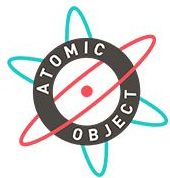


# Demystifying DevOps and CI/CD

# THANKS TO ALL OUR SPONSORS!



# "DevOps"

Image credit: <https://github.com/infrabricks/wordcloud>

 @IAmJerdog

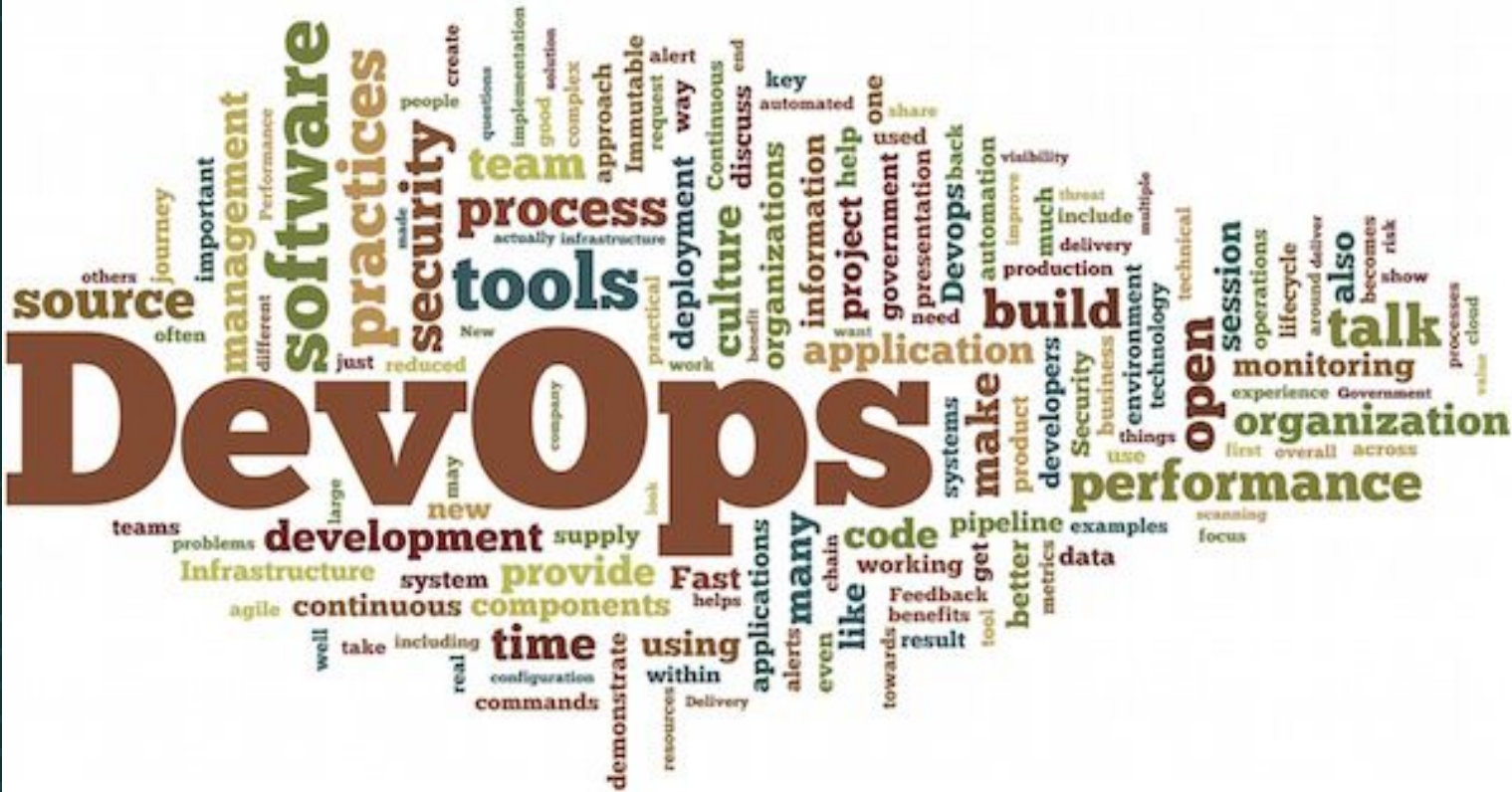


Image credit: <https://github.com/infrabricks/wordcloud>



AI Ops      DevTestOps  
TreeOps  
GitOps      MLOps  
DevSecOps

*Coming Soon(ish)*

Made with by [Creative Tim](#). Free download [here](#).



# Jeremy Meiss

Director, DevRel & Community



[timeline.jerdog.me](https://timeline.jerdog.me)



