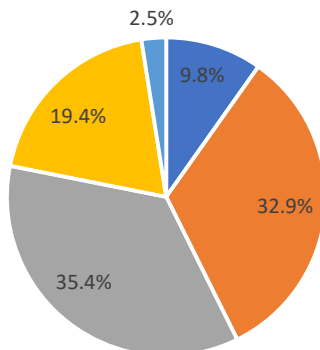
- Essential
- Somewhat important
- Not important
- I have no experience with this tool
- No Response

Rustup



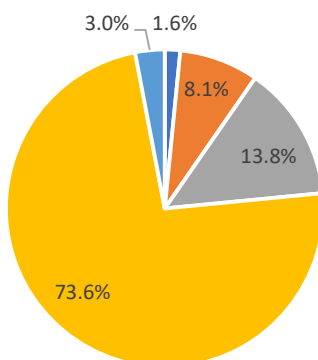
- Essential
- Somewhat important
- Not important
- I have no experience with this tool
- No Response

play.rust-lang.org



- Essential
- Somewhat important
- Not important
- I have no experience with this tool
- No Response

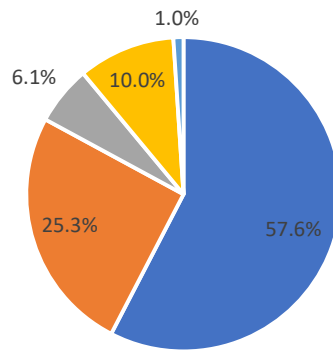
miri



- Essential
- Somewhat important
- Not important
- I have no experience with this tool
- No Response

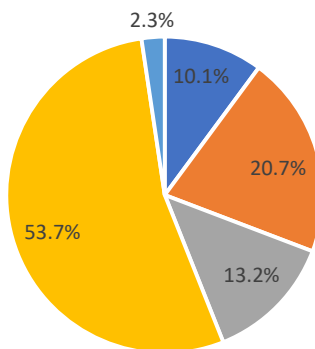


Rustfmt/cargo fmt



- Essential
- Somewhat important
- Not important
- I have no experience with this tool
- No Response

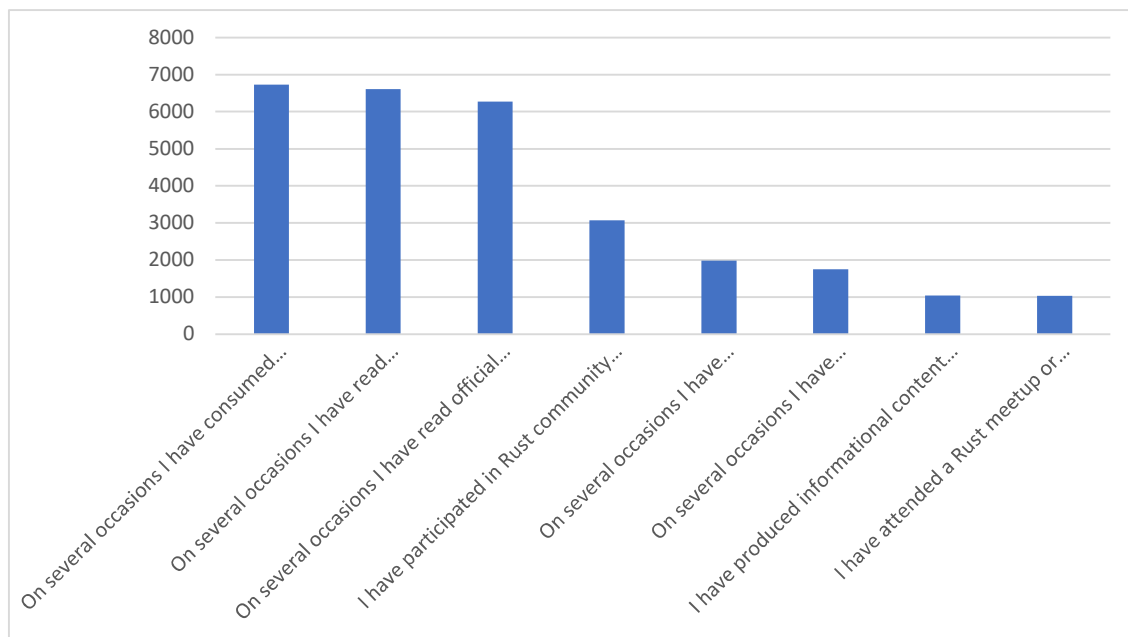
Bindgen



- Essential
- Somewhat important
- Not important
- I have no experience with this tool
- No Response

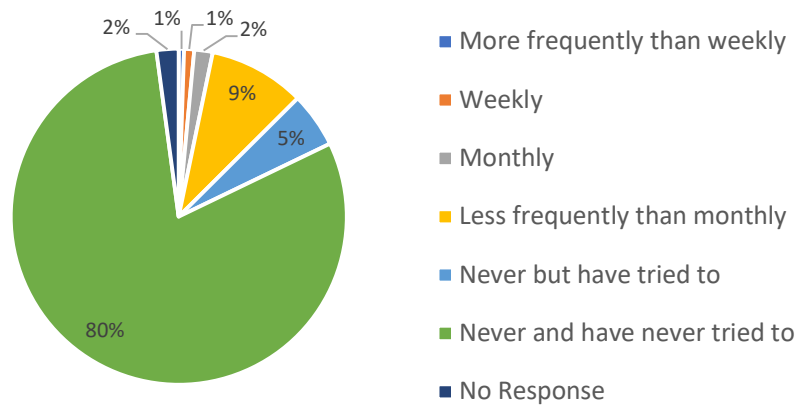
In which of the following ways have you participated in the Rust community in the last 6 months

On several occasions I have consumed informational content about Rust (e.g., blogs, live streams, YouTube videos, etc.)	6724
On several occasions I have read comments about Rust content on "news" sites (e.g., Hacker News, reddit.com/r/rust, lobste.rs/t/rust, etc.)	6611
On several occasions I have read official Rust communication channels (e.g., This Week in Rust, the official Rust blog, the Rust Twitter account, etc.)	6268
I have participated in Rust community forums or chats (e.g., users.rust-lang.org, Rust Discord, a local Rust chat community, etc.)	3077
On several occasions I have commented on Rust content on "news" sites (e.g., Hacker News, reddit.com/r/rust, lobste.rs/t/rust, etc.)	1979
On several occasions I have participated in conversations about Rust on social media (Twitter, Facebook, LinkedIn, etc.)	1748
I have produced informational content about Rust (e.g., blogged, live streamed, made a YouTube video, presented at a conference/meetup, etc.)	1042
I have attended a Rust meetup or conference (virtual or in-person)	1025

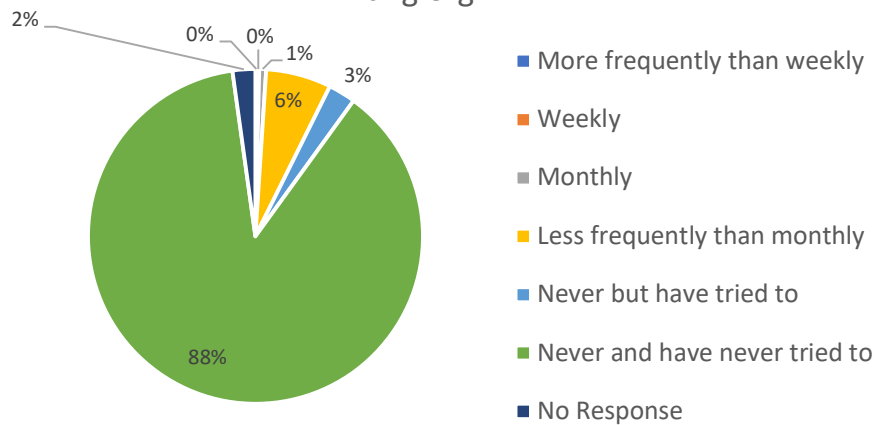


Roughly how often do you contribute to the Rust project?

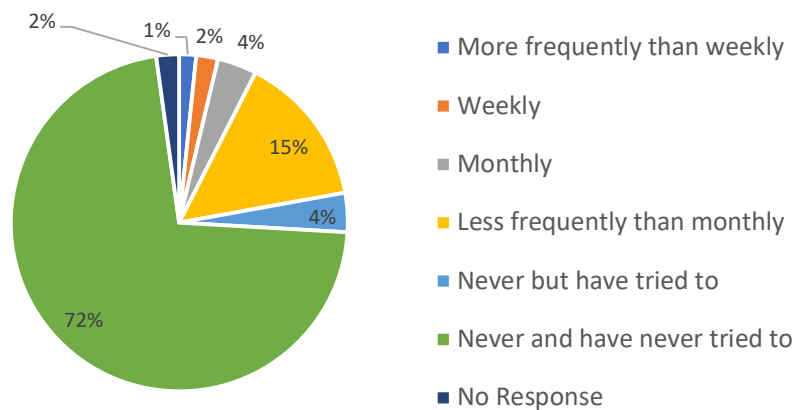
Comment on, contribute to discussion of, or provide edits to an open RFC



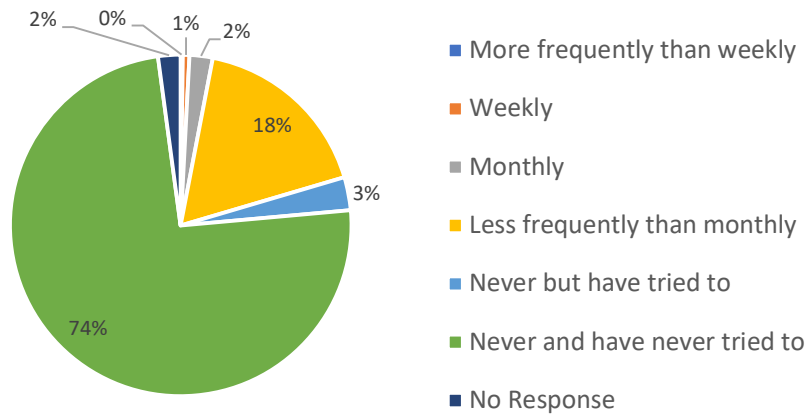
Create a new thread or comment on [internals.rust-lang.org](https://internals.rust-lang.org)



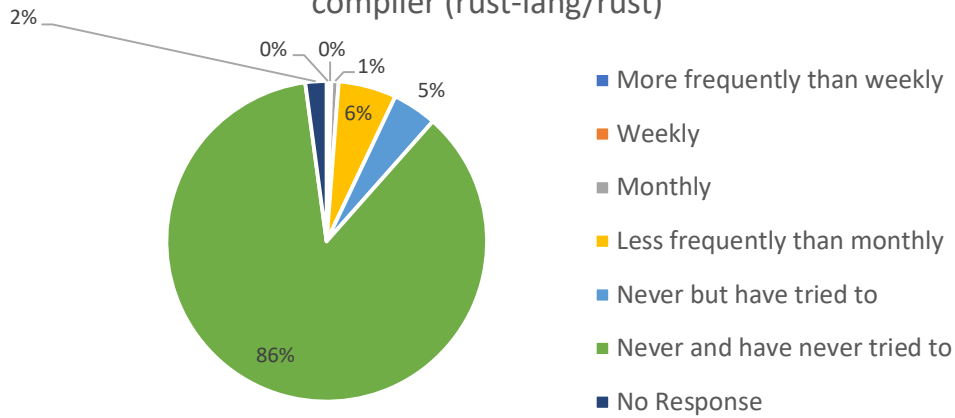
Discuss the Rust project in an official chat (either Zulip or Discord)



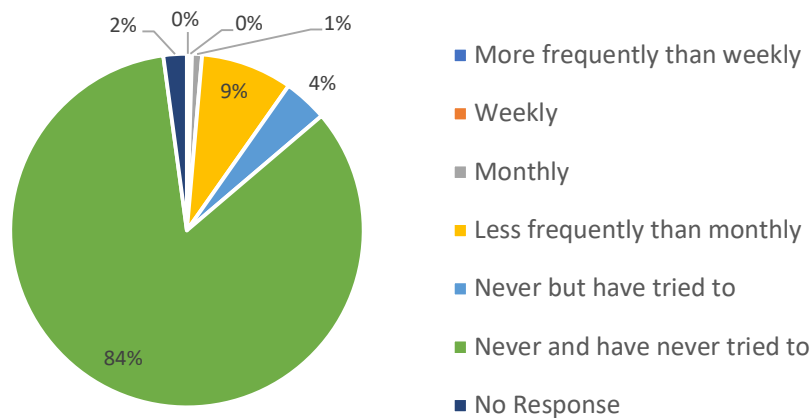
### Open an issue on any repo in the rust-lang GitHub organization



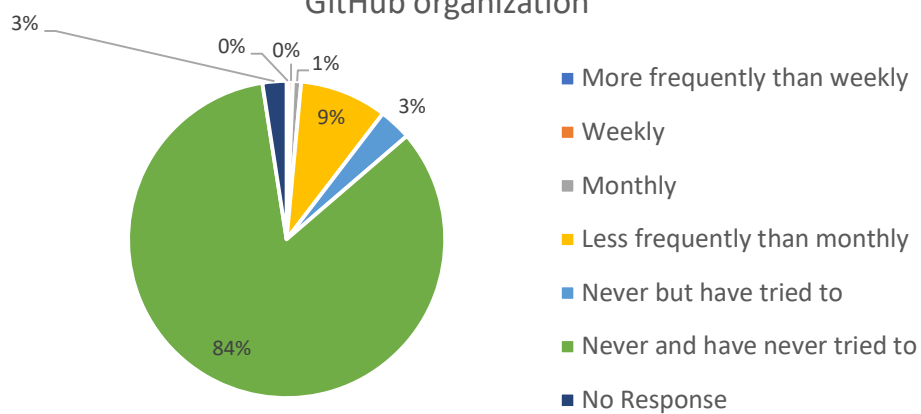
### Contribute code changes (including tests) to the Rust compiler (rust-lang/rust)



### Contribute code changes (including tests) to any other project in the rust-lang GitHub organization

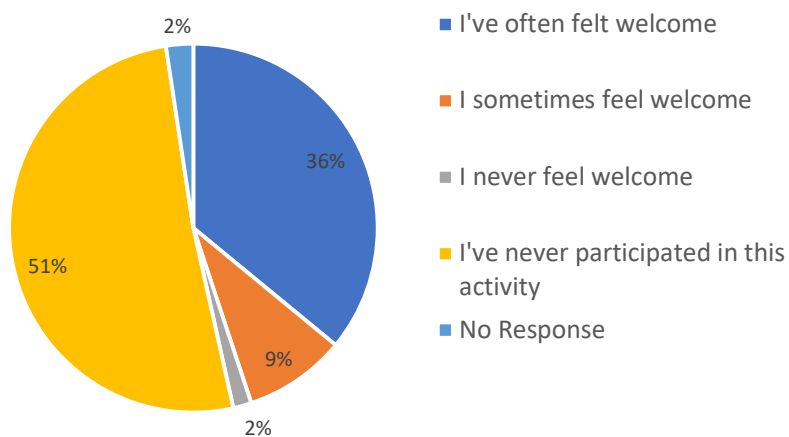


Contribute non-code changes (documentation, comments, etc.) to any project in the rust-lang GitHub organization

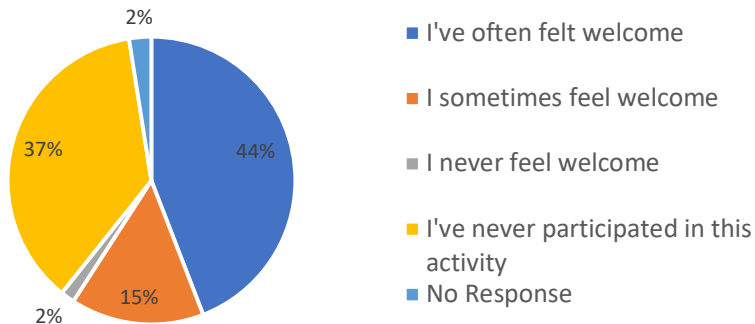


How often have you felt explicitly welcome in the Rust community?

