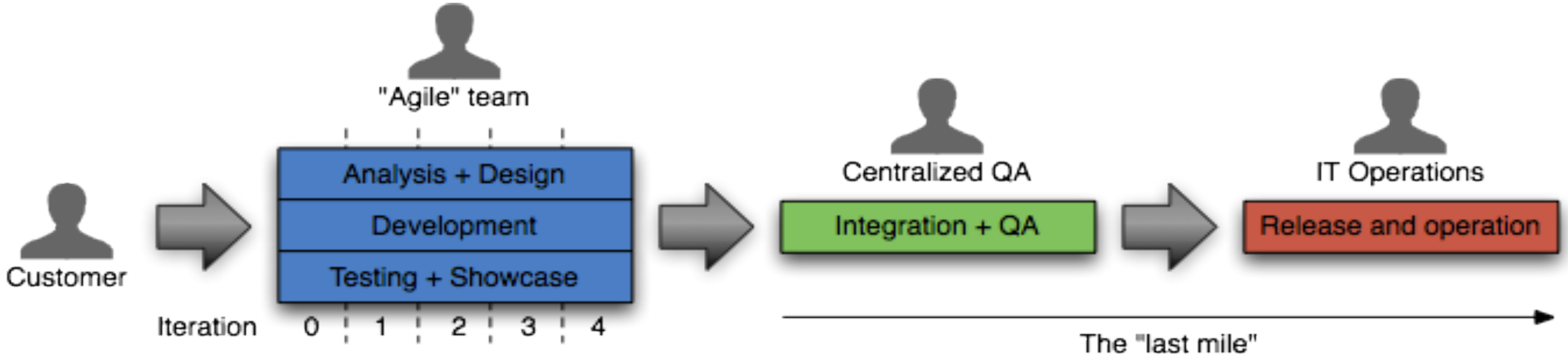
ThoughtWorks®

Continuous Delivery

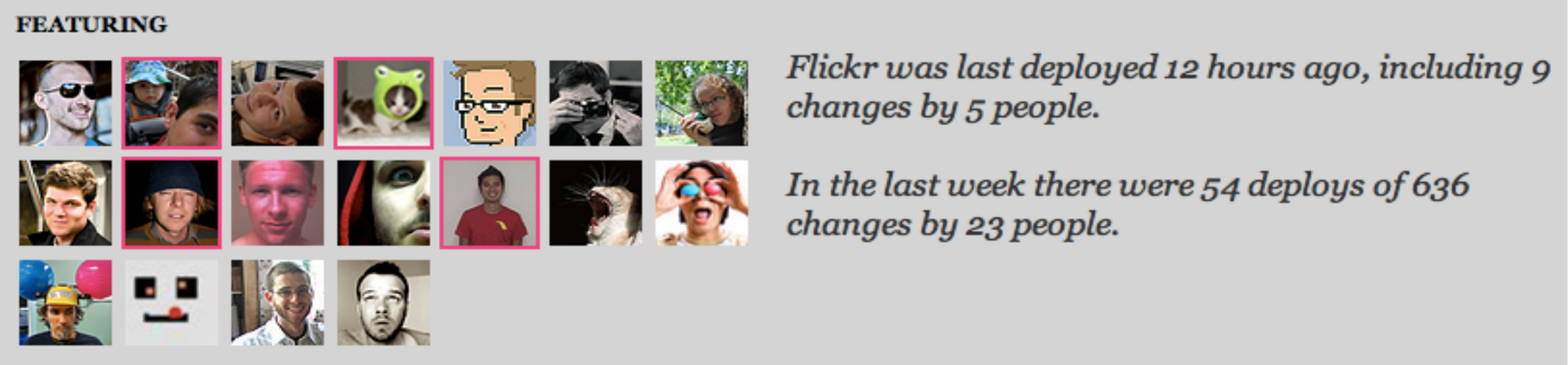
Ajey Gore
Head of Technology
ThoughtWorks India

Agile Delivery



web 2.0

- disrupting traditional businesses



FEATURING

Flickr was last deployed 12 hours ago, including 9 changes by 5 people.