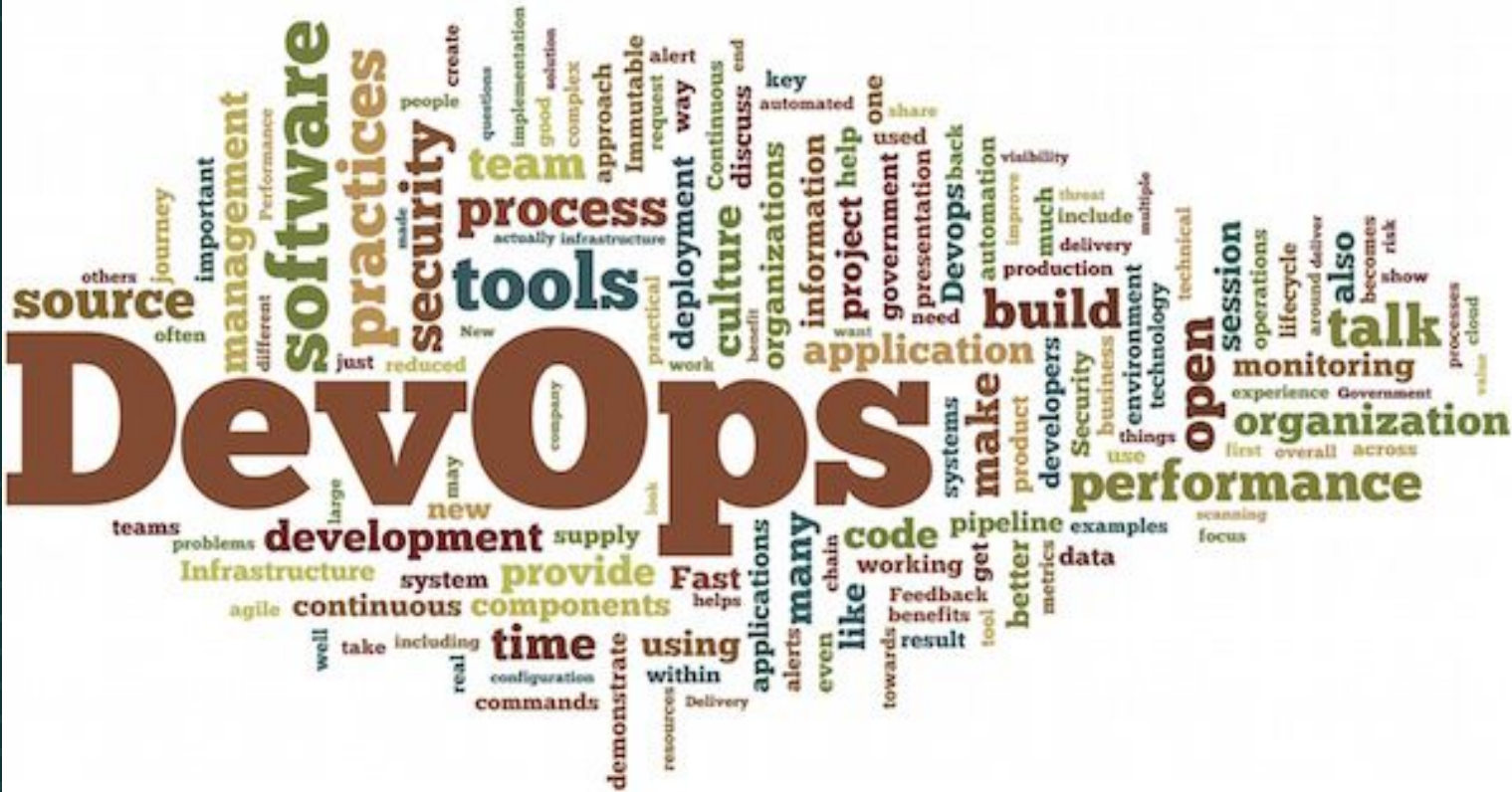


Image credit: <https://github.com/infrabricks/wordcloud>





# What is DevOps?

amazon

Hello 

Select your address

All 

devops

🇺🇸

 Hello, Sign in 

Account & Lists

Returns & Orders

0

 Cart

All Back to School Off to College Best Sellers Amazon Basics Today's Deals New Releases Customer Service Prime Music Books

1-48 of over 1,000 results for "devops" 

Sort by: Featured

Eligible for Free Shipping

☐ Free Shipping by Amazon

All customers get FREE Shipping on orders over \$25 shipped by Amazon

Delivery Day

☐ Get It Today

Prime Try Before You Buy

☐ Try Before You Buy

Department

Men's Clothing

Running Clothing

Books

Computer Programming

Information Management

Computers & Technology Industry

Women's Clothing

Men's Soccer Clothing

Cycling Clothing

[See All 8 Departments](#)

Customer Reviews

★★★★★ & Up

★★★★☆ & Up

★★★☆☆ & Up


★★☆☆☆ & Up

★☆☆☆☆ & Up

RESULTS

Price and other details may vary based on product size and color.

Best Seller



DEVOPS

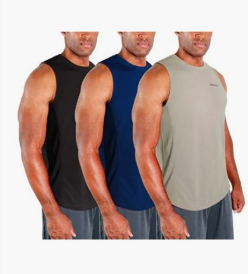
2-3 Pack Men's Athletic Long Sleeve Compression Shirts

★★★★★ ~ 14,494

\$29<sup>98</sup>

prime

Best Seller




DEVOPS

3 Pack Men's Muscle Shirts Sleeveless Dri Fit Gym Workout...

★★★★★ ~ 14,343

\$25<sup>57</sup>

prime




DEVOPS

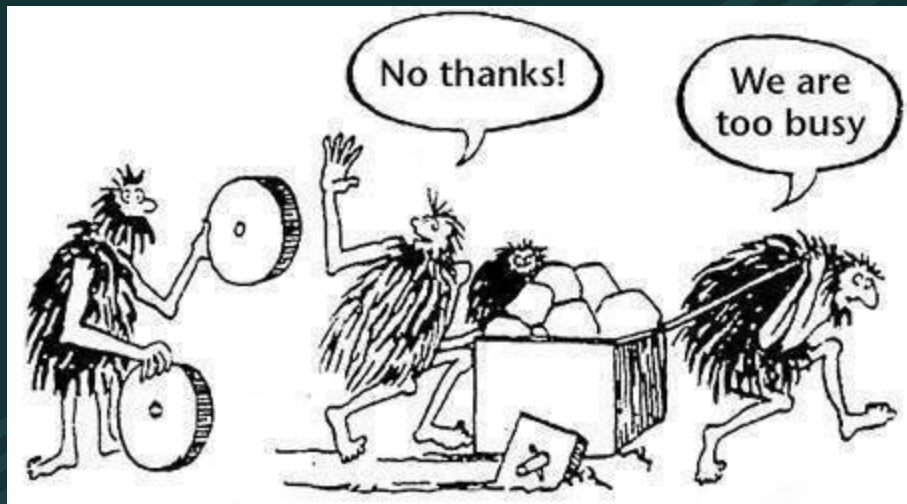
2-3 Pack Men's Athletic Compression Shirts Sleeveless

★★★★★ ~ 17,760

\$20<sup>97</sup>

prime

 @IAmJerdog



# 1910s to 1930s



"Flow production"



