

# How to Ship 1000+ Commits to a Rails Monolith Daily

Hana Harencarova  
Software Developer at GitHub  
@hharen

How to Ship 1000+ Commits to a Rails Monolith Daily



Hana Harencarova

# Number guessing game

200+

500/2'500

8

18

30

1752

# Number guessing game

- **8** - Lowest number of commits last year, on 25th of December 2024
- **18** - deploys groups yesterday (11th of December)
- **30** - max PRs in a merge group
- **200+** - CI checks
- **1752** - highest number of commits per day in monolith in first half a year of 2025, 26th of March
- **500/2,500** - 500 engineers merge 2,500 pull requests monthly (number from a year ago)



# Hi, I'm Hana



???

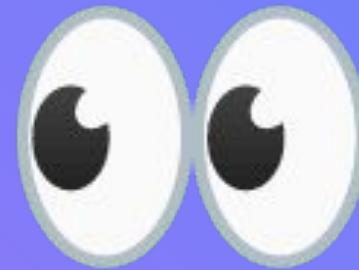
# My questions for you

*Shipping fast is not magic*  
—  
*it's process, culture, and  
tooling*

# What you'll hear today

1. Small changes for fast deploys
2. Testing
3. Deploys
4. How we shipped Autofix API

# Make changes easy to review



# Make changes easy to review

1. Ship small PRs
2. Add print screen/screen recordings of changes
3. Good PR description
4. Review your own PR

# Ship small



# Feature flags



# Local Testing & Review Lab

# Local tests (Codespaces)

- Unit and integration tests
- Running Rails server to do manual testing

# Review lab aka branch deployments

- Deploy of a specific branch
- Unique URL
- Connected to production database
- Enables testing with real data
- Time limited

# CI Checks



# CI Checks

- Static analysis (CodeQL) to check for security vulnerabilities
- Feature-flag awareness (all on)
- Different environments
- OpenAPI
- Code and service owners

# CI Checks

- Unit/integration tests
- Linter
- Sorbet
- Others, like dead code, PR size

# Deploys

# Merge queue

- Running CI
- Max 30 PRs in a group
- People are expected to watch the deploy
- Fully integrated with PR experience

# Pipeline Overview

- Deploy to canary
- Deploy to staff cloud
- 2%
- 5xx, timeouts stop
- 5%
- Production deploy
- Deploy to different cloud regions
- Merge

# Observability

## Automatic

- Detect and stop issues early
- Auto-pause and auto-rollback

## Human

- Check for problems (Datadog, Sentry)



# ChatOps

- Slack integration
- Review lab deploys
- Queue for deploy
- Posts information about deploy and exceptions
- Retries

# Rollback

- One click
- Automatic
- Deploy main
- Short version of a deploy
- Jumped and solo deploys

# First deploy, then merge

# Merge queue in practice

# Deploying

- Enqueue PR in the 
- Handovers 
- Other work to do during a deploy

# Autofix public API for historical alerts

# Autofix public API

- 
- 3 endpoints
  - Get
  - Create
  - Commit
- 10 PRs
- 2.5 weeks



How can you use what  
you learned today?

What's one thing you  
can change or  
introduce?

# Happy deploys





# you may want to read

- [Docs: Managing a merge queue](#)
- [Blog: How GitHub uses Merge Queue](#)
- [Blog: Building GitHub with Ruby and Rails](#)
- [Blog: Enabling branch deploys](#)
- [Repo: Branch deploy Action](#)
- [Blog: 5 things to ship more secure](#)

Thank you! Let's stay in touch 😊



Hana Harencarova



hharen

# Q & A



Hana Harencarova



hharen