In the last week there were 54 deploys of 636 changes by 23 people.

http://code.flickr.com/

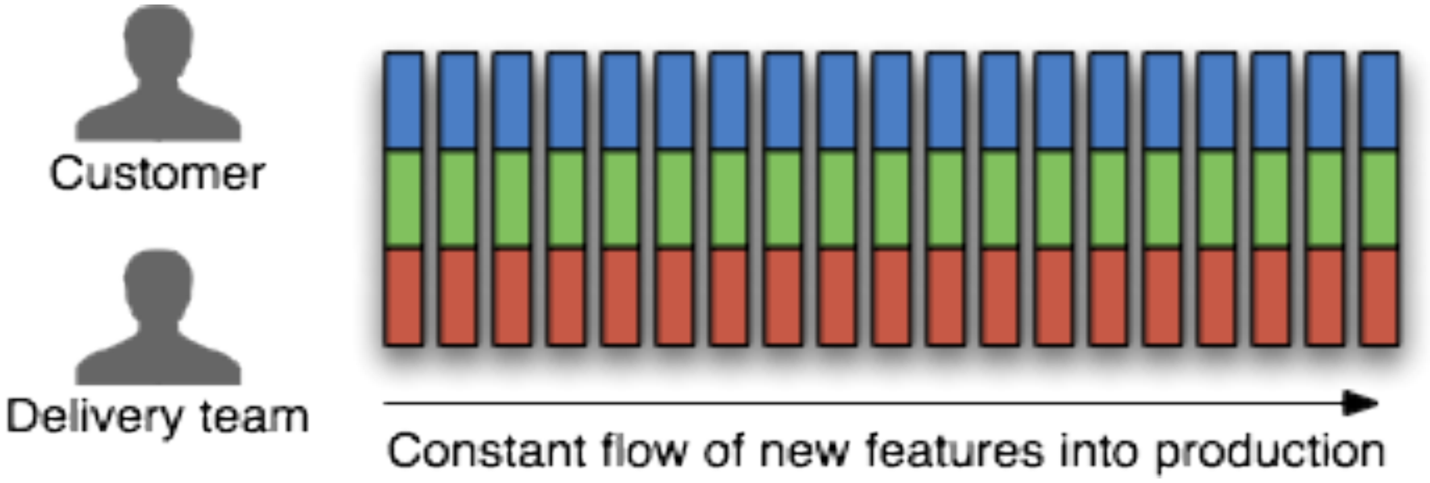
releasing frequently

- feedback from users
- reduce risk of release
- real project progress

production-ready software

- **Fast, automated feedback** on the **production readiness** of your applications **every** time there is a **change** – to code, infrastructure, or configuration

continuous delivery



- software always production ready
- releases tied to business needs, not operational constraints

Puzzle

- Dev team's job is to add features, fix bugs and make software better
  - Owns software features and evolution
- Ops job is to keep software stable and fast
  - Owns availability and performance

Reality

- Everyone’s job is to **enable** business
- Business **requires** change

culture

Ops have requirements too!

Ops at inceptions, showcases, retrospectives

Devs work in ops and get notifications

Trust / access

Cross-functional delivery teams

automation

Provisioning and managing environments

Push-button deployments

Database migrations

managing environments

Cloud computing / virtualization

Puppet / Chef

Continuous Integration for environment changes

managing environments

- If someone threw a server out of the window, how long would it take to recreate it?