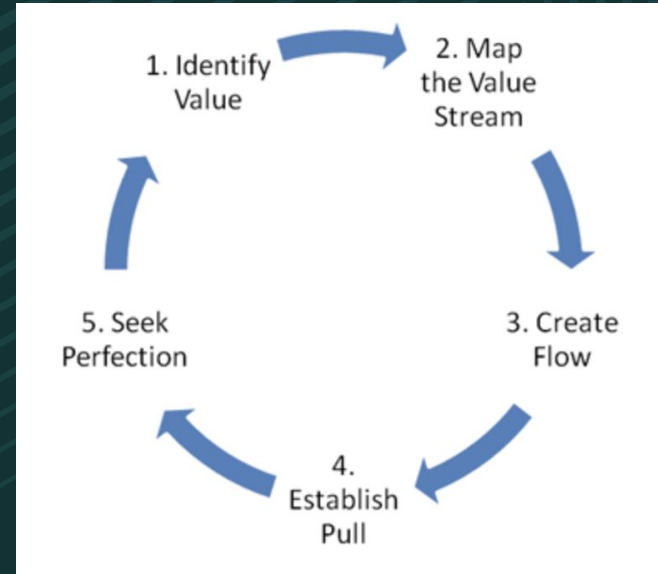
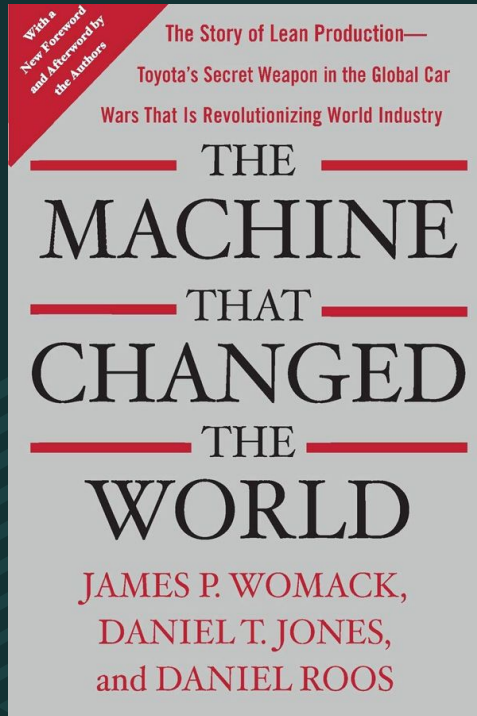
Henry Ford



Kiichiro Toyoda



# 1990s





## Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

**Individuals and interactions** over processes and tools

**Working software** over comprehensive documentation

**Customer collaboration** over contract negotiation

**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

O'REILLY

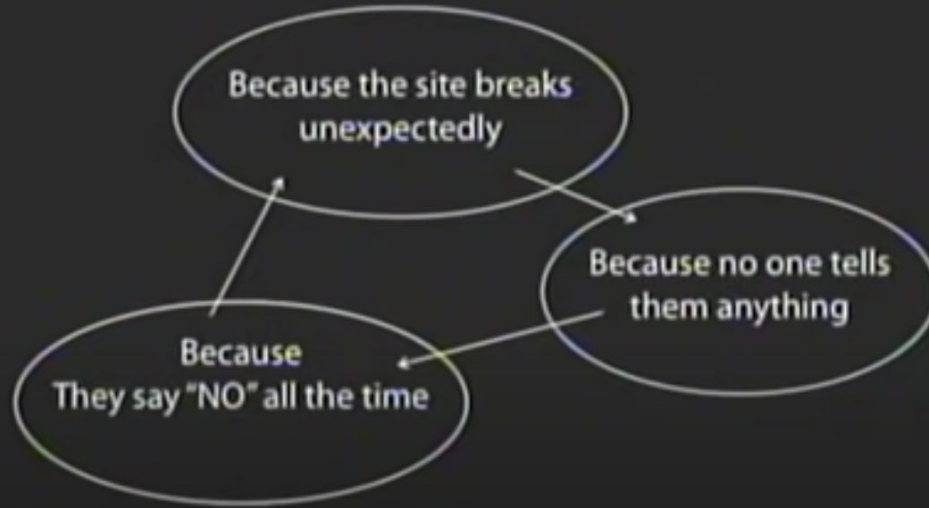
# Velocity

Web Performance and  
Operations Conference



*"10+ Deploys per Day: Dev and Ops Cooperation at Flickr"*  
- John Allspaw, Paul Hammond

# Ops stereotype









**The conference that brings development  
and operations together.**



Home



Contact



Events



Presentations



Blog

## Devopsdays Ghent 2009



**welcome program reactions speakers participants**

[Tweets from devopsdays events](#)

This is how the first devopsdays was announced:



The first devopsdays happened in Belgium - Ghent and was a great success. Have a look at the [reactions](#) is created and the [presentations](#) that were held. See you next time!





**Patrick Debois - #thinktogether make code not war**

@patrickdebois

Normal

0%

**Dev(sec)Ops: everything you do to overcome the friction created by silos ... All the rest is plain engineering**

11:24 AM · Jan 15, 2021 · Twitter for Android



matty stratton, jewish prank dinosaur 🇺🇸💜

@mattstratton

Normal 0%

You can't buy devops, but I can sell it to you.



Patrick Debois - #thinktogether make code not ... @patrickdeb... · Mar 30

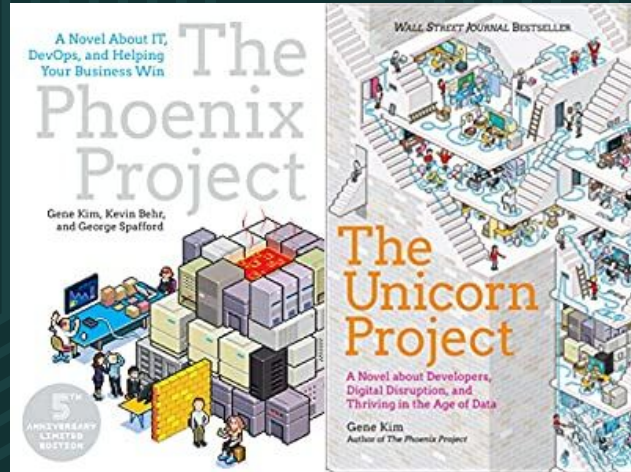
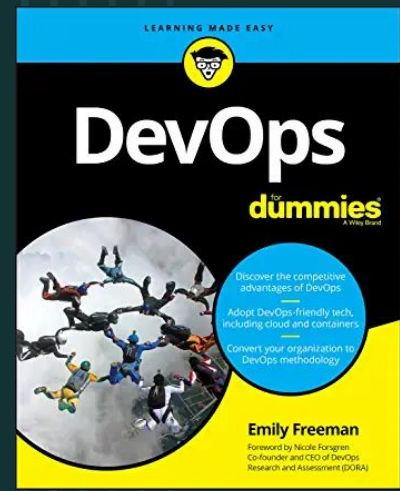
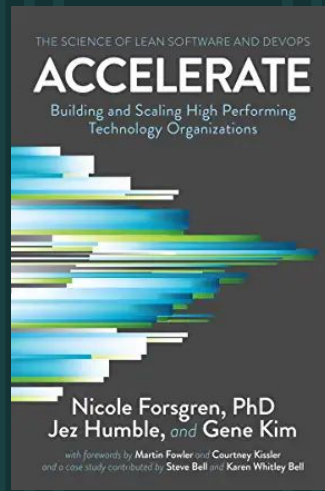
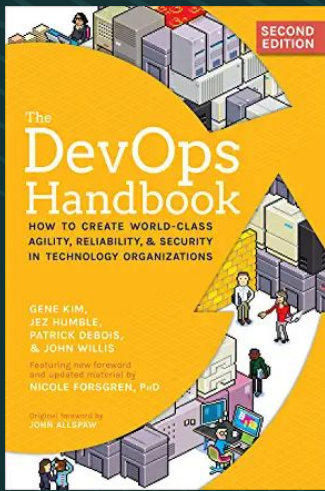
Normal 0%

