Continuous Delivery

continuous integration

integrate early & often

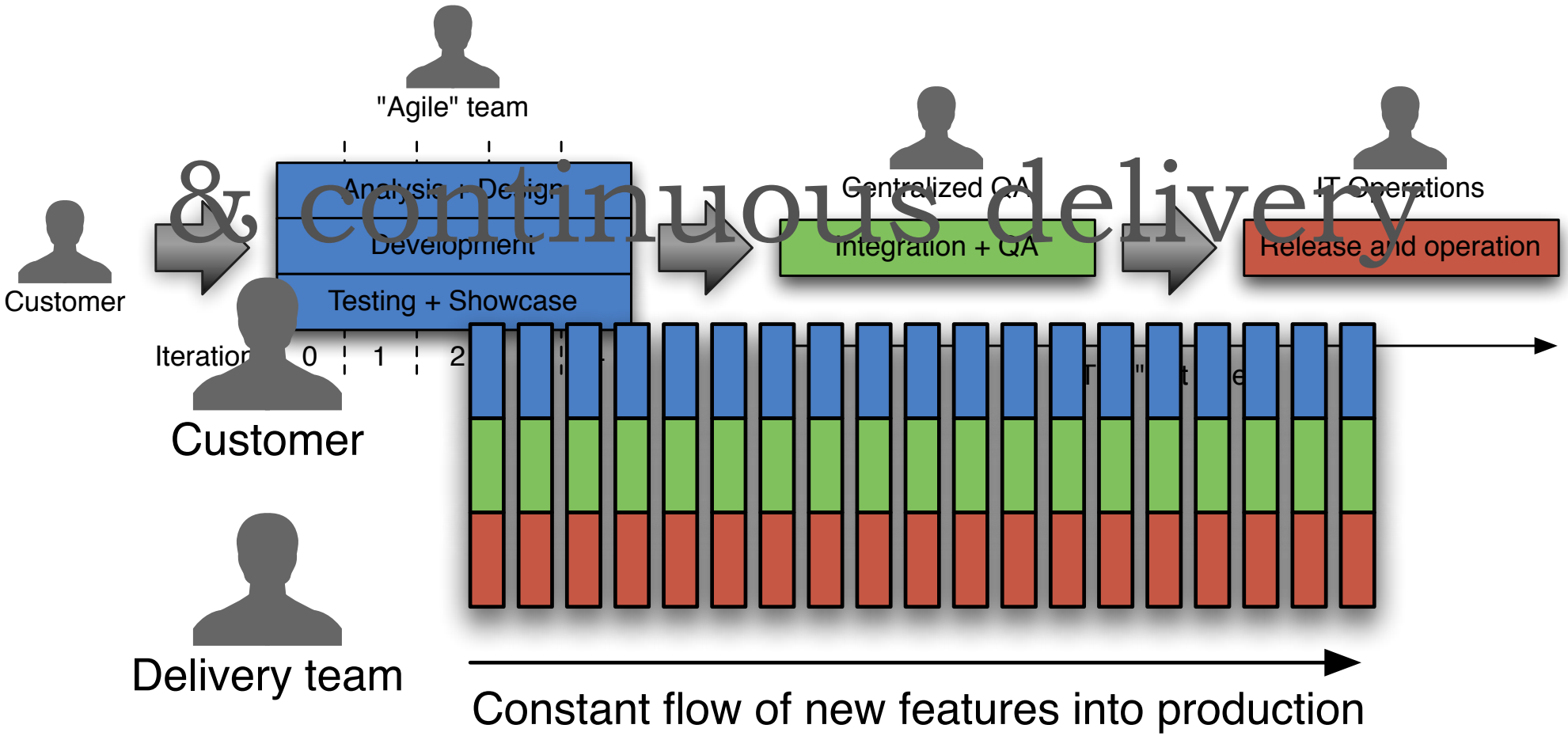
deployment

deploy as the final stage of CI

delivery

software is always deployable

agile 101



& continuous delivery

continuous integration

Fast, automated feedback on the correctness of your application every time there is a change to code

