Continuous Delivery
@jezhumble
#Agile2012, 15 August 2012

http://thoughtworks-studios.com/
agile 101

The "last mile"
web 2.0

disrupting traditional businesses

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

build the right thing

Customer development

Agile product development

You can't just ask customers what they want and then try to give that to them. By the time you get it built, they'll want something new.

Steve Jobs
scientific method

create hypothesis

deliver minimum viable product

get feedback

(repeat)

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?”

releasing frequently

build the right thing
reduce risk of release

John Allspaw: “Ops Metametrics” http://slideshare.net/dsSZIr
optimize for mtrs

Time to restore service

Time to detect  Find causes  RemEDIATE
optimize for mtrs

MTBF

MTRS
releasing frequently

build the right thing
reduce risk of release
real project progress
Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
production-ready software

Fast, automated feedback on the production readiness of your applications every time there is a change – to code, infrastructure, or configuration
Software always production ready
Releases tied to business needs, not operational constraints
continuous delivery

automation

patterns and practices

collaboration
ingredients

configuration management

continuous integration

automated testing
Mainline Server

Local Workstation

Build

Develop

pull

push

✔

Done!

Thursday, August 16, 12
Everyone Commits To the Mainline Every Day
build quality in

“Cease dependence on mass inspection to achieve quality. Improve the process and build quality into the product in the first place”

W. Edwards Deming
different kinds of testing

- Functional acceptance tests
- Integration tests
- System tests
- Unit tests
- Showcases
- Usability testing
- Exploratory testing
- Non-functional acceptance tests (performance, scaling, ...)

Diagram invented by Brian Marick
deployment pipeline

an automated implementation of your system’s build, deploy, test, release process

visibility

feedback

control
deployment pipeline
deployment pipeline
reducing release risk

automate provisioning and deployment

ensure devs, testers and ops collaborate throughout
reducing release risk

devops

incrementalism

decoupling deployment and release
devops

culture
automation
measurement
sharing
low risk releases are incremental

blue-green deployments
canary releases
dark launching

feature toggles
production immune system
blue–green deployments
Canary Releasing
canary releasing

Reduce risk of release

Multi-variant testing

Performance testing
immune system

what if someone replaced your “buy” button with spacer.gif?

T cells http://www.flickr.com/photos/gehealthcare/3326186490/
release != deployment
feature toggles

Config File

[featureToggles]
wobblyFoobars: true
flightyForkHandles: false

some.php

<?if ($wobblyFoobars) {?>
  ... various UI elements
<?}?>

other.php

$fork_handle = ($featureConfig->isOn('flightlyForkHandles')) ?
  new flightyForkHandler(aCandle) :
  new forkHandler(aCandle);

Stolen from Martin Fowler http://martinfowler.com/bliki/FeatureToggle.html
Dark Launching
enterprise governance

risk management

SOX, ITIL, COBIT

segregation of duties

change management

auditing and compliance
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)
questions

jez@thoughtworks.com
http://continuousdelivery.com/

@jezhumble #continuousdelivery