fixing your devops problems with machine learning



GIF ALT

8:09 AM · Mar 30, 2022 from Lisle, IL · Twitter for iPhone





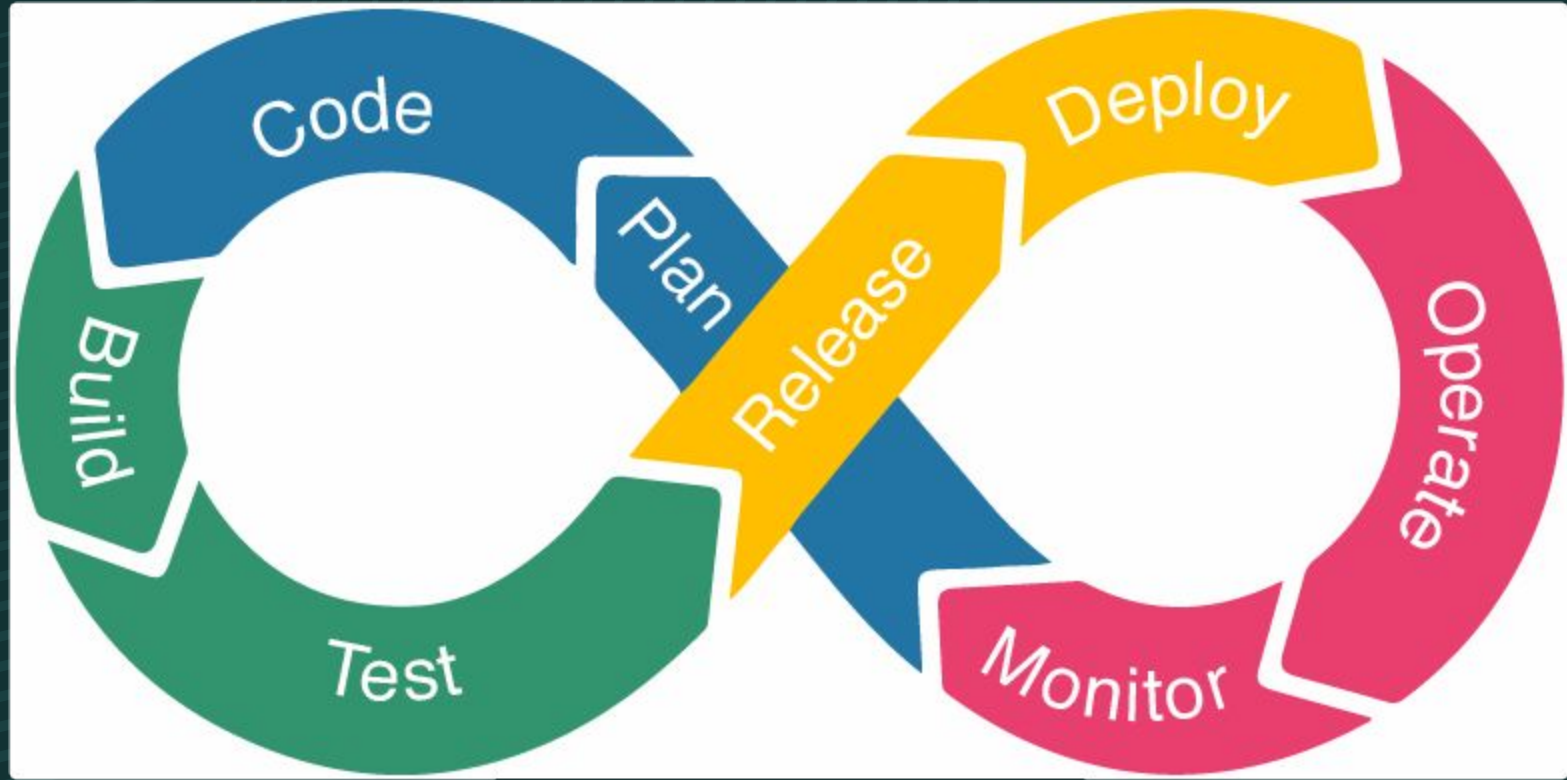


Image credit: [ROELBOB](#) at [DevOps.com](#)