continuous delivery

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

continuous delivery ideal

software is always production ready

deployments are reliable and commonplace

everyone can self-service deployments

releases occur according to business needs, not operational constraints

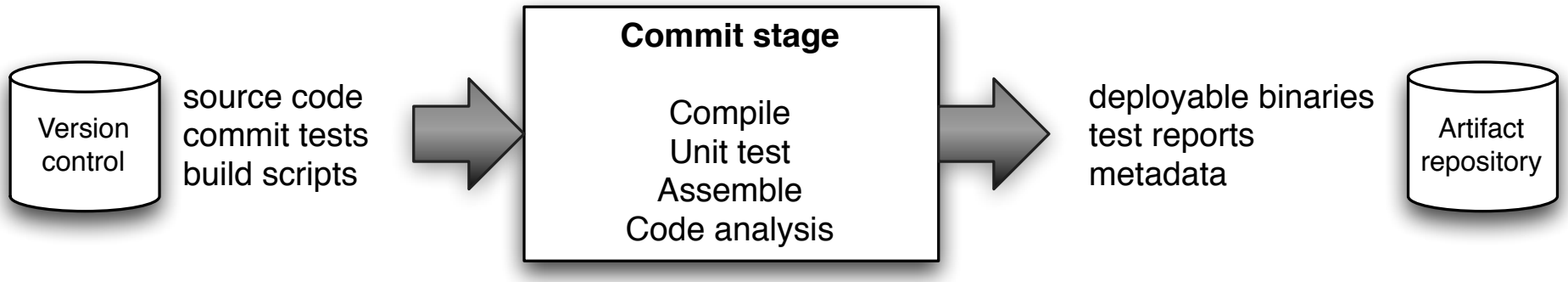
prerequisites

comprehensive configuration management

excellent automated testing at multiple levels

continuous integration

`commit` Stage

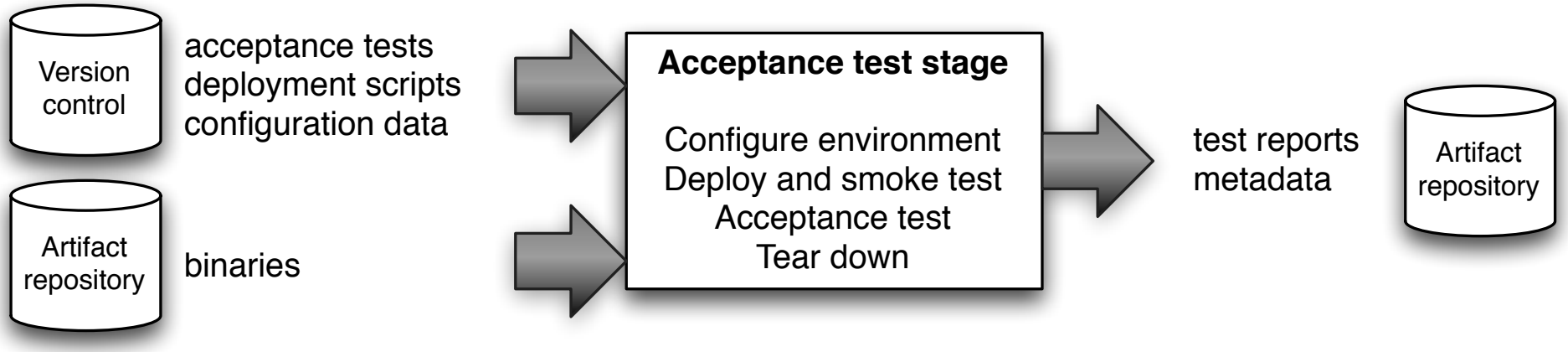


run against each check-in (continuous integration)