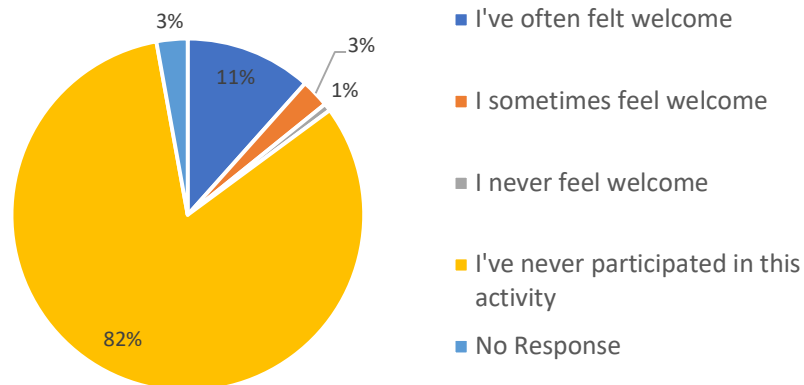
Official Rust community forums or chats  
(users.rust-lang.org, internals.rust-lang.org, the  
official Rust Discord, or the Rust Zulip)



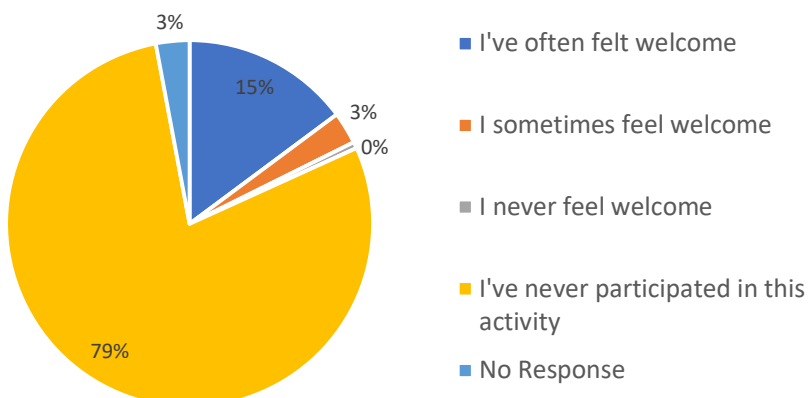
Unofficial Rust community forums or chats (e.g.,  
reddit.com/r/rust, Hacker News, the Rust  
Community Discord, etc.)



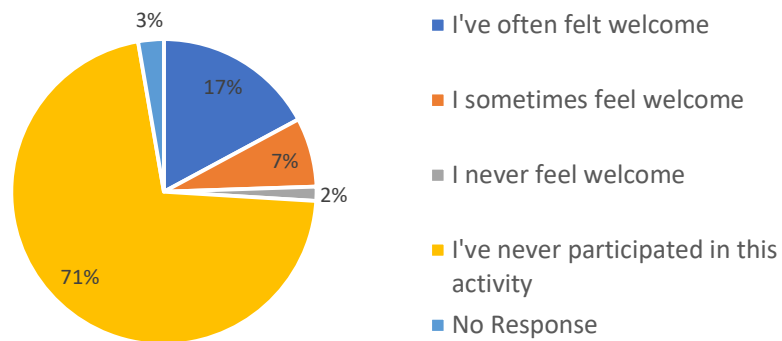
Attending a Rust conference



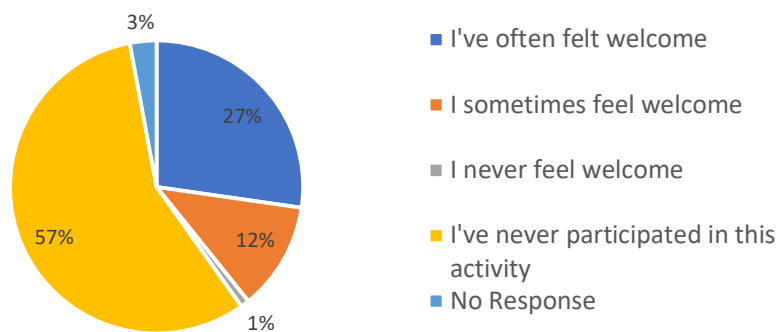
Attending a Rust meetup or local community event



Discussion (issues, pull requests, etc.) on a repository inside the rust-lang GitHub organization

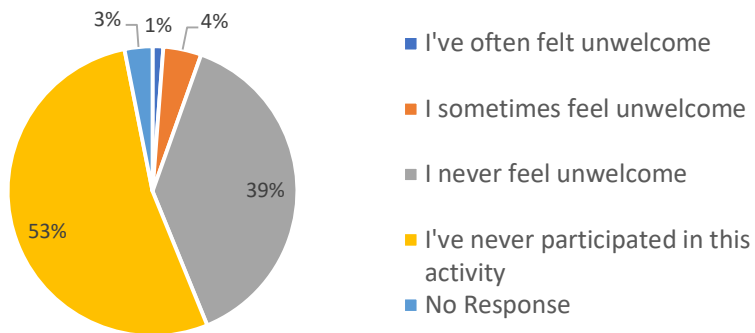


Discussion (issues, pull requests, etc.) on a repository outside of the rust-lang GitHub organization

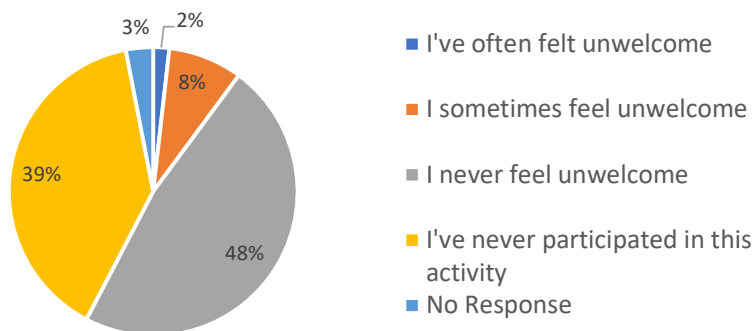


How often have you felt explicitly unwelcome in the Rust community?

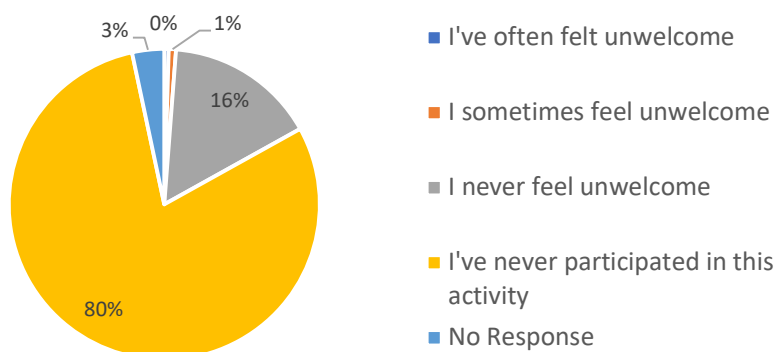
Official Rust community forums or chats  
(users.rust-lang.org, internals.rust-lang.org, the  
official Rust Discord, or the Rust Zulip)



Unofficial Rust community forums or chats (e.g.,  
reddit.com/r/rust, Hacker News, the Rust  
Community Discord, etc.)

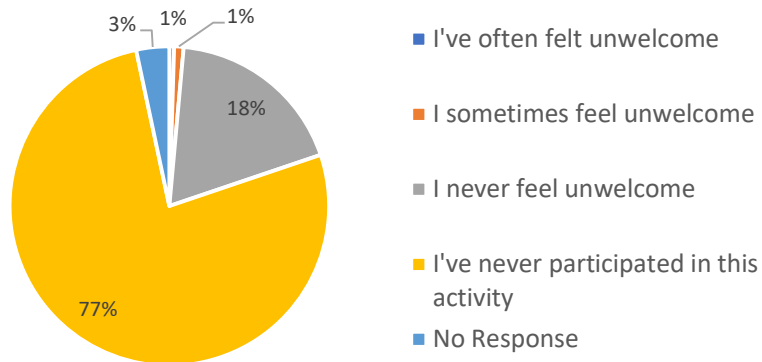


Attending a Rust conference

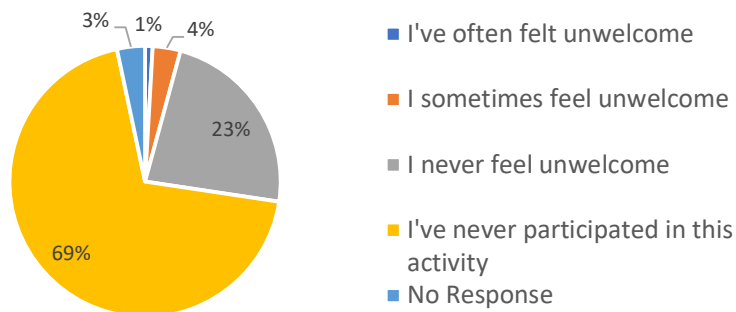




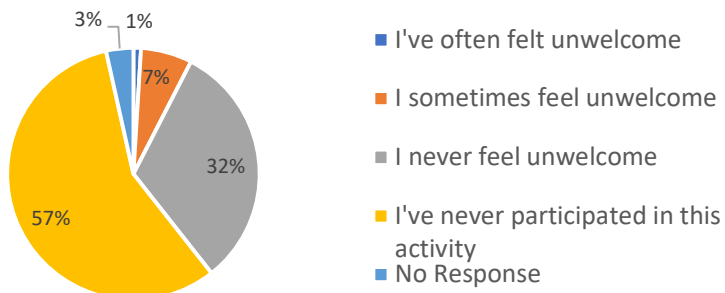
### Attending a Rust meetup or local community event



### Discussion (issues, pull requests, etc.) on a repository inside the rust-lang GitHub organization

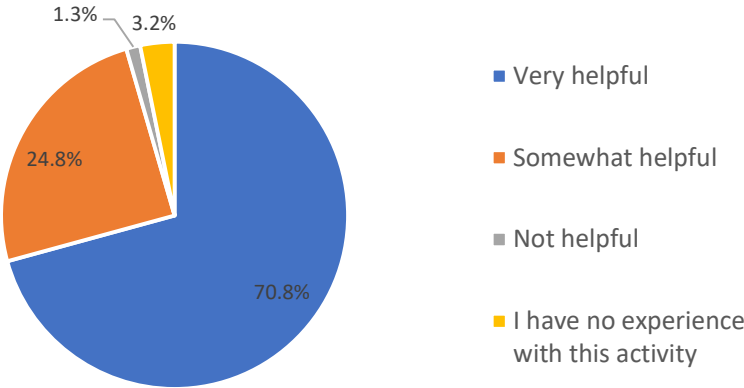


### Discussion (issues, pull requests, etc.) on a repository outside of the rust-lang GitHub organization

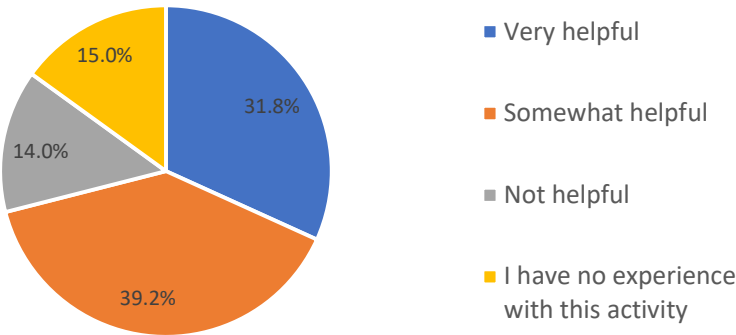


Which of the following activities do you find helpful or effective for learning Rust or improving your Rust skills?