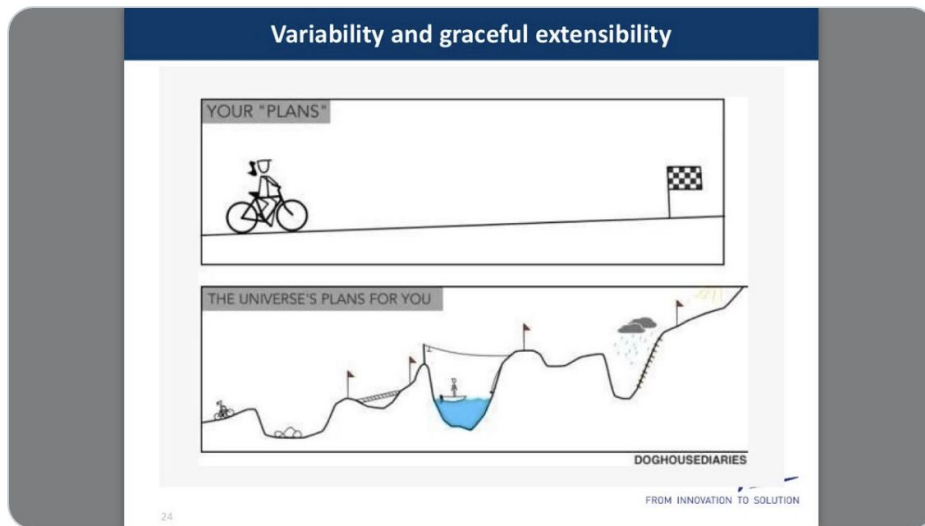
**John Allspaw**

@allspaw

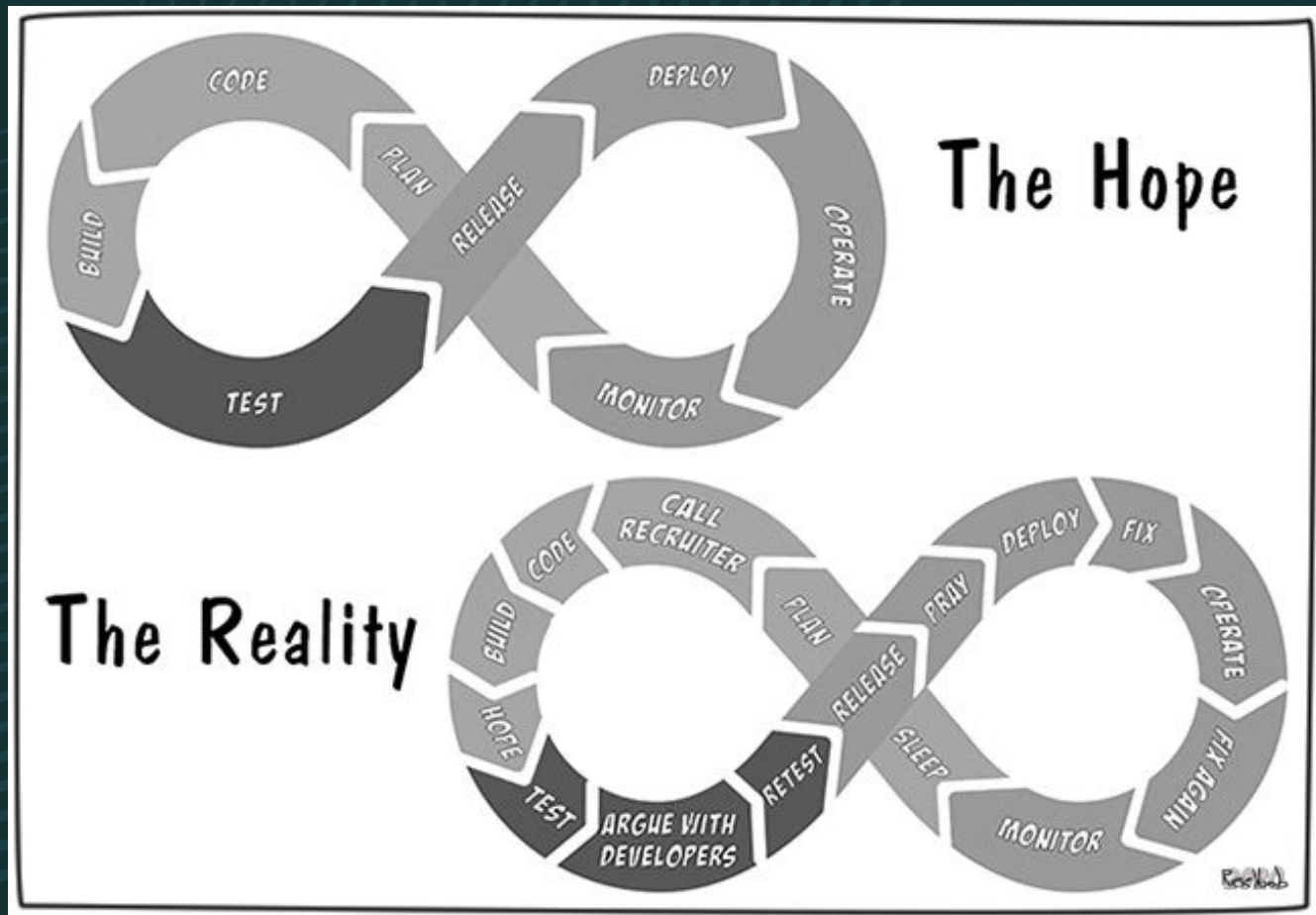
Normal 0%



## Work-as-imagined versus work-as-done



3:00 AM · Apr 28, 2016 · Twitter for iPhone



Imagine you're getting ready to ship  
something...





You press the big red  
button



\*click\*





\*silence\*

Do you...

a) Sweat and wait to see if it blows up

# Do you...

- a) Sweat and wait to see if it blows up
- b) Pack up, go home & enjoy your weekend



It all boils down to  
automation.



Image source: Forbes









Image source: Horizon Zero Dawn



PEBKAC

Tech support can't help you when you're the problem.

*Image source: Demotivational*

 @IAmJerdog

# Continuous {Integration|Deployment}



## Agile software development principles [\[ edit \]](#)

The *Manifesto for Agile Software Development* is based on twelve principles:<sup>[23]</sup>

1. Customer satisfaction by early and continuous delivery of valuable software.
2. Welcome changing requirements, even in late development.
3. Deliver working software frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the primary measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Best [architectures](#), requirements, and designs emerge from self-organizing teams
12. Regularly, the team reflects on how to become more effective, and adjusts accordingly

Framework	Main contributor(s)
Adaptive software development (ASD)	Jim Highsmith, Sam Bayer
Agile modeling	Scott Ambler, Robert Cecil Martin
Agile unified process (AUP)	Scott Ambler
Disciplined agile delivery	Scott Ambler
Dynamic systems development method (DSDM)	Jennifer Stapleton
Extreme programming (XP)	Kent Beck, Robert Cecil Martin
Feature-driven development (FDD)	Jeff De Luca
Lean software development	Mary Poppendieck, Tom Poppendieck
Lean startup	Eric Ries
Kanban	Taiichi Ohno
Rapid application development (RAD)	James Martin
Scrum	Ken Schwaber, Jeff Sutherland
Scrumban	
Scaled agile framework - SAFe	Scaled Agile, Inc.



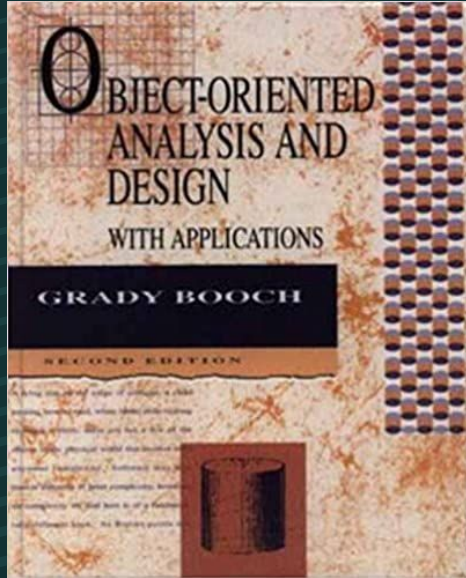
# Scrumban!