builds downstream artifacts

fix immediately upon failure

`acceptance` Stage

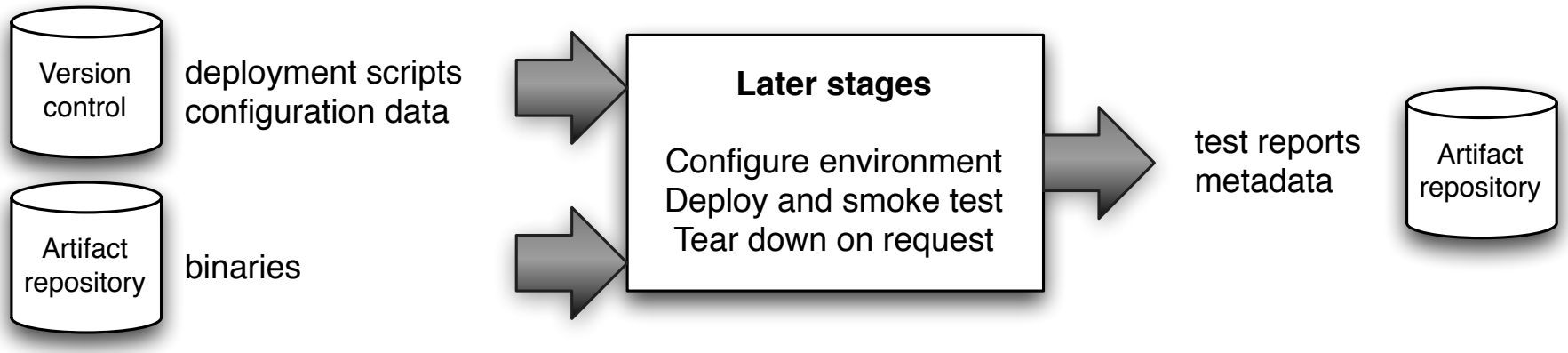


end-to-end tests in a production-like environment

triggered by upstream success

fix immediately upon failure

`manual` stage(s)