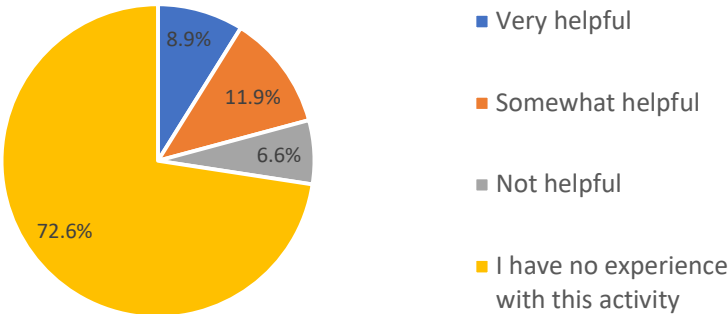
Reading books or other written material geared towards learning Rust



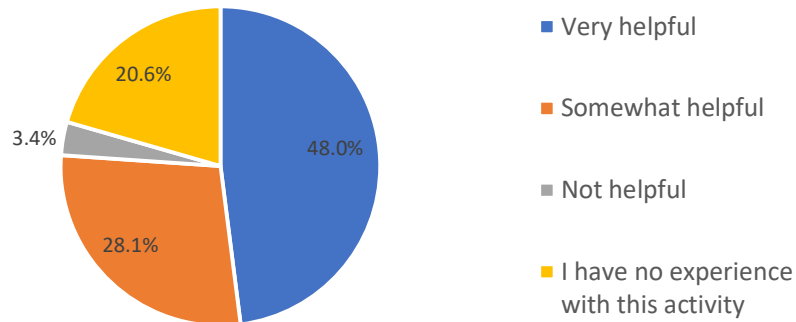
Watching videos, streams, etc. geared towards learning Rust



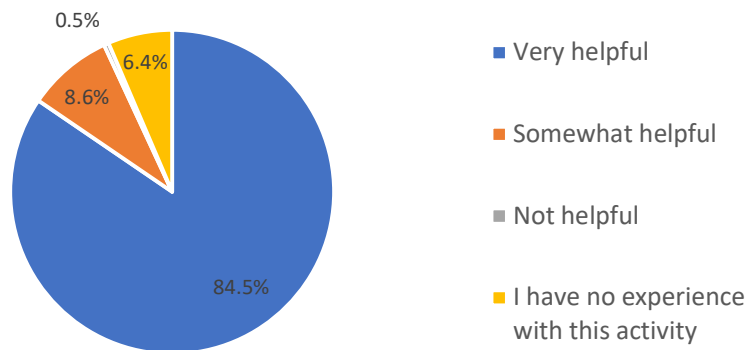
Attending an organized training session or course (in person or virtual)



### Doing Rust coding exercises or challenges created to help learn Rust

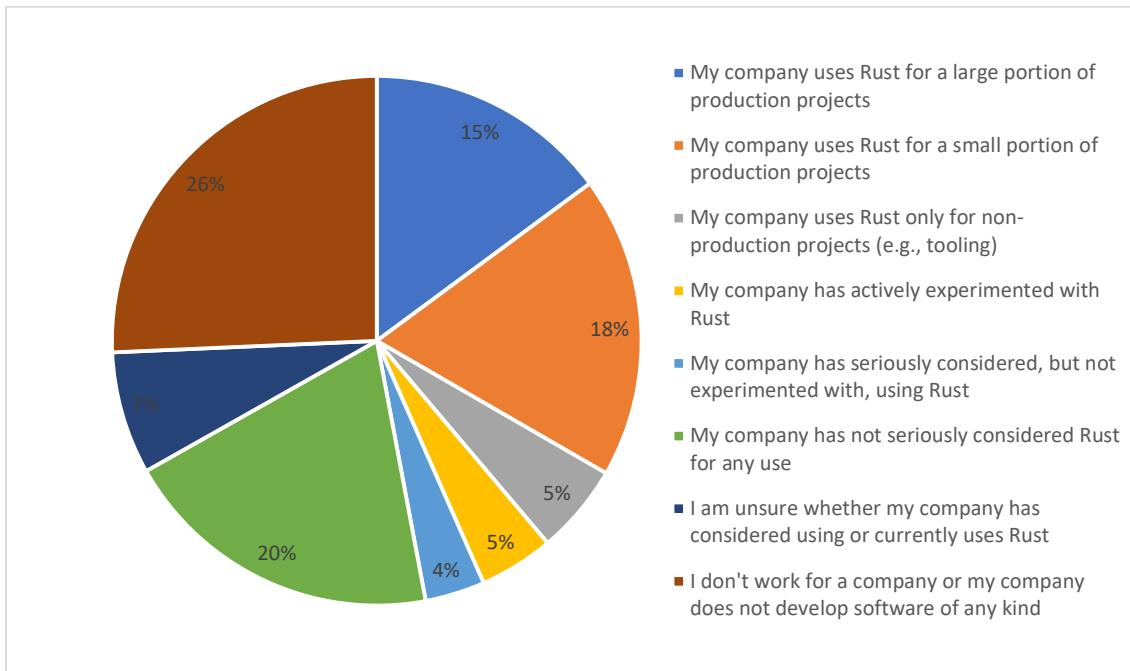


### Building a non-trivial project in Rust or contributing to an open source project



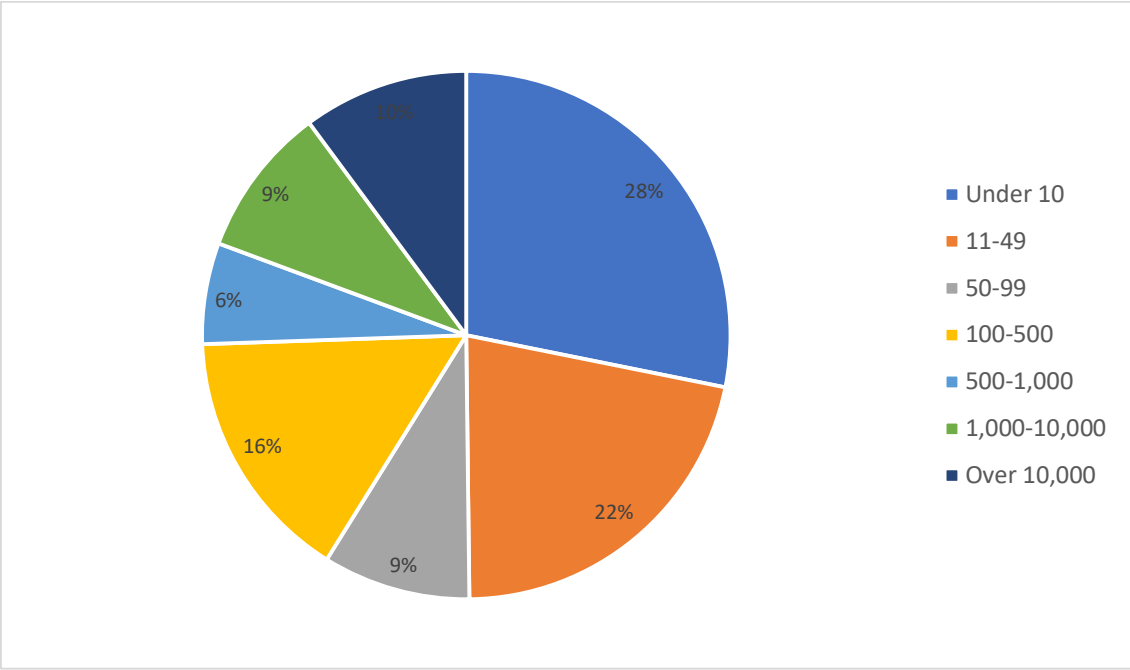
To what extent is Rust currently being used by your company?

My company uses Rust for a large portion of production projects	1243	14.9%
My company uses Rust for a small portion of production projects	1542	18.5%
My company uses Rust only for non-production projects (e.g., tooling)	459	5.5%
My company has actively experimented with Rust	380	4.5%
My company has seriously considered, but not experimented with, using Rust	306	3.7%
My company has not seriously considered Rust for any use	1652	19.8%
I am unsure whether my company has considered using or currently uses Rust	626	7.5%
I don't work for a company or my company does not develop software of any kind	2146	25.7%



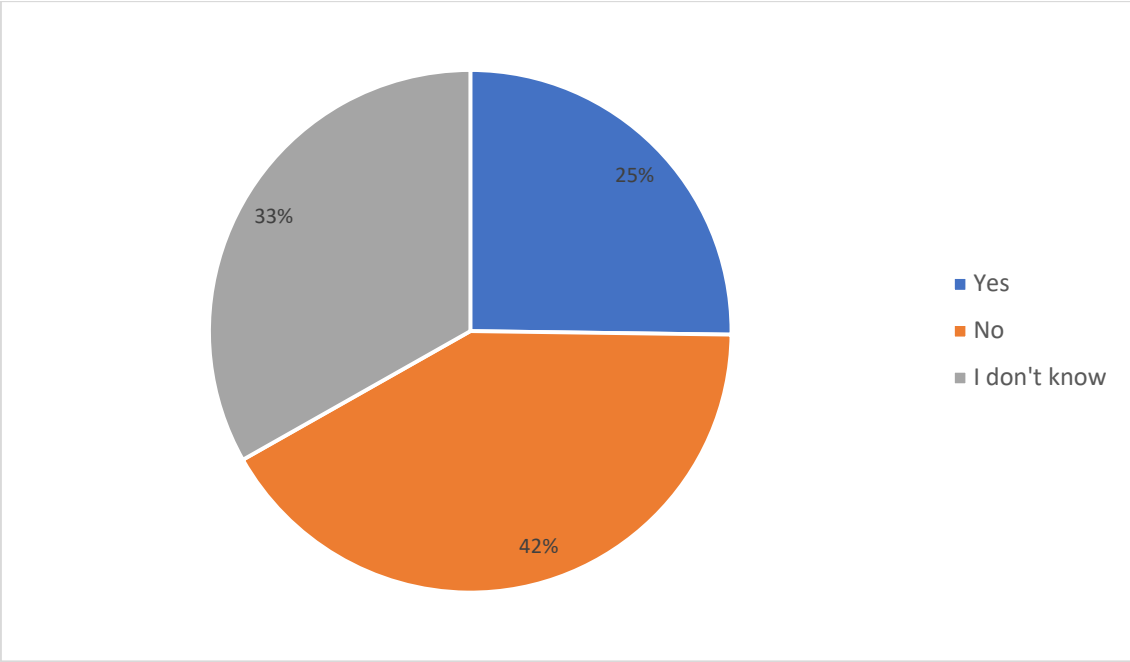
Approximately how many total developers does your company employ?

	Frequency	Percent
Under 10	1744	28.2%
11-49	1340	21.6%
50-99	560	9.0%
100-500	965	15.6%
500-1,000	384	6.2%
1,000-10,000	570	9.2%
Over 10,000	628	10.1%



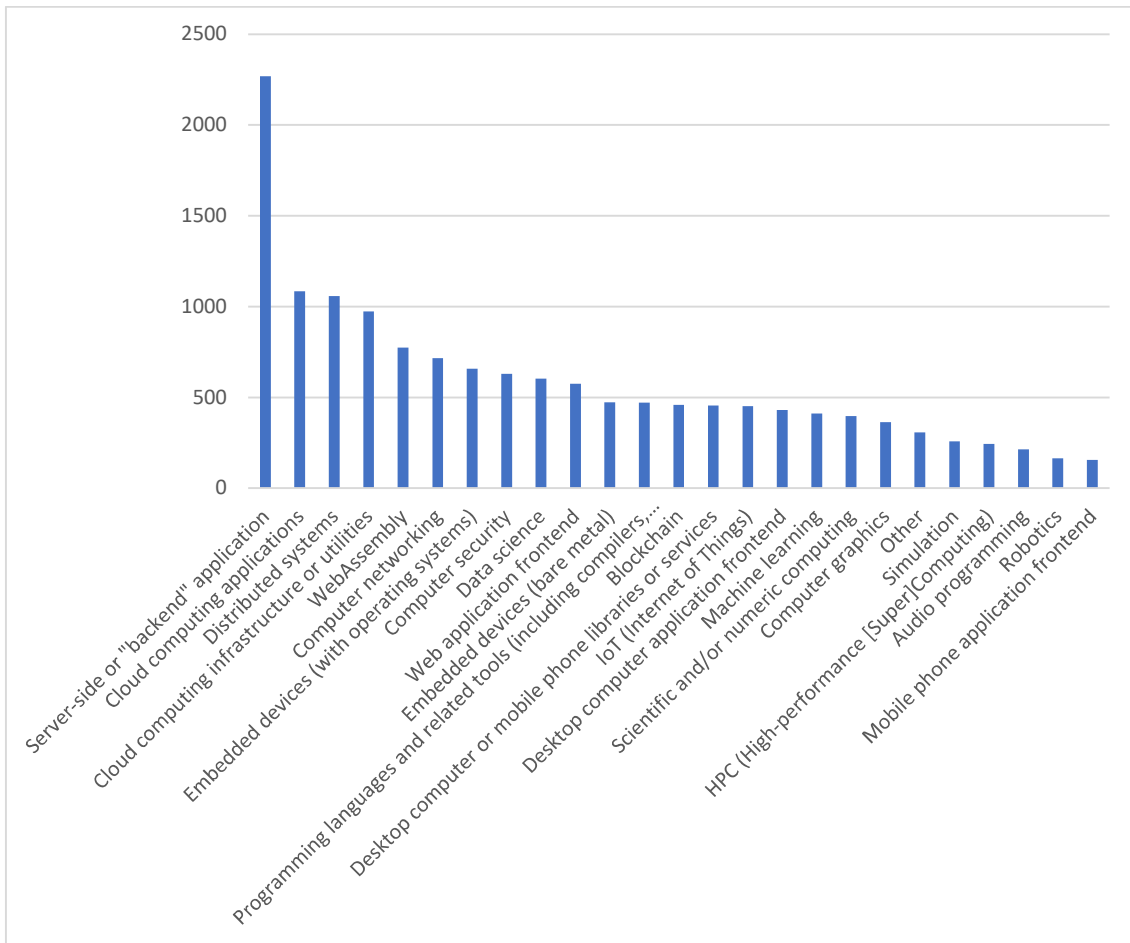
Is your company planning on hiring Rust developers in the next year?

	Frequency	Percent
Yes	1563	25.2%
No	2581	41.6%
I don't know	2058	33.2%



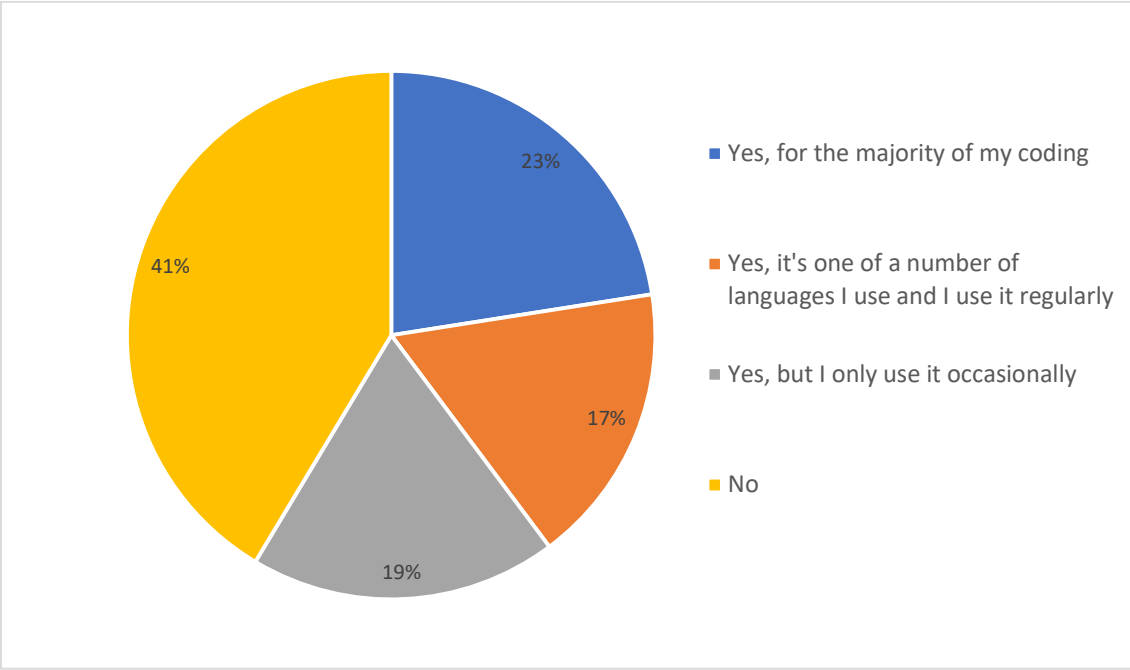
In what technology domain(s) is Rust used at your company?