Practice	Main contributor(s)
Acceptance test-driven development (ATDD)	
Agile modeling	
Agile testing	
Backlogs (Product and Sprint)	Ken Schwaber
Behavior-driven development (BDD)	Dan North, Liz Keogh
Continuous integration (CI)	Grady Booch
Cross-functional team	
Daily stand-up / Daily Scrum	James O Coplien
Domain-driven design (DDD)	Eric Evans
Iterative and incremental development (IID)	
Pair programming	Kent Beck
Planning poker	James Grenning, Mike Cohn
Refactoring	Martin Fowler
Retrospective	
Scrum events (sprint planning, sprint review and retrospective)	
Specification by example	
Story-driven modeling	Albert Zündorf
Test-driven development (TDD)	Kent Beck
Timeboxing	
User story	Alistair Cockburn
Velocity tracking	

Continuous integration (CI)

Grady Booch

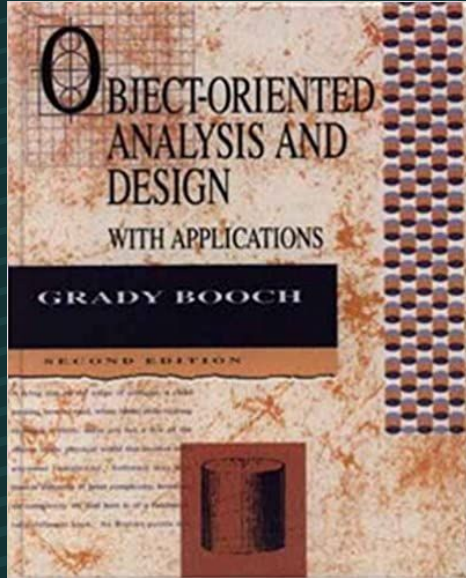
# Quick History of CI/CD



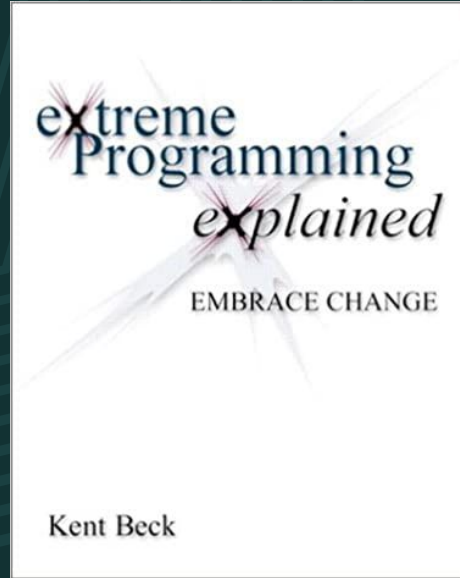
1991



# Quick History of CI/CD

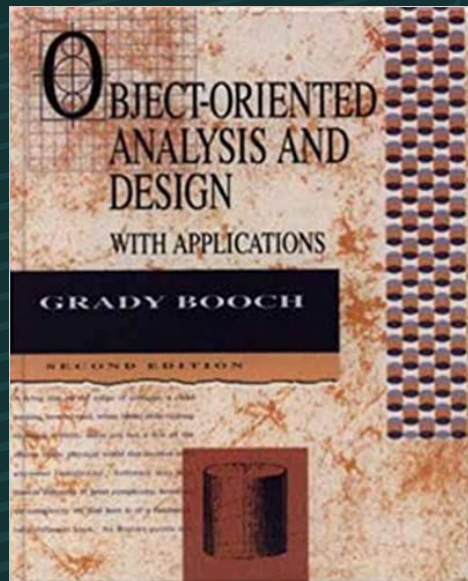


1991

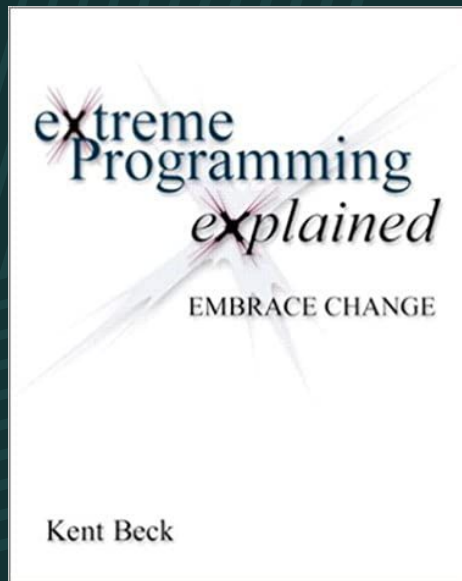


1997

# Quick History of CI/CD



1991



1997



2001

# Quick History of CI/CD



# Quick History of CI/CD



# Quick History of CI/CD



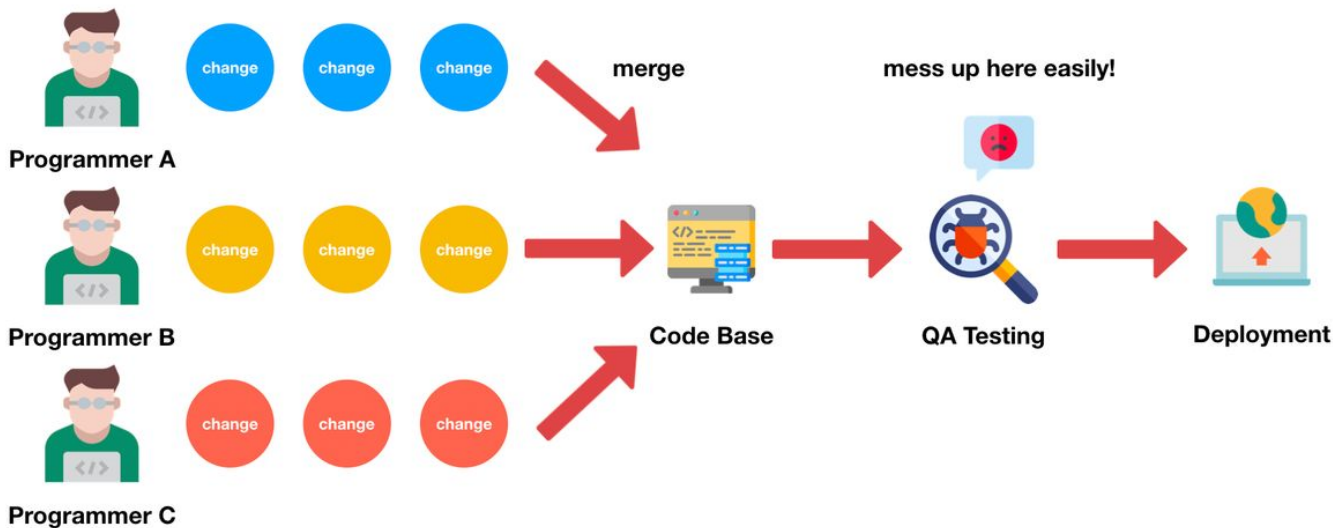
# Software Development without CI/CD





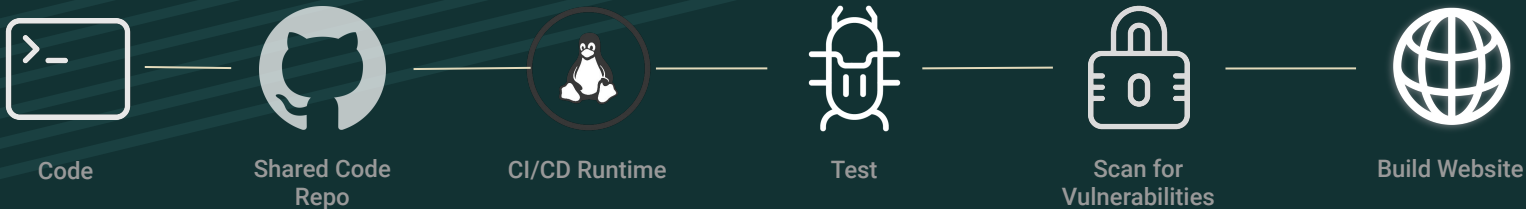