ask this question

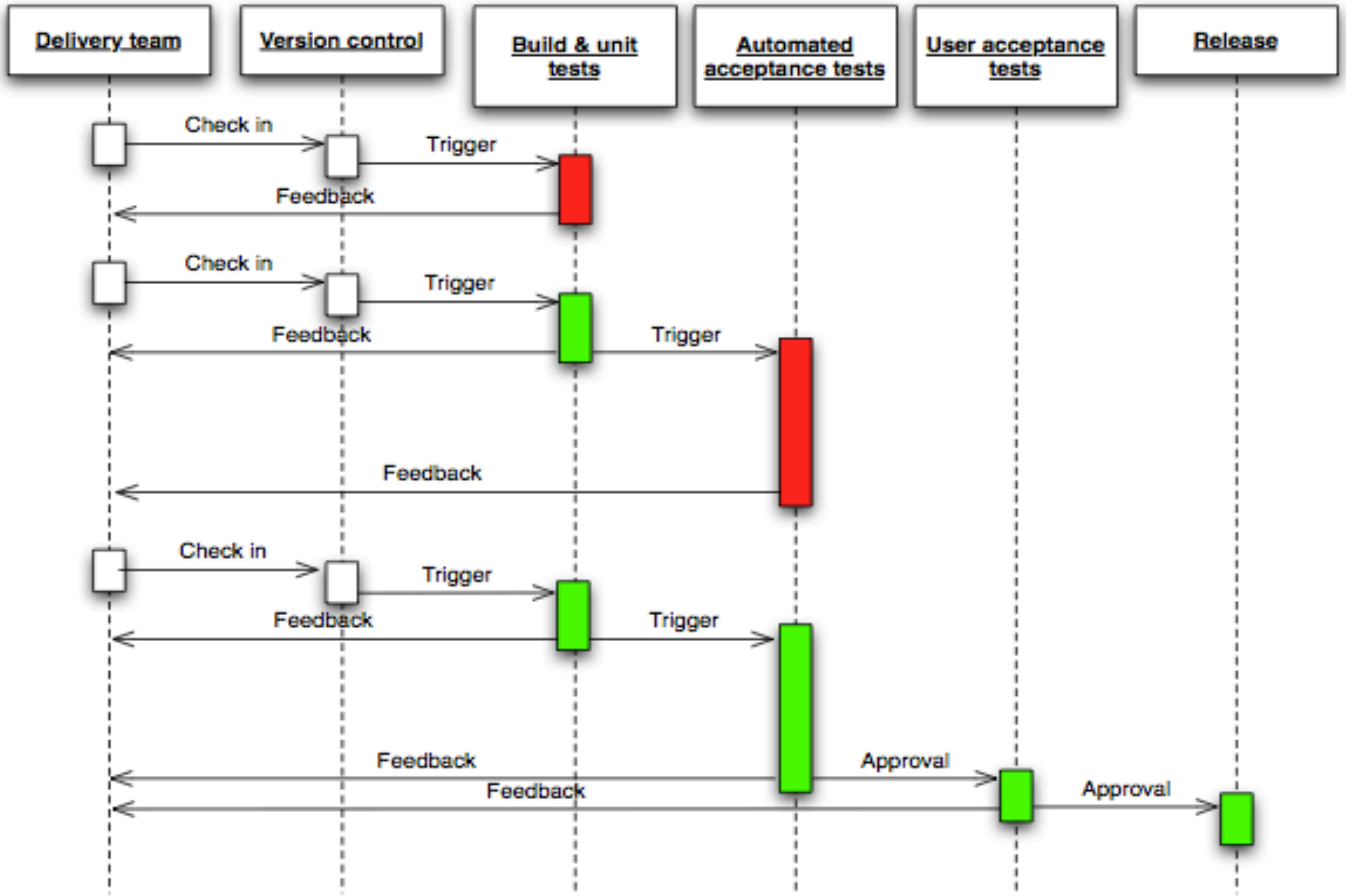
- “How long would it take your organization to deploy a change that involved just one single line of code? Do you do this on a repeatable, reliable basis?”
- What gets in the way of getting software out of the door?

Mary and Tom Poppendieck, Implementing Lean Software Development, p59.