Server-side or "backend" application	2269
Cloud computing applications	1085
Distributed systems	1059
Cloud computing infrastructure or utilities	973
WebAssembly	775
Computer networking	715
Embedded devices (with operating systems)	657
Computer security	629
Data science	603
Web application frontend	574
Embedded devices (bare metal)	474
Programming languages and related tools (including compilers, IDEs, standard libraries, etc.)	471
Blockchain	459
Desktop computer or mobile phone libraries or services	455
IoT (Internet of Things)	452
Desktop computer application frontend	431
Machine learning	410
Scientific and/or numeric computing	397
Computer graphics	364
Other	307
Simulation	258
HPC (High-performance [Super]Computing)	243
Audio programming	214
Robotics	164
Mobile phone application frontend	156



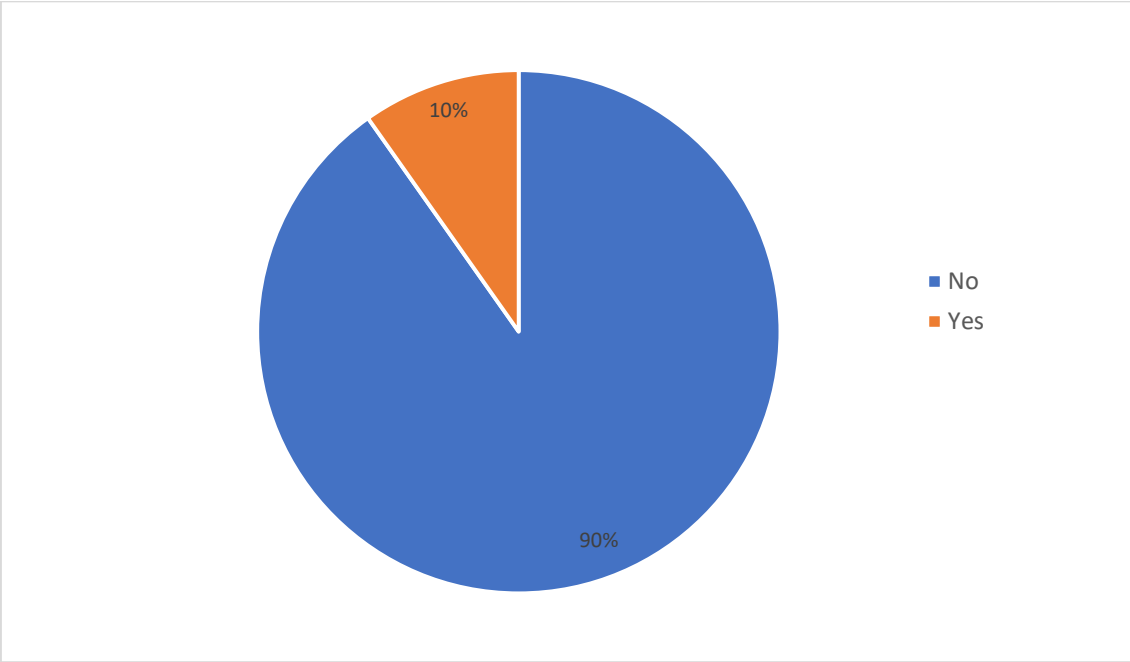
Are you using Rust at work?

	Frequency	Percent
Yes, for the majority of my coding	1401	22.5%
Yes, it's one of a number of languages I use and I use it regularly	1074	17.3%
Yes, but I only use it occasionally	1169	18.8%
No	2576	41.4%



Are you currently, or have you in the last year, taken a course or training which uses or teaches Rust?

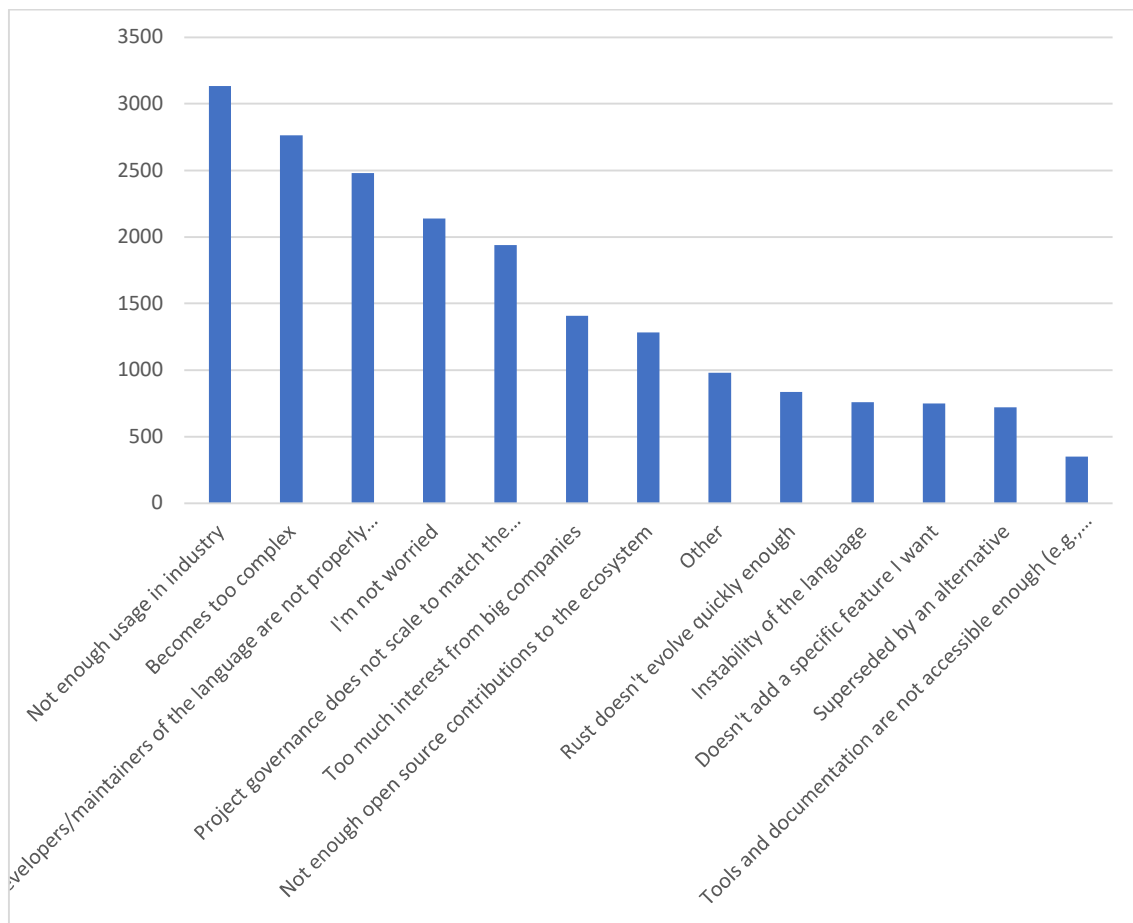
	Frequency	Percent
No	7553	90.2%
Yes	819	9.8%





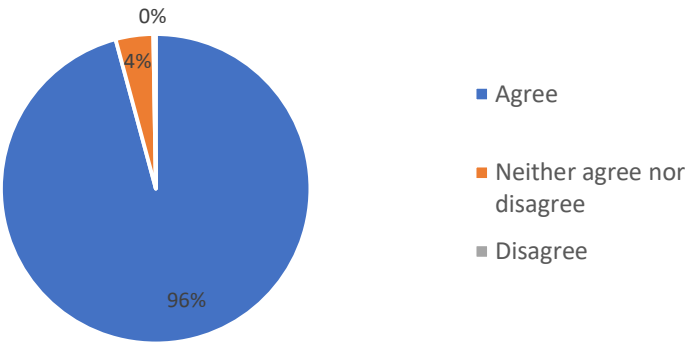
What are your biggest worries for the future of Rust?

Not enough usage in industry	3133
Becomes too complex	2763
Developers/maintainers of the language are not properly supported	2479
I'm not worried	2138
Project governance does not scale to match the size/requirements of the community	1938
Too much interest from big companies	1409
Not enough open source contributions to the ecosystem	1284
Other	982
Rust doesn't evolve quickly enough	837
Instability of the language	760
Doesn't add a specific feature I want	751
Superseded by an alternative	721
Tools and documentation are not accessible enough (e.g., due to language or incompatibility with screen readers)	351

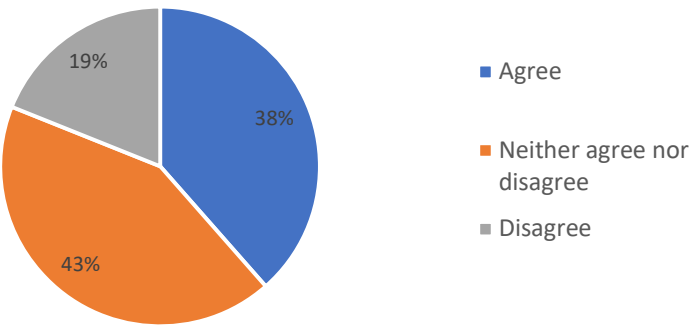


Please rate your agreement with the following statements about Rust

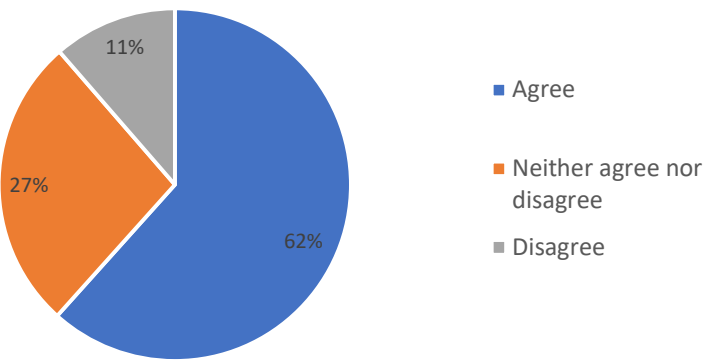
Rust provides a real benefit over other programming languages



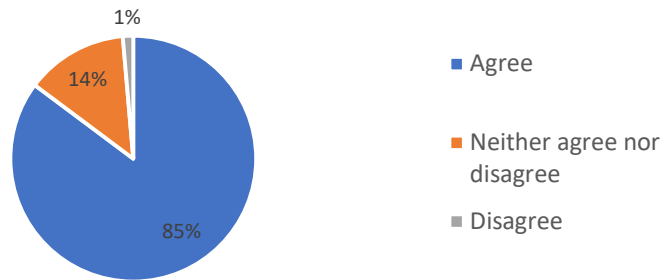
Rust is significantly more complicated to program in than other programming languages



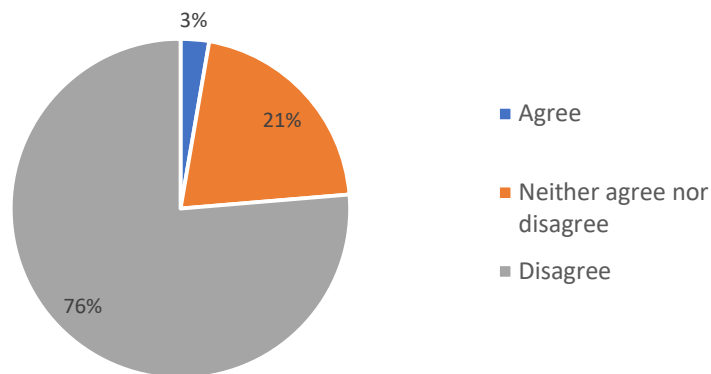
Rust requires significantly more effort to learn than other programming languages



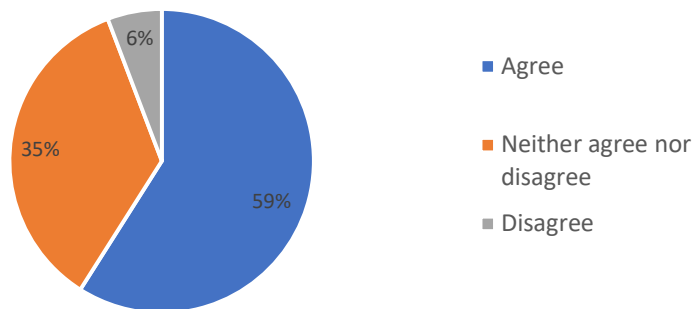
Rust code tends to contain significantly fewer bugs than equivalent code written in another programming language would have



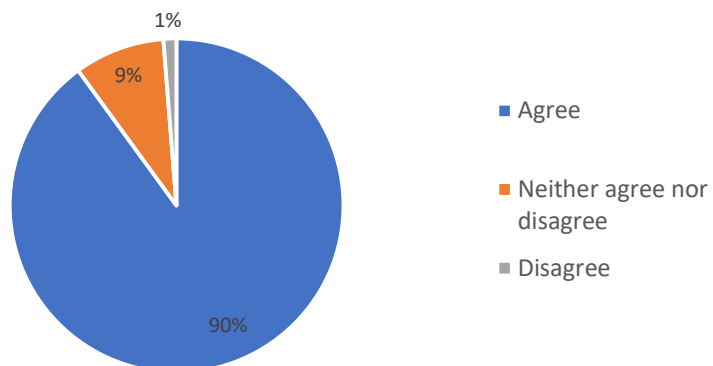
Rust is risky to use in production



Rust makes me more productive

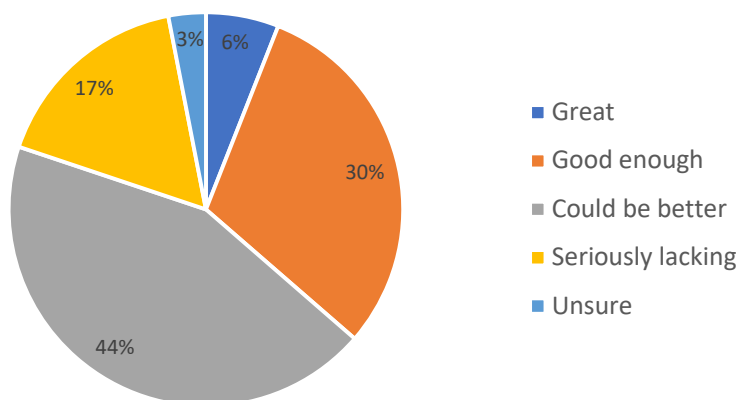


### Rust is fun to use

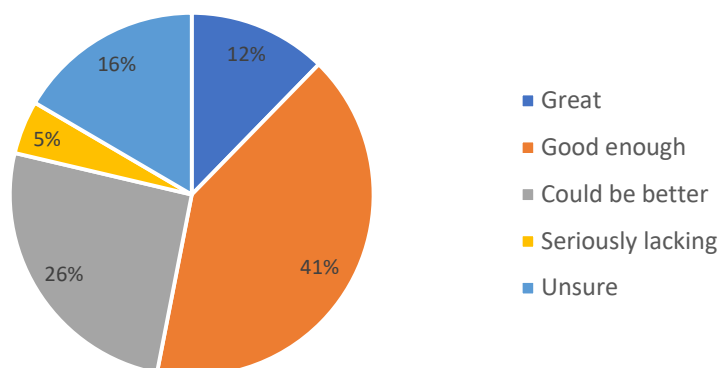


In your opinion, how do you find the following aspects of Rust?