# Software Development without CI/CD

## Traditional Way



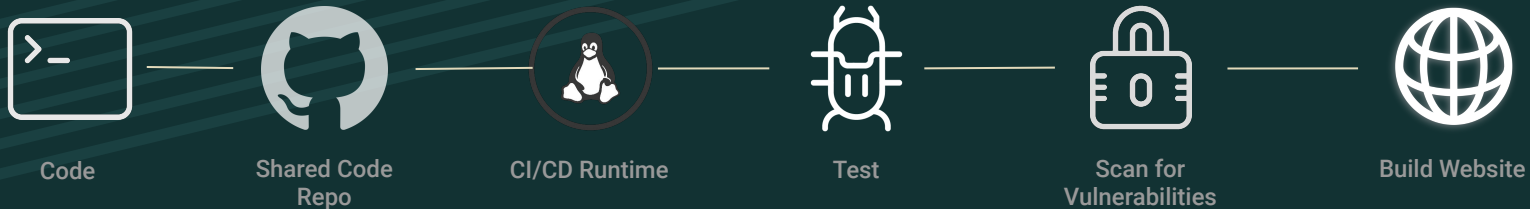
# CI == Continuous Integration

the **practice** of merging all developers' working copies to a **shared** code repository



# CI == Continuous Integration

- Merge code changes often



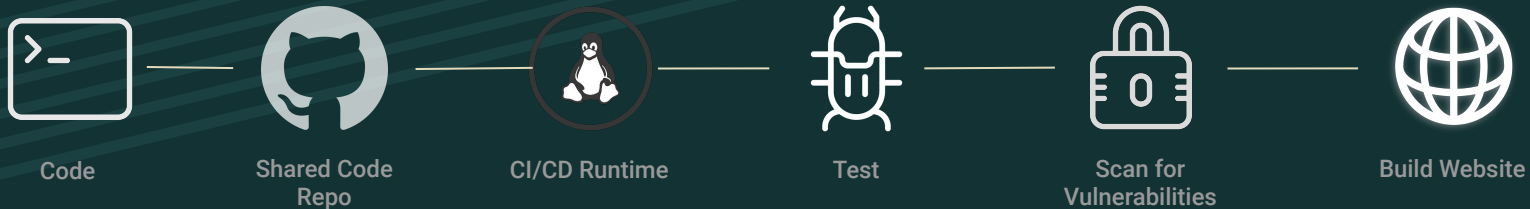
# CI == Continuous Integration

- Merge code changes often
- Run automated tests to validate builds



# CI == Continuous Integration

- Merge code changes often
- Run automated tests to validate builds
- Only integrate tested code into code base



# CI == Continuous Integration

- Merge code changes often
- Run automated tests to validate builds
- Only integrate tested code into code base
- Changes frequently merged into release branches



# CI == Continuous Integration

the **practice** of merging all developers' working copies to a **shared** code repository





# CI == Continuous Integration

- Merge code changes often



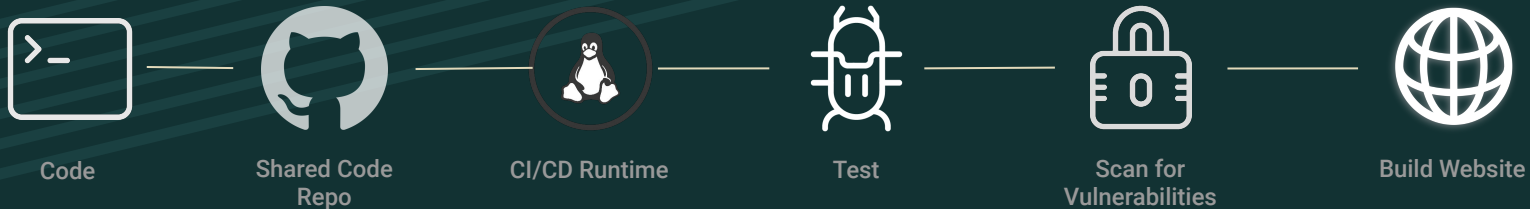
# CI == Continuous Integration

- Merge code changes often
- Run automated tests to validate builds



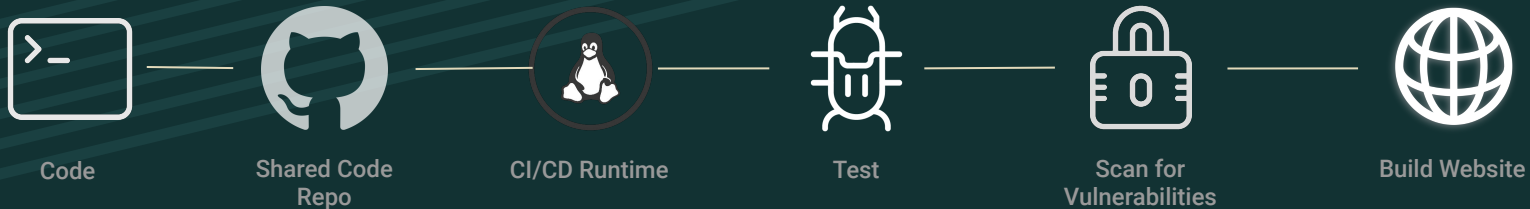
# CI == Continuous Integration

- Merge code changes often
- Run automated tests to validate builds
- Only integrate tested code into code base