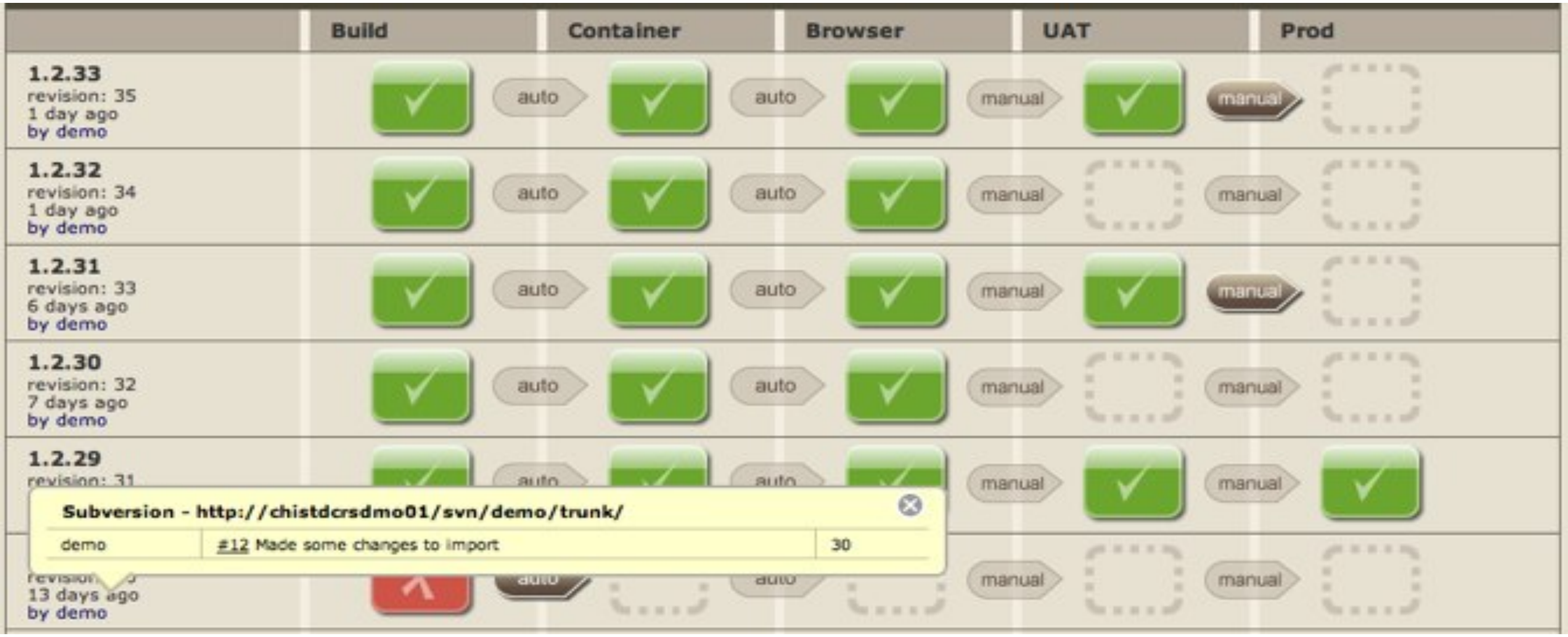
practices

- only build your binaries once
- deploy the same way to every environment
- smoke test your deployments
- keep your environments similar
- if anything fails, stop the line

deployment pipeline



deployment pipeline



	Build		Container		Browser		UAT		Prod
1.2.33 revision: 35 1 day ago by demo	✓	auto	✓	auto	✓	manual	✓	manual	
1.2.32 revision: 34 1 day ago by demo	✓	auto	✓	auto	✓	manual		manual	
1.2.31 revision: 33 6 days ago by demo	✓	auto	✓	auto	✓	manual	✓	manual	
1.2.30 revision: 32 7 days ago by demo	✓	auto	✓	auto	✓	manual		manual	
1.2.29 revision: 31	✓	auto	✓	auto	✓	manual	✓	manual	✓
revisio 13 days ago by demo	✗	auto		auto		manual		manual	

Subversion - http://chistdcrsdmo01/svn/demo/trunk/

demo	#12 Made some changes to import	30

principles

- create a repeatable, reliable process for releasing software
- automate almost everything
- keep everything in version control
- if it hurts, do it more often, and bring the pain forward
- build quality in
- done means released
- everybody is responsible for delivery
- continuous improvement

people are the key

Get everyone together at the beginning

Keep meeting

Make it easy for everyone to see what’ s happening

Continuous improvement (kaizen)

thank you!

agore@thoughtworks.com

Twitter: netbuzzme

http://continuousdelivery.com/

http://studios.thoughtworks.com/go

http://thoughtworks.com/