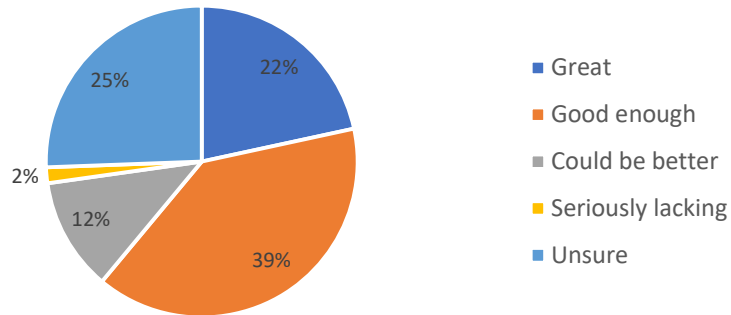
### Compile times



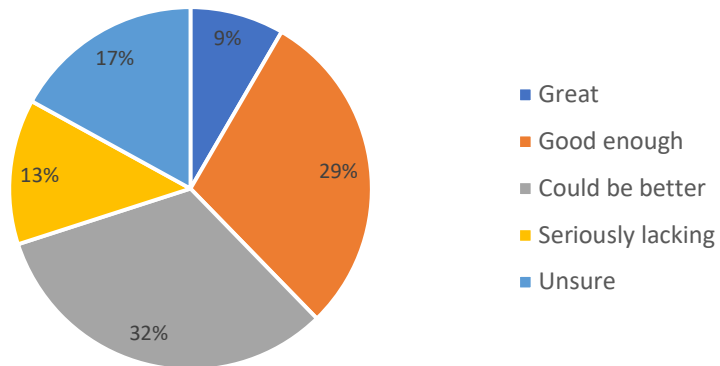
### Binary size



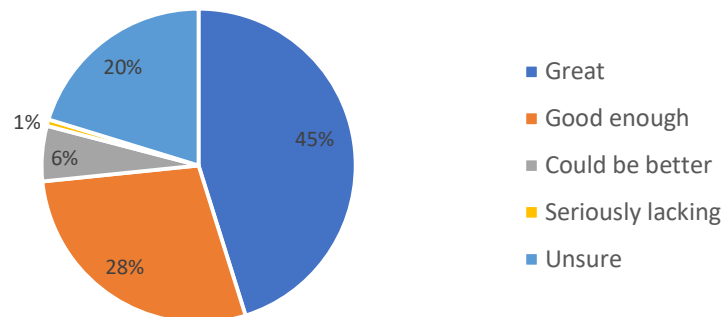
Memory usage (i.e., how much RAM rustc uses when compiling)



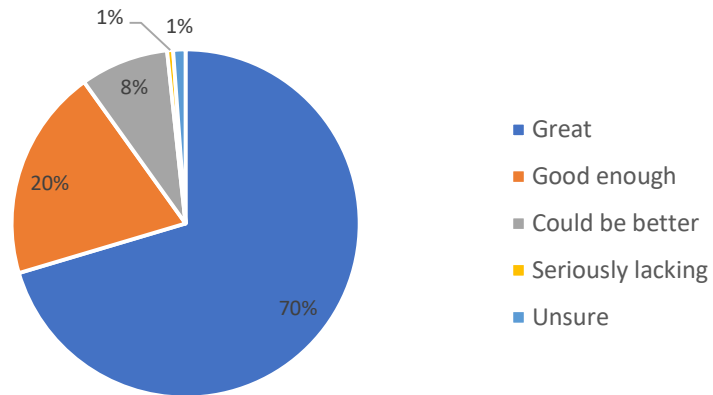
Disk space usage (e.g., the size of target folder)



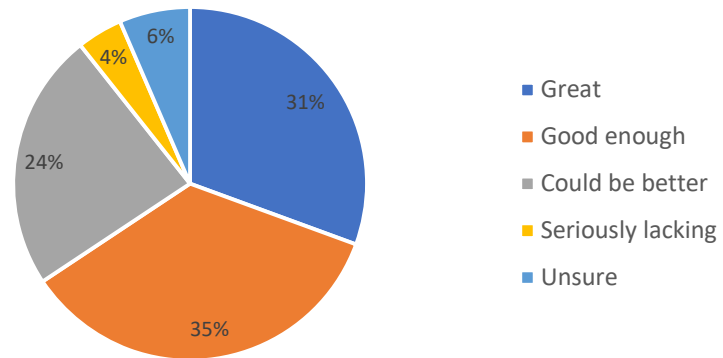
Bugs in the compiler (i.e., ICEs a.k.a. internal compiler errors, miscompilations, etc.)



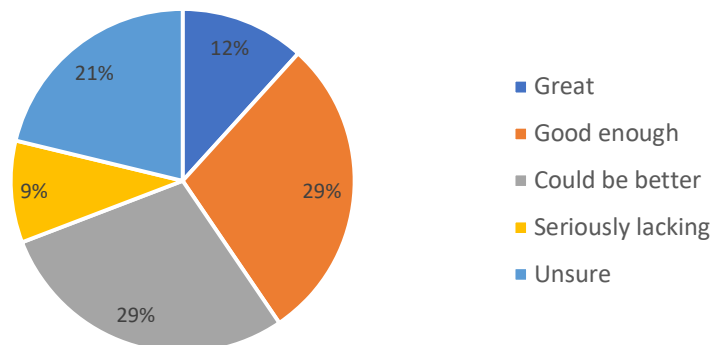
Compiler error messages



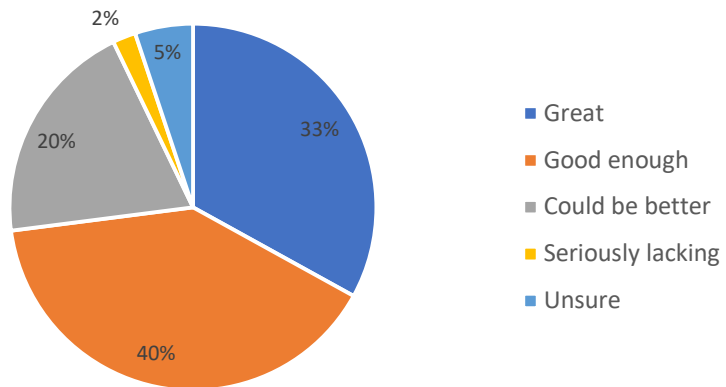
IDE experience



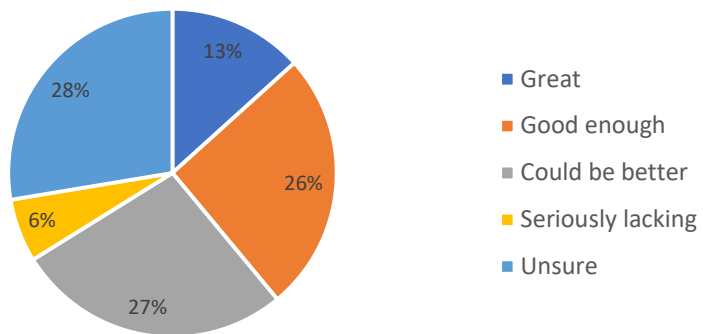
Debugging experience



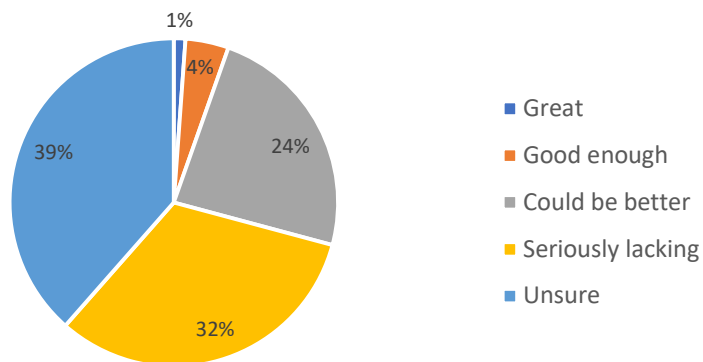
### Available tools and support



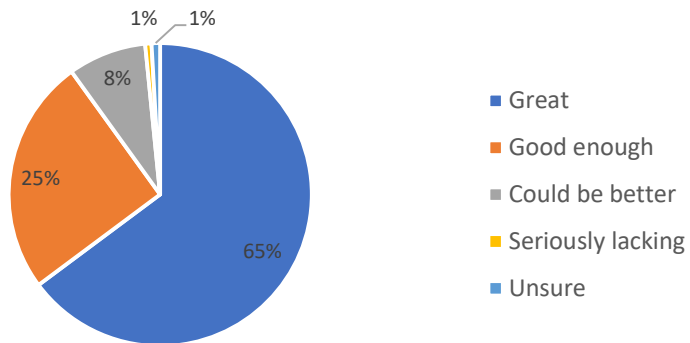
### Async programming



### GUI development

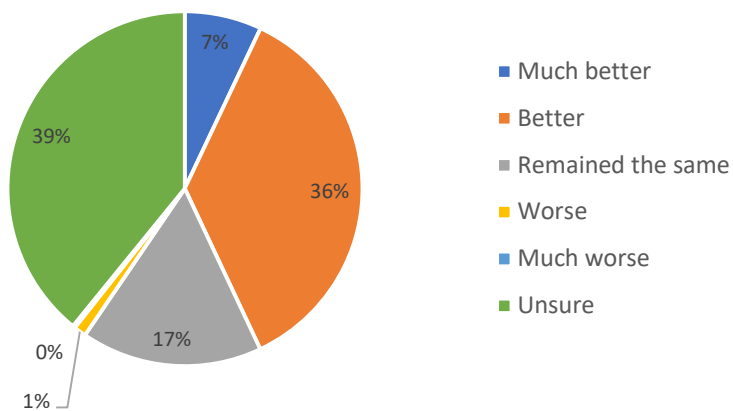


### Rust language and standard library documentation

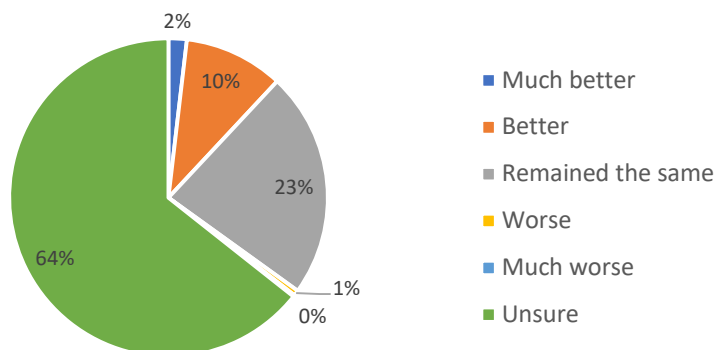


In your opinion, have the following aspects of Rust gotten better or worse over the past year?

### Compile times

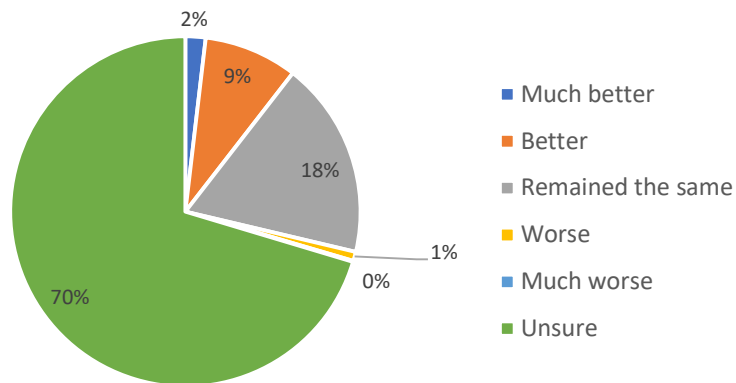


### Binary size

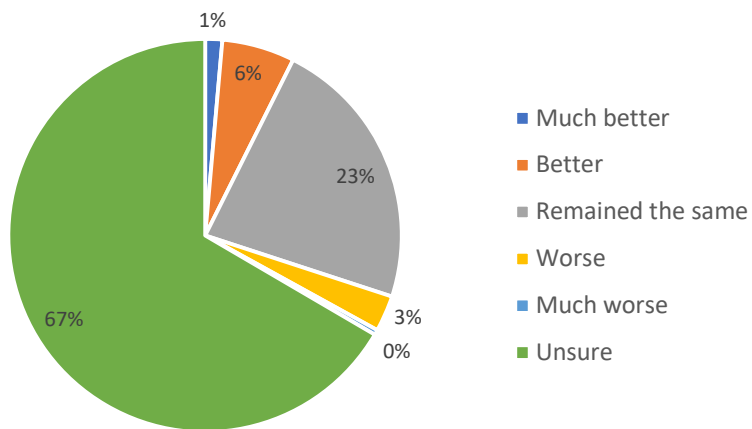




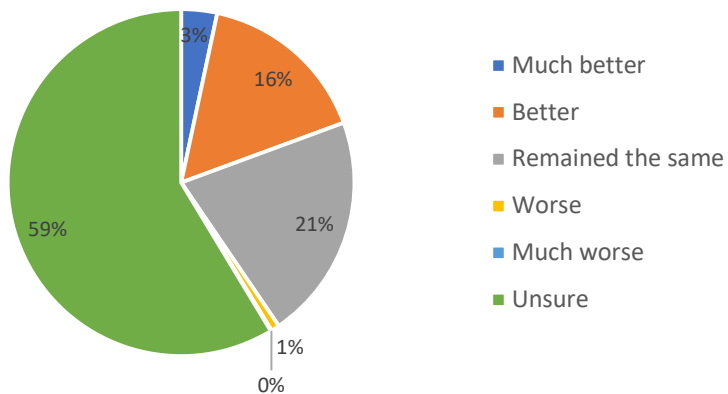
Memory usage (i.e., how much RAM rustc uses when compiling)