# CI == Continuous Integration

- Merge code changes often
- Run automated tests to validate builds
- Only integrate tested code into code base
- Changes frequently merged into release branches



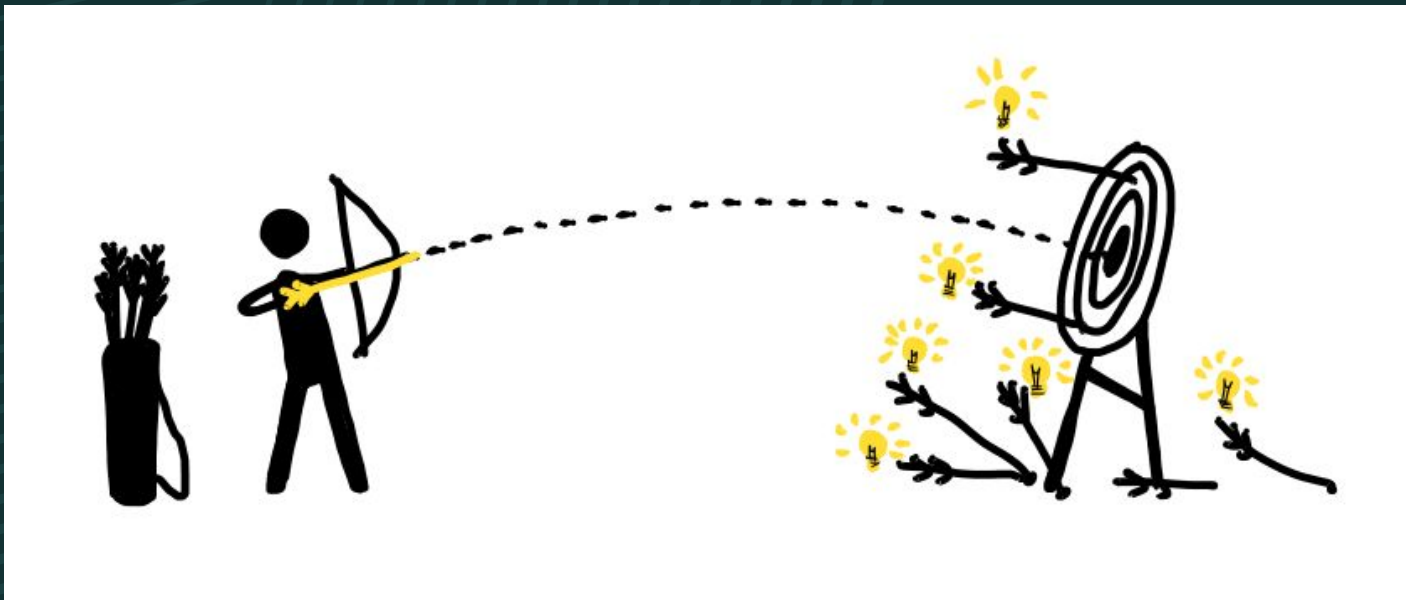


Image credit: [Product Coalition](#)

# CD == Continuous Deployment

the **practice** of automatically deploying new software releases to **target environments**



Build Website



Server  
deployment



Active Users

# CD == Continuous Deployment

- Faster release cycles



Build Website



Server  
Deployment



Active Users



# CD == Continuous Deployment

- Faster release cycles
- Low-risk releases



Build Website



Server  
Deployment



Active Users

# CD == Continuous Deployment

- Faster release cycles
- Low-risk releases
- Higher quality



Build Website



Server  
Deployment



Active Users

# CD == Continuous Deployment

- Faster release cycles
- Low-risk releases
- Higher quality
- Lower costs



Build Website



Server  
Deployment



Active Users

Is CI/CD a TOOL?

Are there benefits to CI/CD?

# CI/CD Benefits



- Improve team productivity & efficiency
- Accelerate speed to market
- Identify product/market fit
- Release higher quality, more stable products
- Increase customer satisfaction
- Keep devs happy and shipping code





# I feel the need... the need for SPEED!



- Customers get access to new features faster
- Company gets a faster ROI from new features

# Implementing CI/CD

Make sure everyone is on the same page

Always start small

Do what works for you

Always measure



# CI/CD Benchmarks



Throughput

At will



Duration

<10 minutes



Success Rate

> 90%



Mean Time  
to Recovery

<1 hour

# Better CI/CD Practices

- Make testing an integral part of the dev process

# Better CI/CD Practices

- Make testing an integral part of the dev process
- Ensure testing environment mirrors production

# Better CI/CD Practices

- Make testing an integral part of the dev process
- Ensure testing environment mirrors production
- Use coding better practices, i.e. pair programming

# Better CI/CD Practices

- Make testing an integral part of the dev process
- Ensure testing environment mirrors production
- Use coding better practices, i.e. pair programming
- Automate the deploy workflow

# Streamline deploy

elated-clarke-f67cc5

• Site deploy in progress

Deploys from [GitHub](#). Created at 5:48 PM.

⚙ Site settings ⚙ Domain settings

🔗 Production deploys

Production: master@HEAD **Starting Up**

5:48 PM: No deploy message

🔗 Deploy Previews

Deploy Preview #4: jerdog-patc... @a727cb0 **Building**


5:54 PM: Update config.toml



## Deploy

Deployment started 9s ago...

- ▶ Fetching Source Code 6s
- ▶ Skipping Build Cache 1s
- ▶ Building
- ▶ Uploading Build Output



## Preview Deployments

Deployments that are currently being worked on.

test-nextjs-vercel-qxdiffra-jerdog.ve... Preview	Queued	Update index.js jerdog-patch-1	Just now by jerdog
---	--------	-----------------------------------	--------------------





# Full Report



<https://circle.ci/ssd2022>

# Thank you.

For feedback and swag: [circle.ci/jeremy](https://circle.ci/jeremy)

BEERCITYCODE



Timeline.jerdog.me



IAmJerdog



jerdog



/in/jeremymeiss