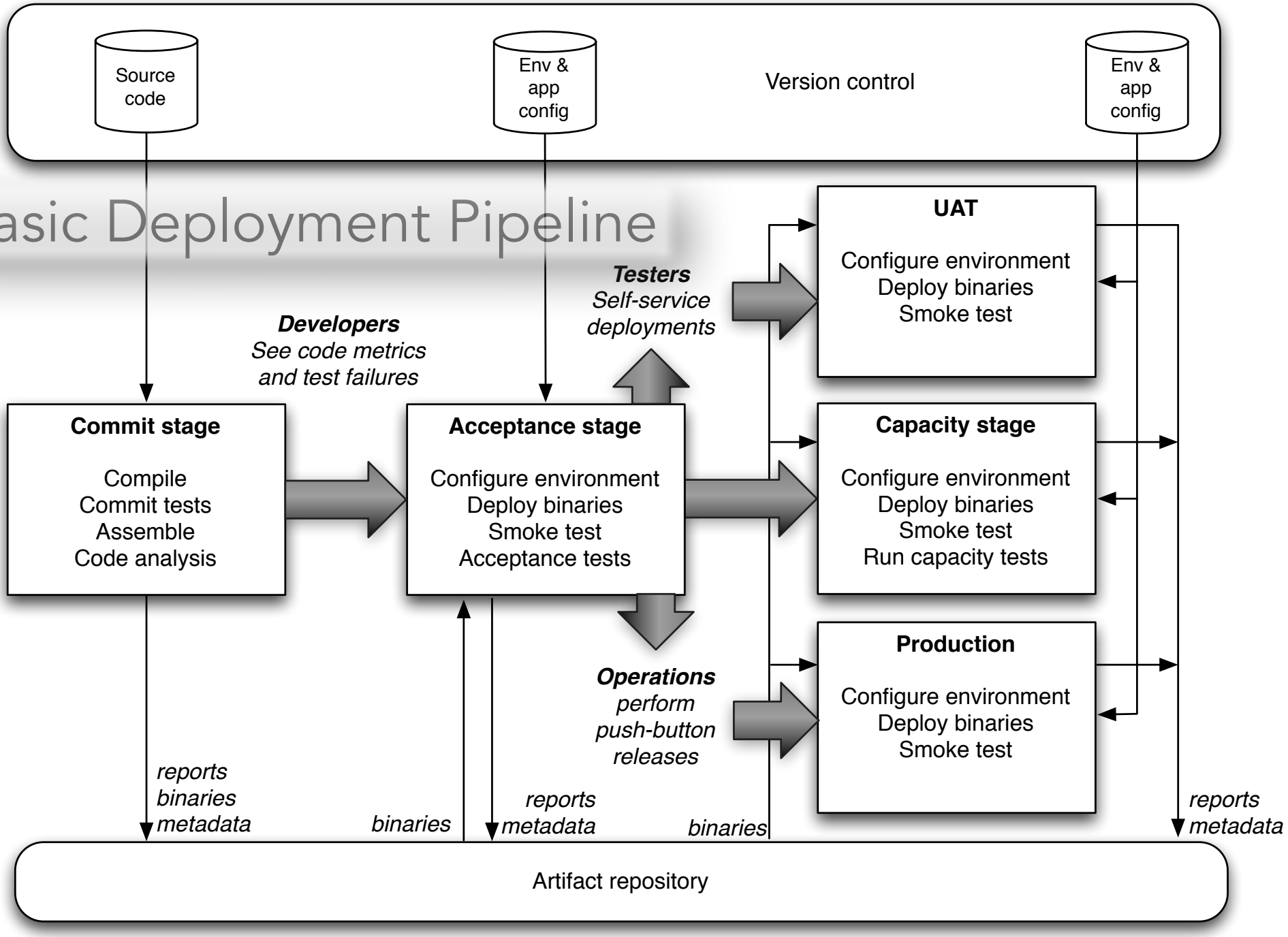


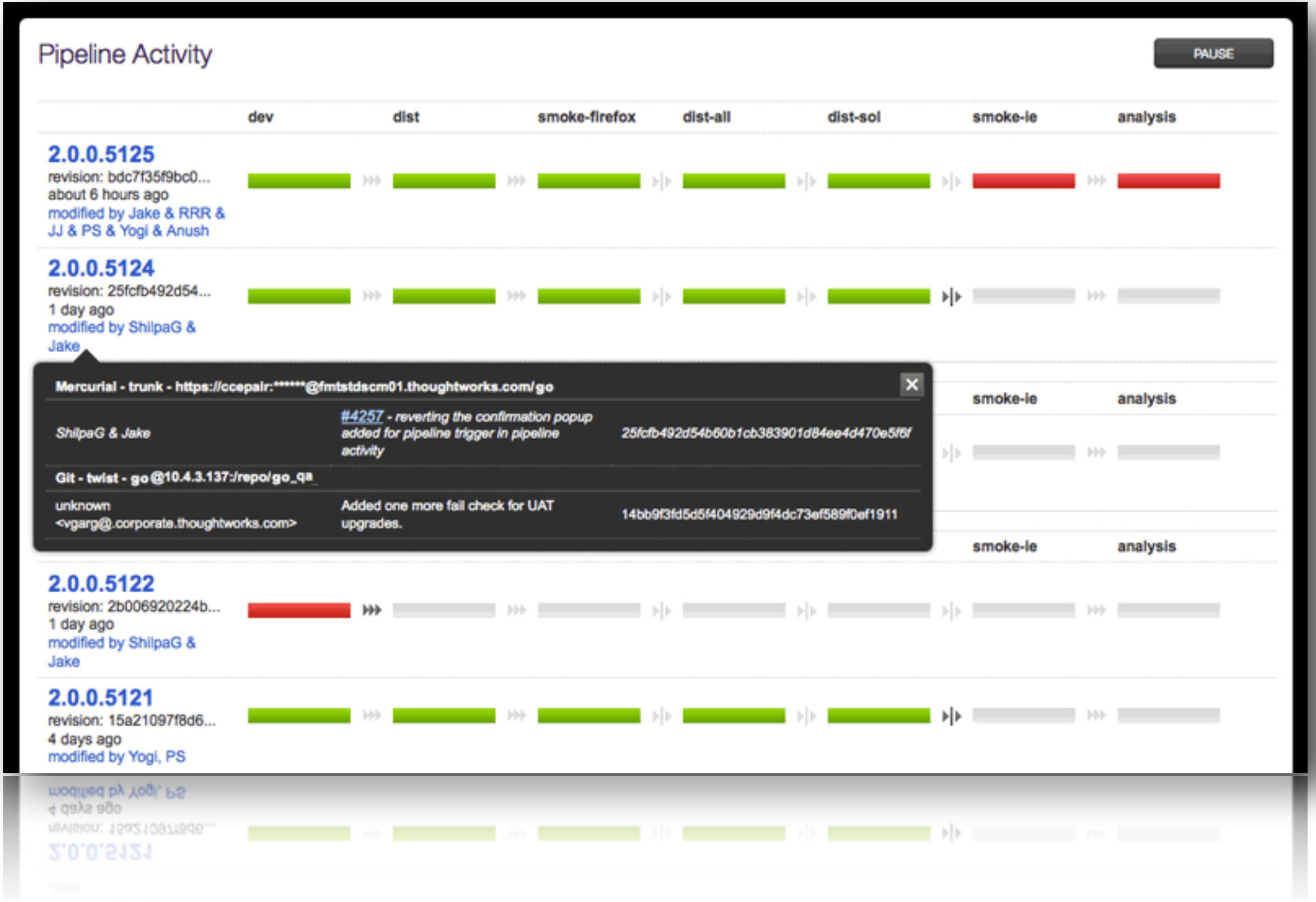
UAT, staging, integration, production

deployments self-serviced (push button)