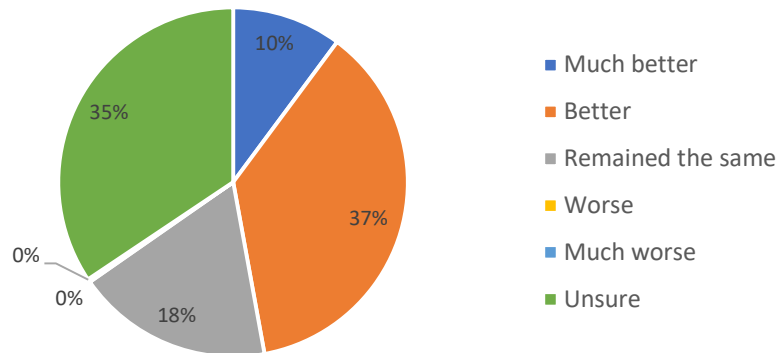
Disk space usage (e.g., the size of target folder)



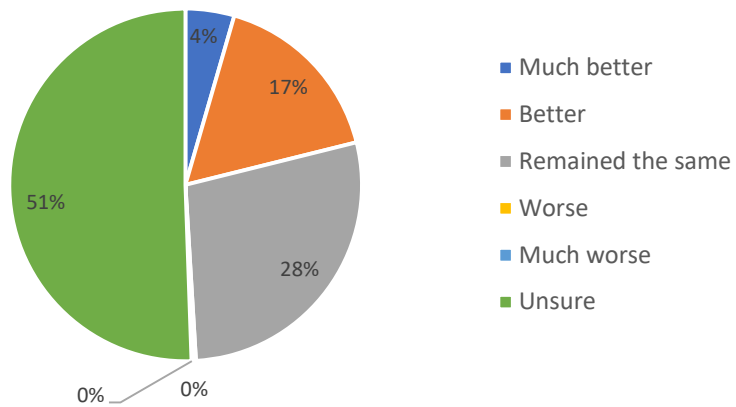
Bugs in the compiler (i.e., ICEs a.k.a. internal compiler errors, miscompilations, etc.)



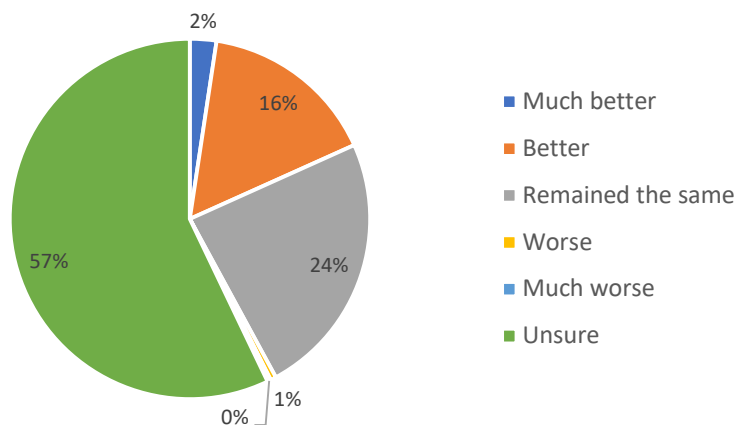
Compiler error messages



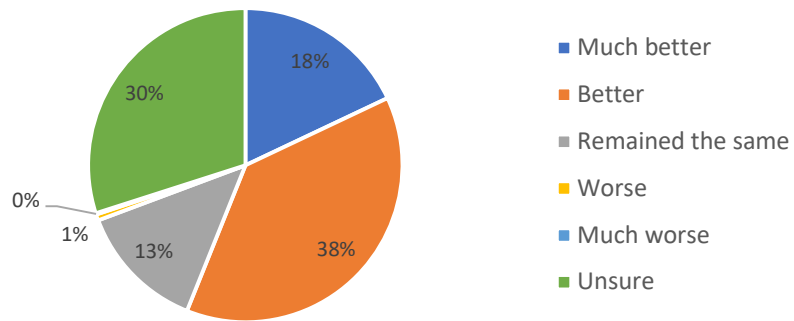
Debugging experience



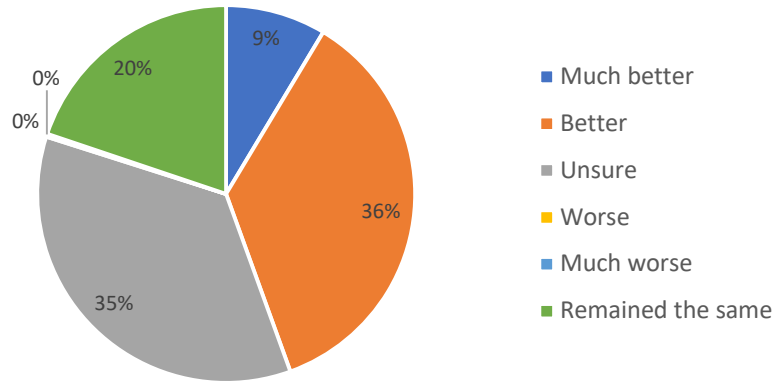
GUI development



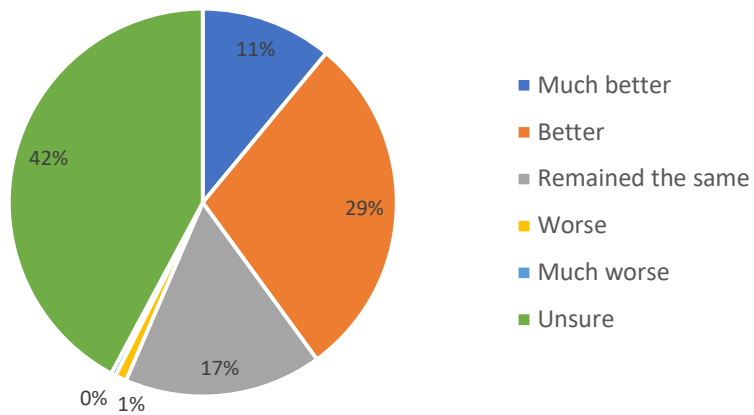
### IDE experience



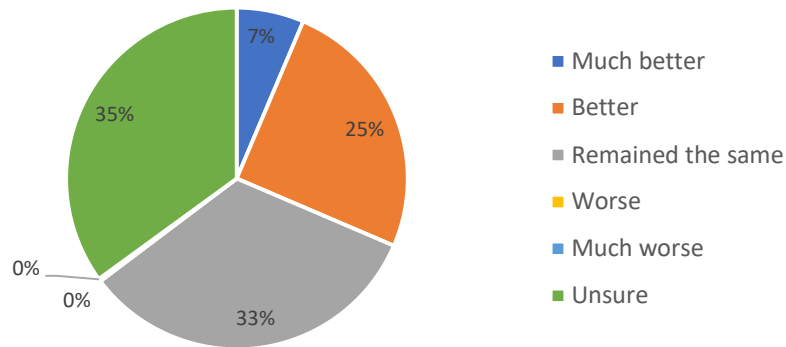
### Available tools and support



### Async programming

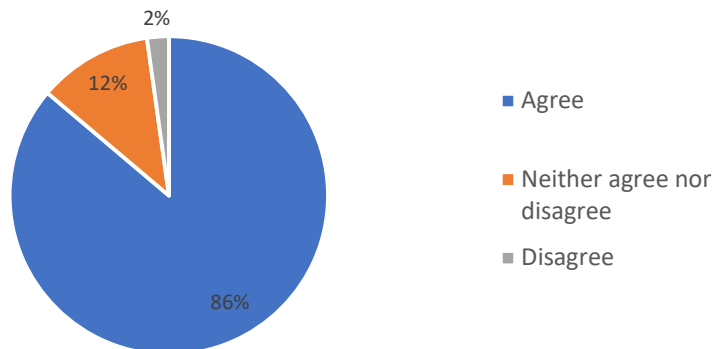


### Rust language and standard library documentation

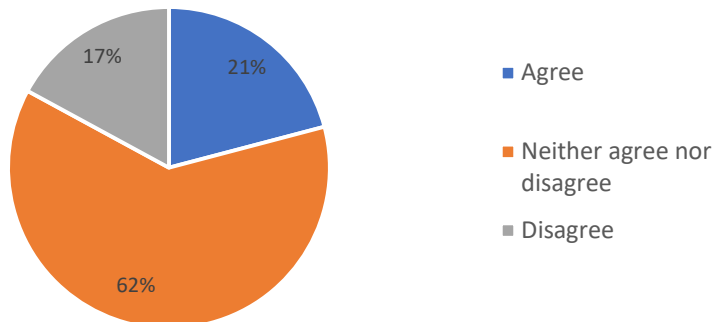


Do you agree with the following statements on Rust stability?

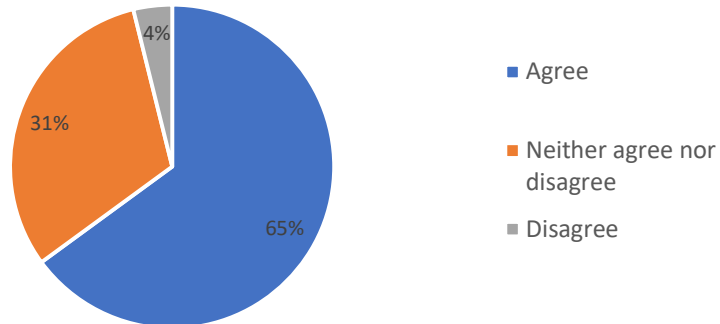
I can upgrade the stable compiler version without fear of my code failing to compile



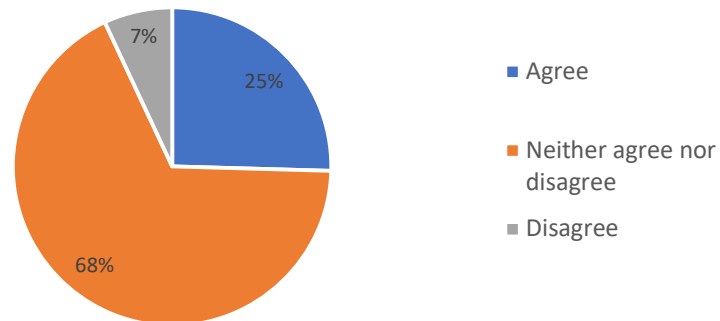
I can upgrade the nightly compiler version without fear of my code failing to compile



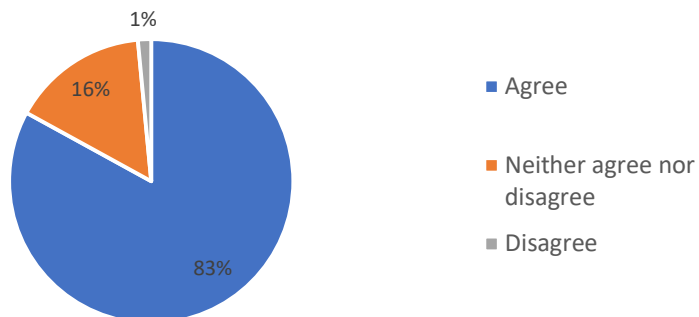
I can upgrade the stable compiler version  
without fear of my code taking longer to compile



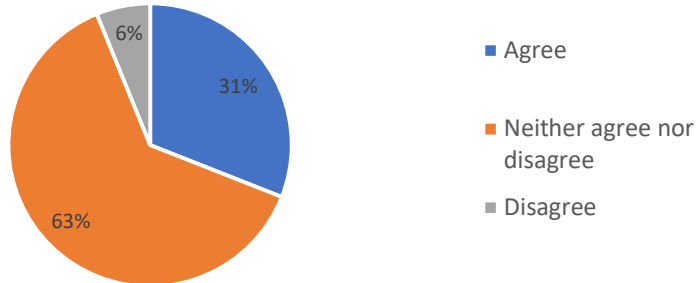
I can upgrade the nightly compiler version  
without fear of my code taking longer to compile



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

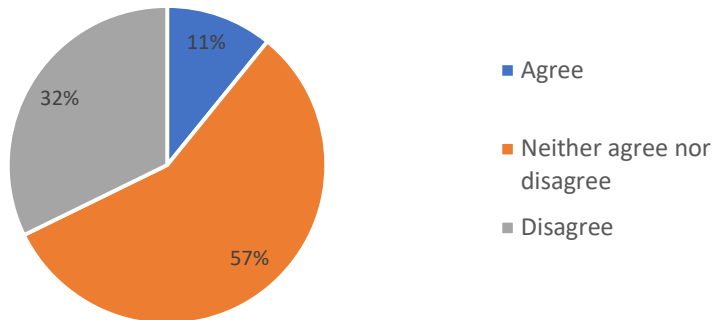


Upgrading to a new nightly compiler version  
requires either no changes or extremely small  
& easy changes to my code

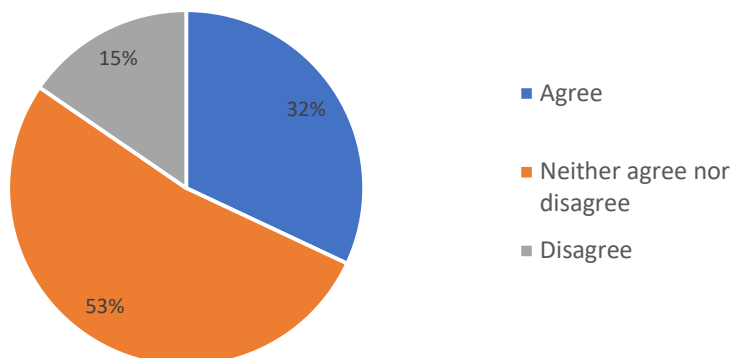


Do you agree with the following statements on Rust employment?

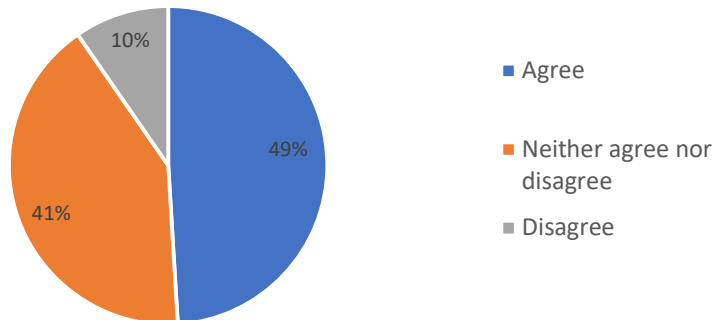
It is easy for qualified applicants to find jobs  
which use Rust for the majority of programming



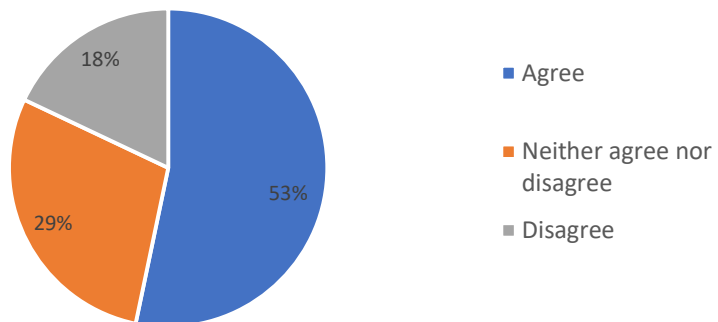
Existing Rust jobs are attractive



Learning Rust provides me with skills that employers seek



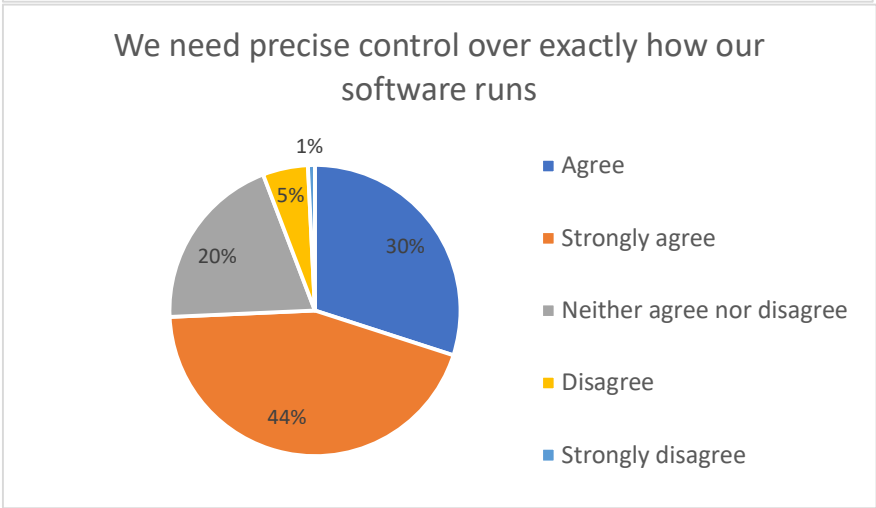
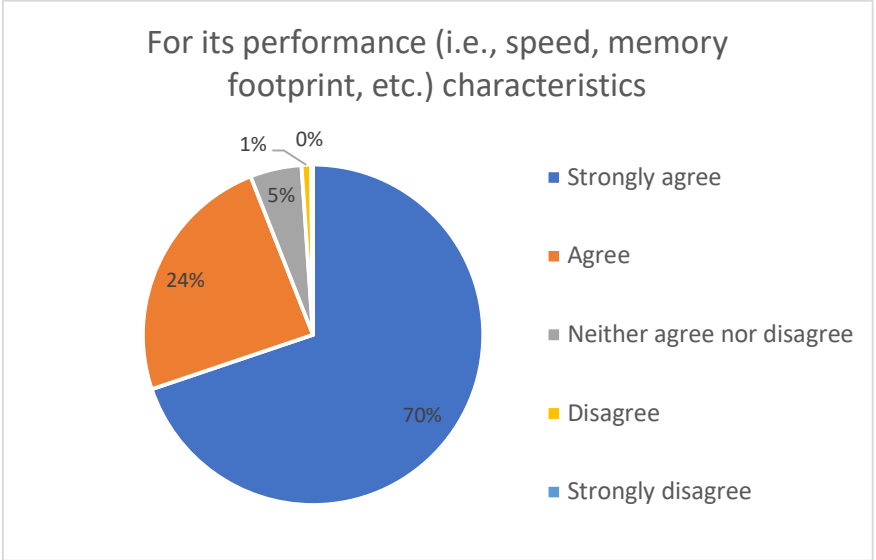
I feel qualified to apply for at least some advertised Rust jobs



# Questions for Rust users who use Rust at work

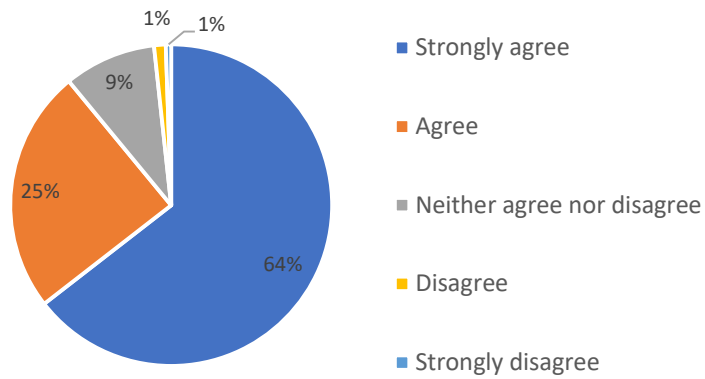
These questions were asked of those respondents who answered 'yes' to the question 'Are you using Rust at work?'.

Rate how much the following statements are reasons why your team uses Rust at work

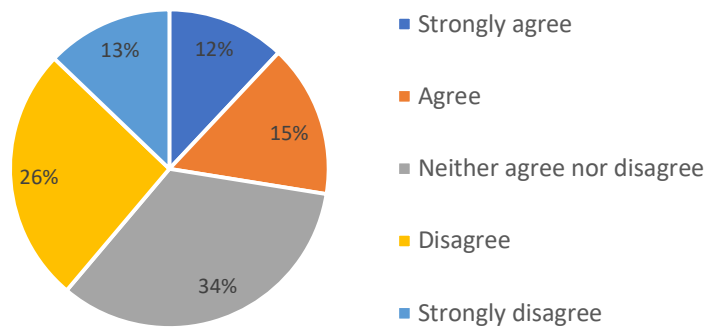




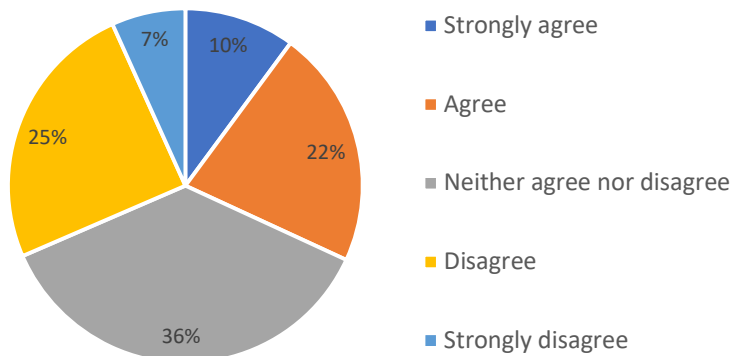
Its security and safety properties are important to us



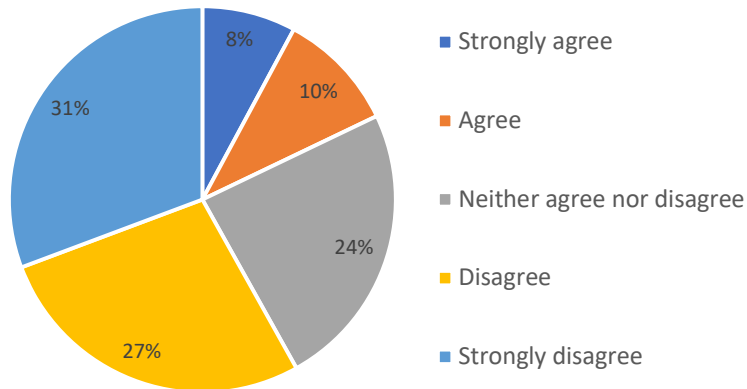
We already know Rust so it's our default choice



We find it easy to prototype with

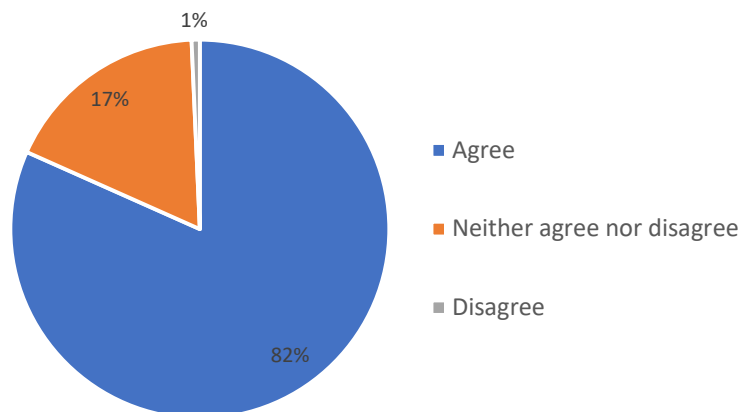


### We must interact with existing Rust code

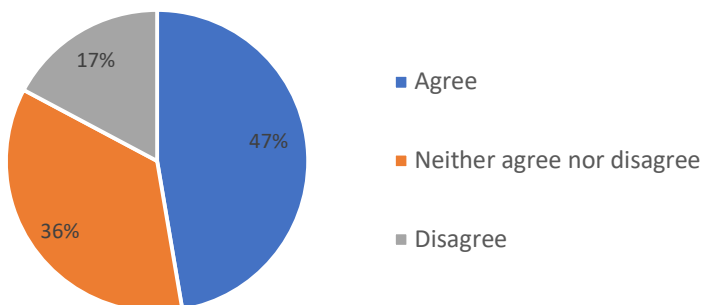


Please rate your agreement with the following statements regarding your team's experience using Rust at work.

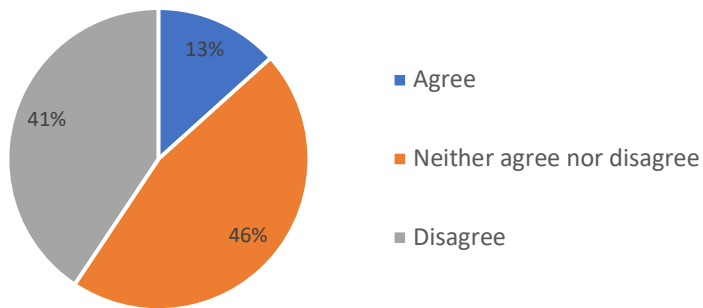
### Using Rust has helped us achieve our goals



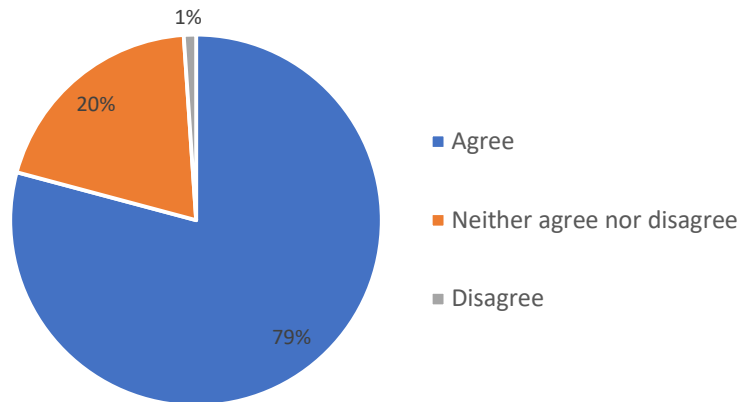
### Adopting Rust has been challenging



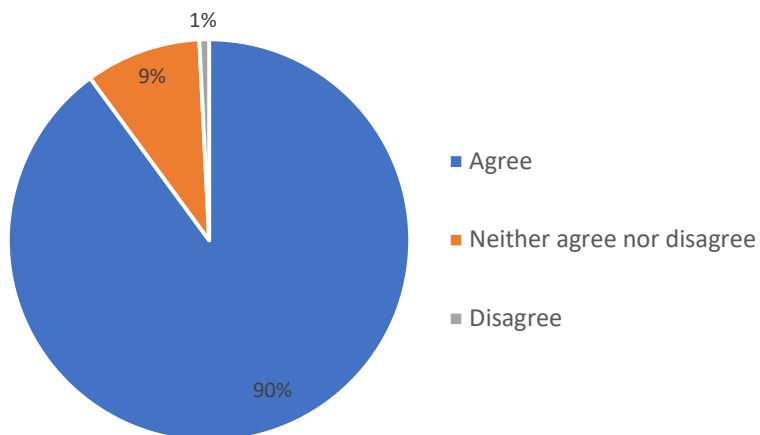
Overall, adopting Rust has slowed down our team



Using Rust has been worth the cost of adoption



We're likely to use Rust again in the future



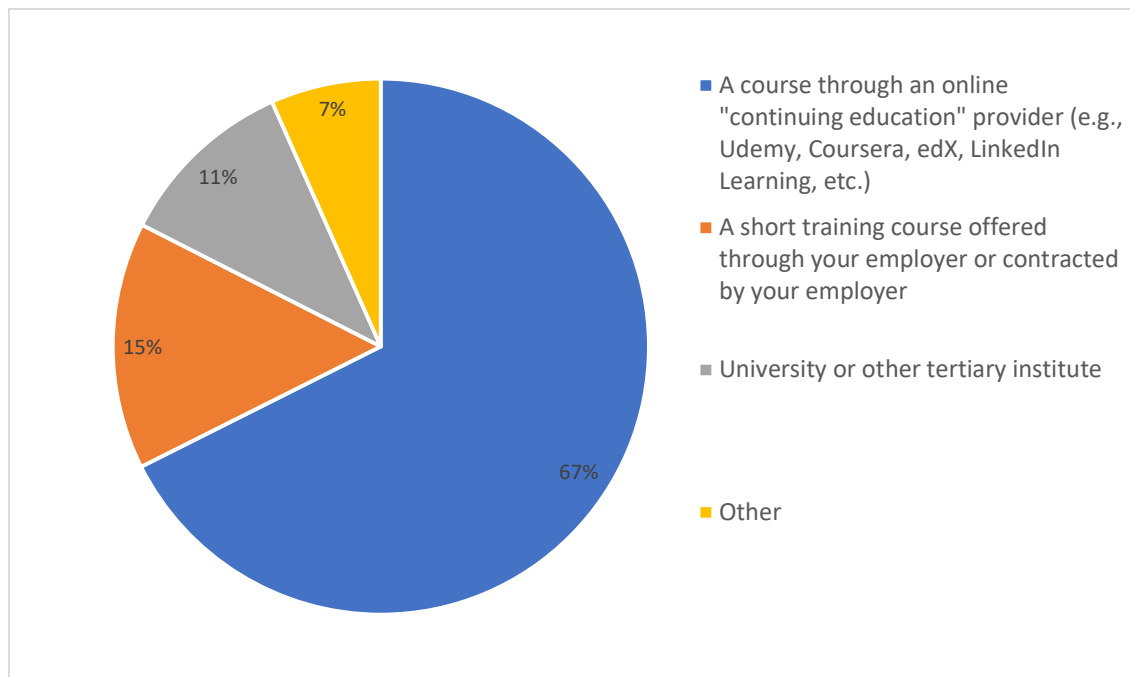
## Questions for Rust users who are taking a course or training which uses/teaches Rust

These questions were asked of those respondents who answered 'yes' to the question 'Are you currently, or have you in the last year, taken a course or training which uses or teaches Rust?'.

Where is/was the course or activity taught?

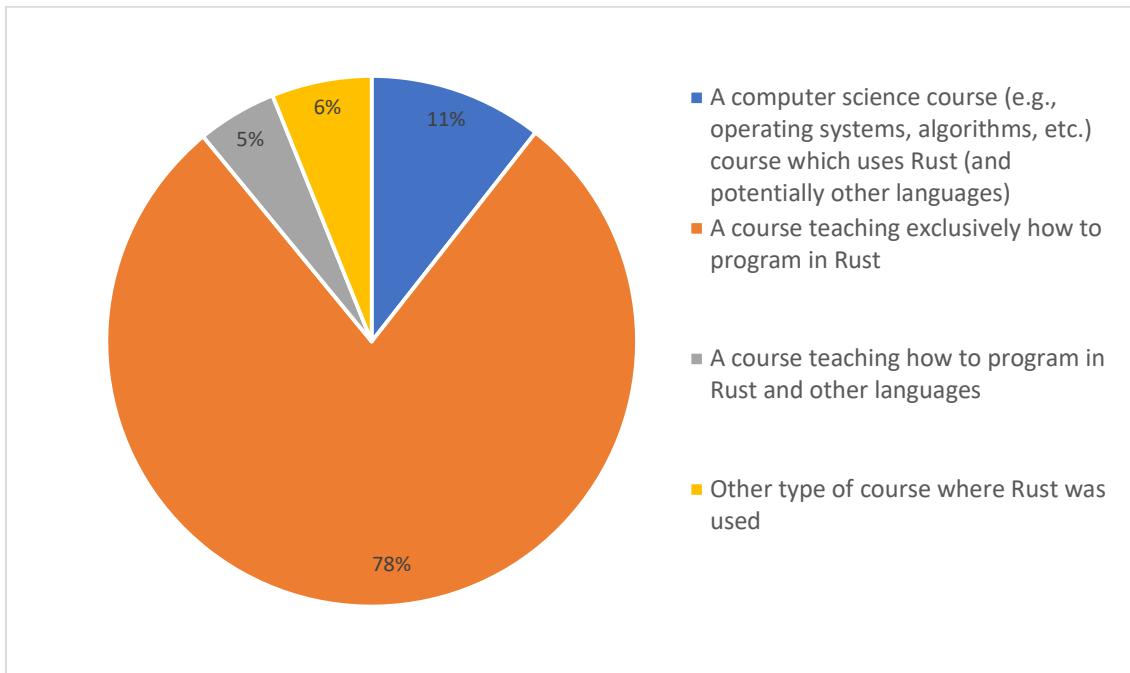
	Frequency	Percent
A course through an online "continuing education" provider (e.g., Udemy, Coursera, edX, LinkedIn Learning, etc.)	549	67.6%
A short training course offered through your employer or contracted by your employer	121	14.9%
University or other tertiary institute	88	10.8%
Other*	54	6.7%

\* 'other' here subsumes 'A bootcamp or other vocational-focused educational institute' and 'High school or secondary school'.



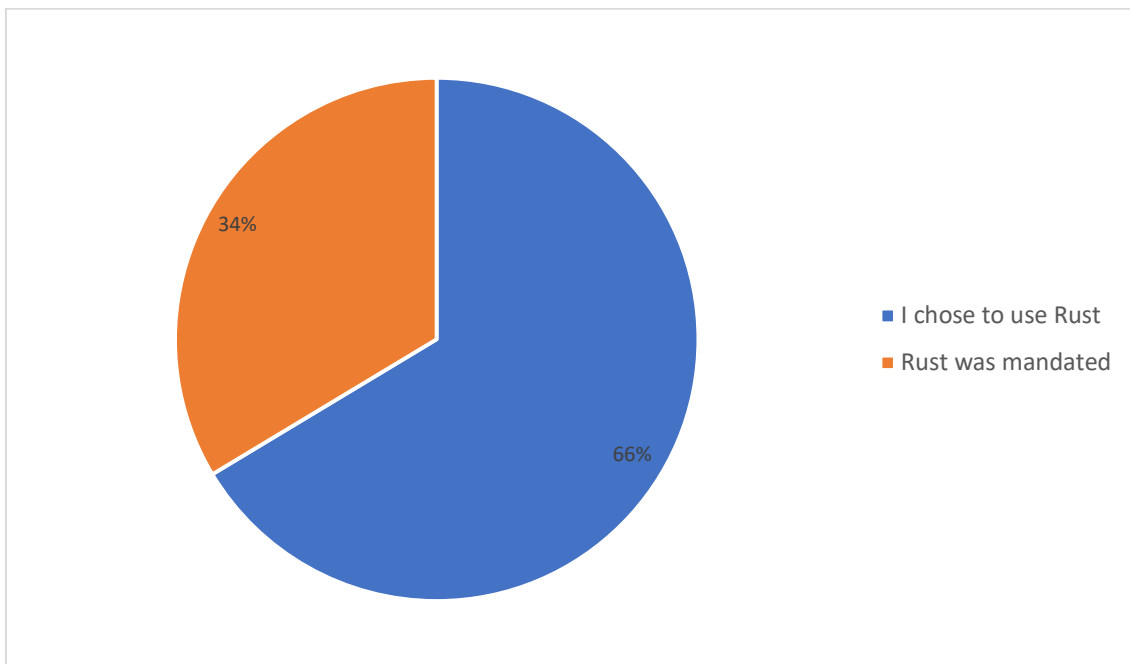
Which best describes your course or activity?

	Frequency	Percent
A computer science course (e.g., operating systems, algorithms, etc.) course which uses Rust (and potentially other languages)	85	10.6%
A course teaching exclusively how to program in Rust	630	78.5%
A course teaching how to program in Rust and other languages	39	4.9%
Other type of course where Rust was used	49	6.1%



Is/was Rust mandated for your course or activity, or did you choose it yourself?

	Frequency	Percent
I chose to use Rust	539	66.4%
Rust was mandated	273	33.6%

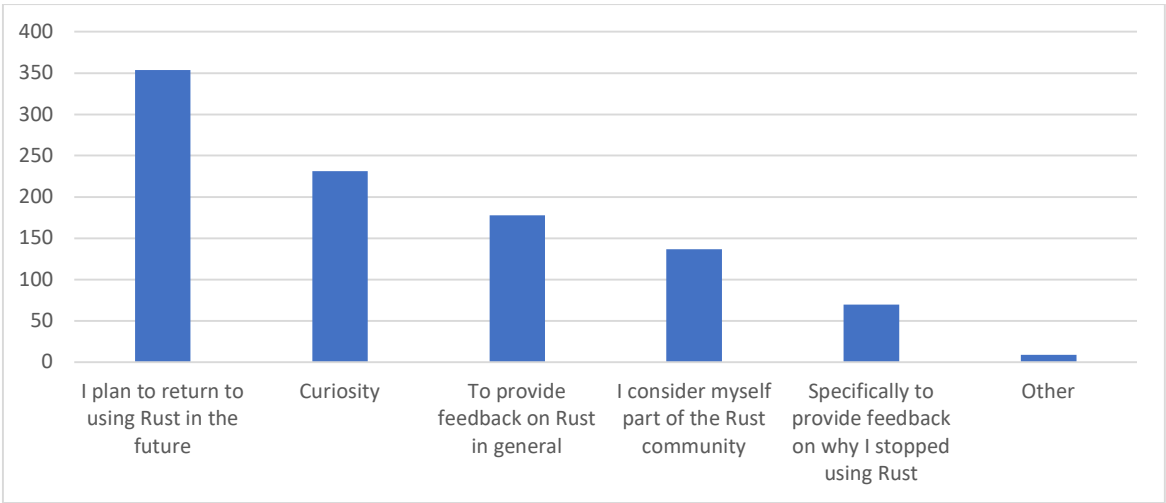


### Question for former Rust users

This question was asked of respondents who answered ‘no, I don't currently use Rust, but I have in the past’ to the first question.

As you have indicated that you're no longer using Rust, what prompted you to participate in this survey?

I plan to return to using Rust in the future	354
Curiosity	231
To provide feedback on Rust in general	178
I consider myself part of the Rust community	137
Specifically to provide feedback on why I stopped using Rust	70
Other	9

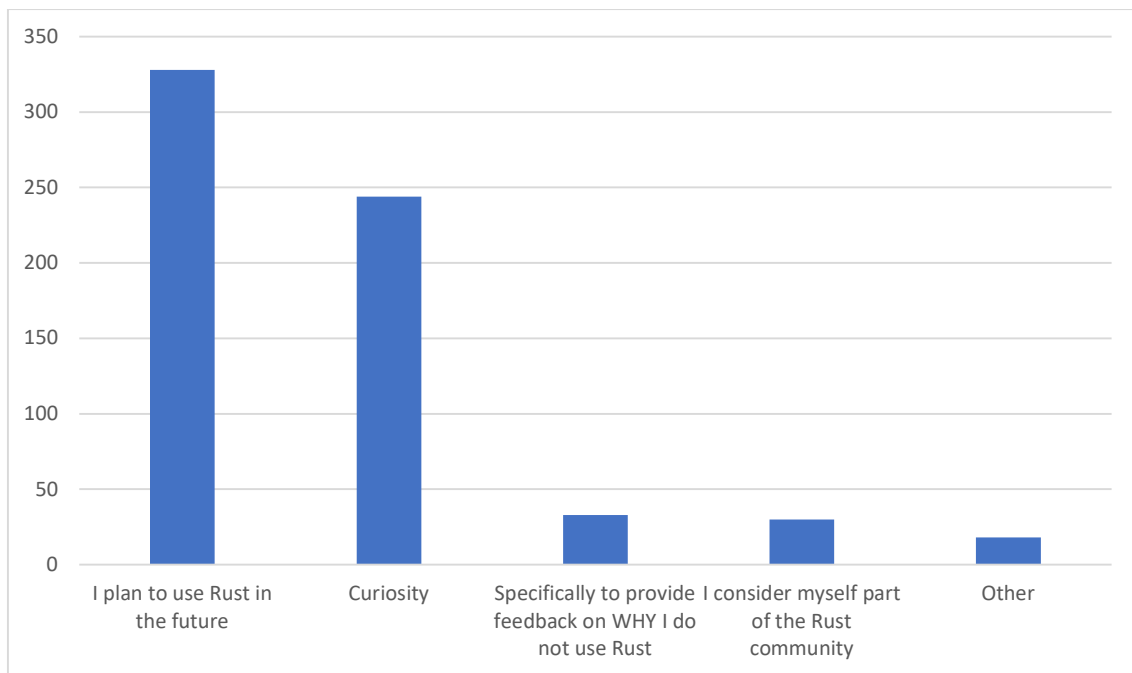


## Question for non-Rust users

This question was asked of respondents who answered 'no, I have never used Rust' to the first question.

As you have indicated that you have not used Rust, what prompted you to participate in this survey?

I plan to use Rust in the future	328
Curiosity	244
Specifically to provide feedback on WHY I do not use Rust	33
I consider myself part of the Rust community	30
Other	18



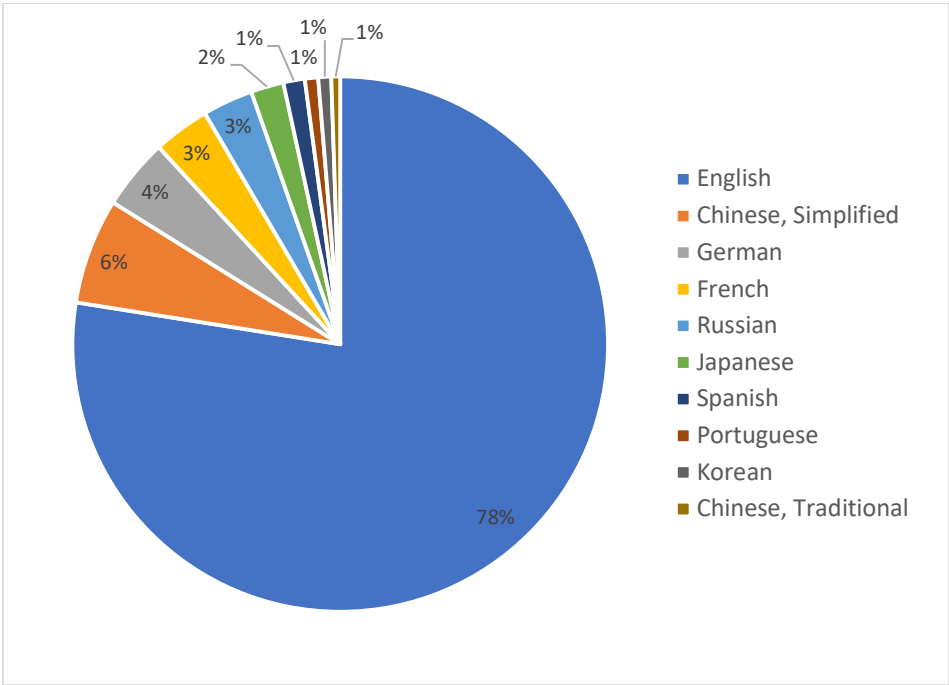
Metadata

Number of respondents

	Frequency	Percent (of viewed)	Percent (of started)
Viewed	16457	100%	
Started	12268	74.5%	100%
Completed	<b>9354</b>	<b>56.8%</b>	76.2%
Not completed	2914	17.7%	23.8%

Survey language

English	7210
Chinese, Simplified	594
German	399
French	317
Russian	283
Japanese	184
Spanish	121
Portuguese	75
Korean	73
Chinese, Traditional	49





## Elided questions

The following questions have been elided from this summary of results since the answers might be sensitive. If you're interested in these results, please get in touch with the survey working group.

- "Is there anything else you'd like to tell us?"
- "Which of the following underrepresented or marginalized groups in technology do you consider yourself a part of?"
- "Do you consider yourself a member of a group which is underrepresented in technology?"
- "Do you feel your belonging to an underrepresented or marginalized group in technology makes it difficult for you to participate in the Rust community?"
- "If you indicated that you did not feel welcome in the Rust community, are there any details about your experience that you would like to share with us?"
- Any 'other' or similar free-form answers for various questions