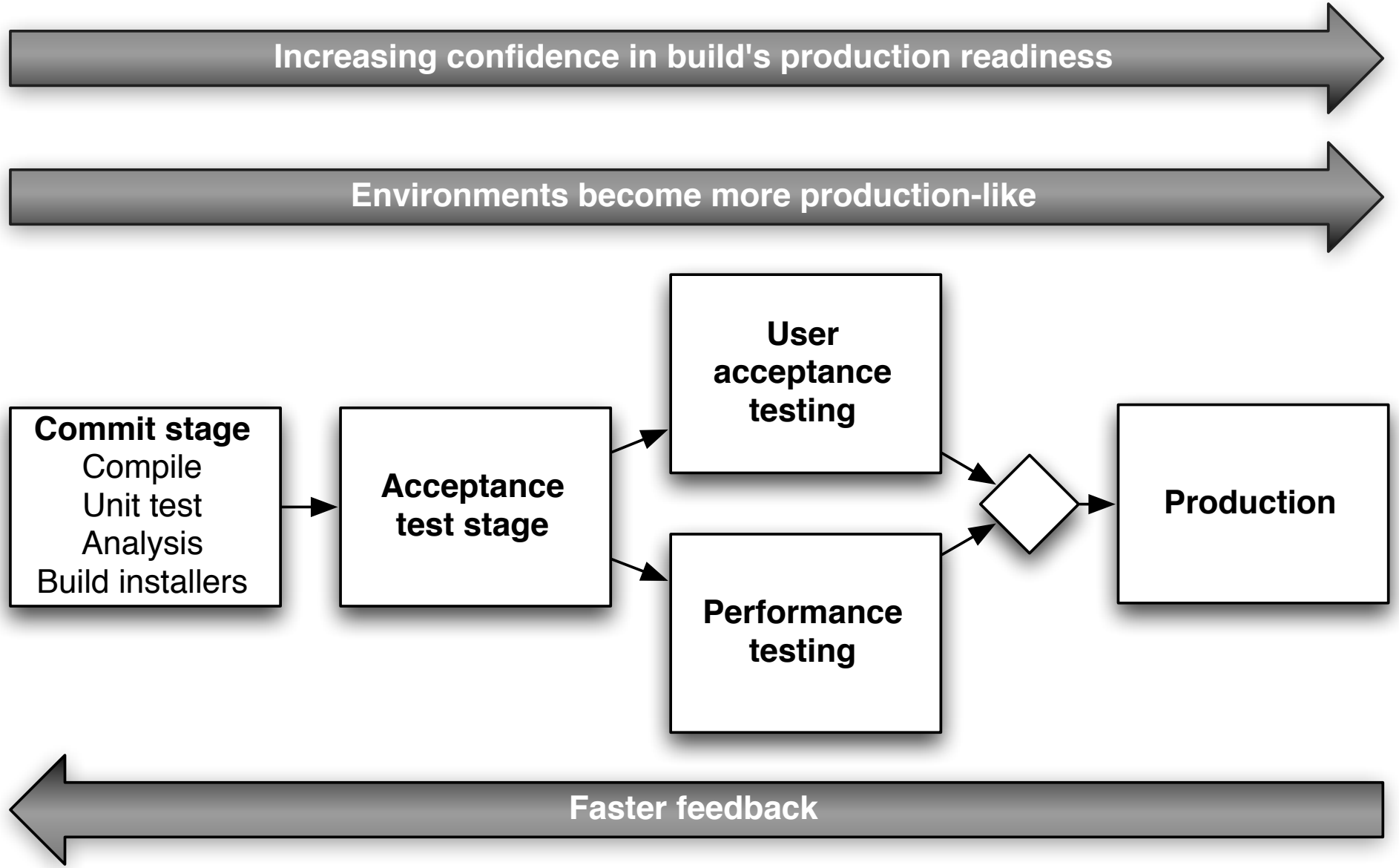
`push` **vs.** `pull` **systems**

Basic Deployment Pipeline




Pipeline Activity
PAUSE
dev
dist
smoke-firefox
dist-all
dist-sol
smoke-ie
analysis
2.0.0.5125
revision: bdc7f35f9bc0...
about 6 hours ago
modified by Jake & RRR & JJ & PS & Yogi & Anush
2.0.0.5124
revision: 25fcfb492d54...
1 day ago
modified by ShilpaG & Jake
Mercurial - trunk - https://ccepair:******@fmtstdscm01.thoughtworks.com/go
ShilpaG & Jake
#4257 - reverting the confirmation popup added for pipeline trigger in pipeline activity
25fcfb492d54b60b1cb383901d84ee4d470e5f6f
Git - twist - go @10.4.3.137:/repo/go_qa
unknown <vgarg@.corporate.thoughtworks.com>
Added one more fail check for UAT upgrades.
14bb9f3fd5d5f404929d9f4dc73ef589f0ef1911
2.0.0.5122
revision: 2b006920224b...
1 day ago
modified by ShilpaG & Jake
2.0.0.5121
revision: 15a21097f8d6...
4 days ago
modified by Yogi, PS

“production-like”



principles of software delivery

create a repeatable, reliable process for releasing software

Provisioning boxes

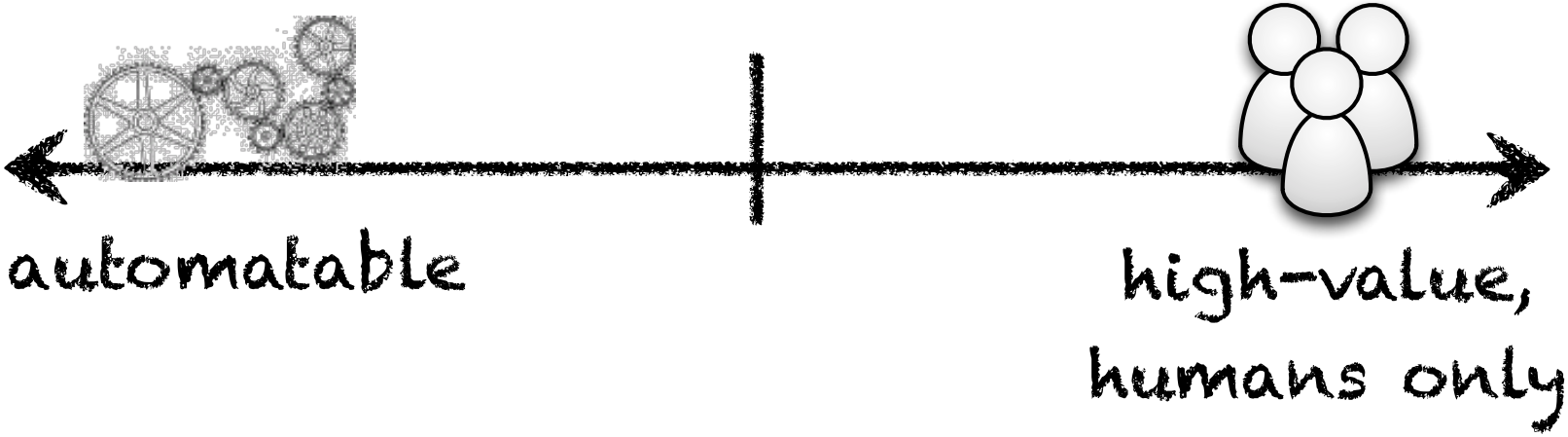
deployments

the more you do it, the better tested it is

automate almost everything

build, deploy, test, release

manual testing, approvals



keep everything you need to build, deploy, test, & release in version control

requirements documents

test scripts

automated test cases

network config scripts

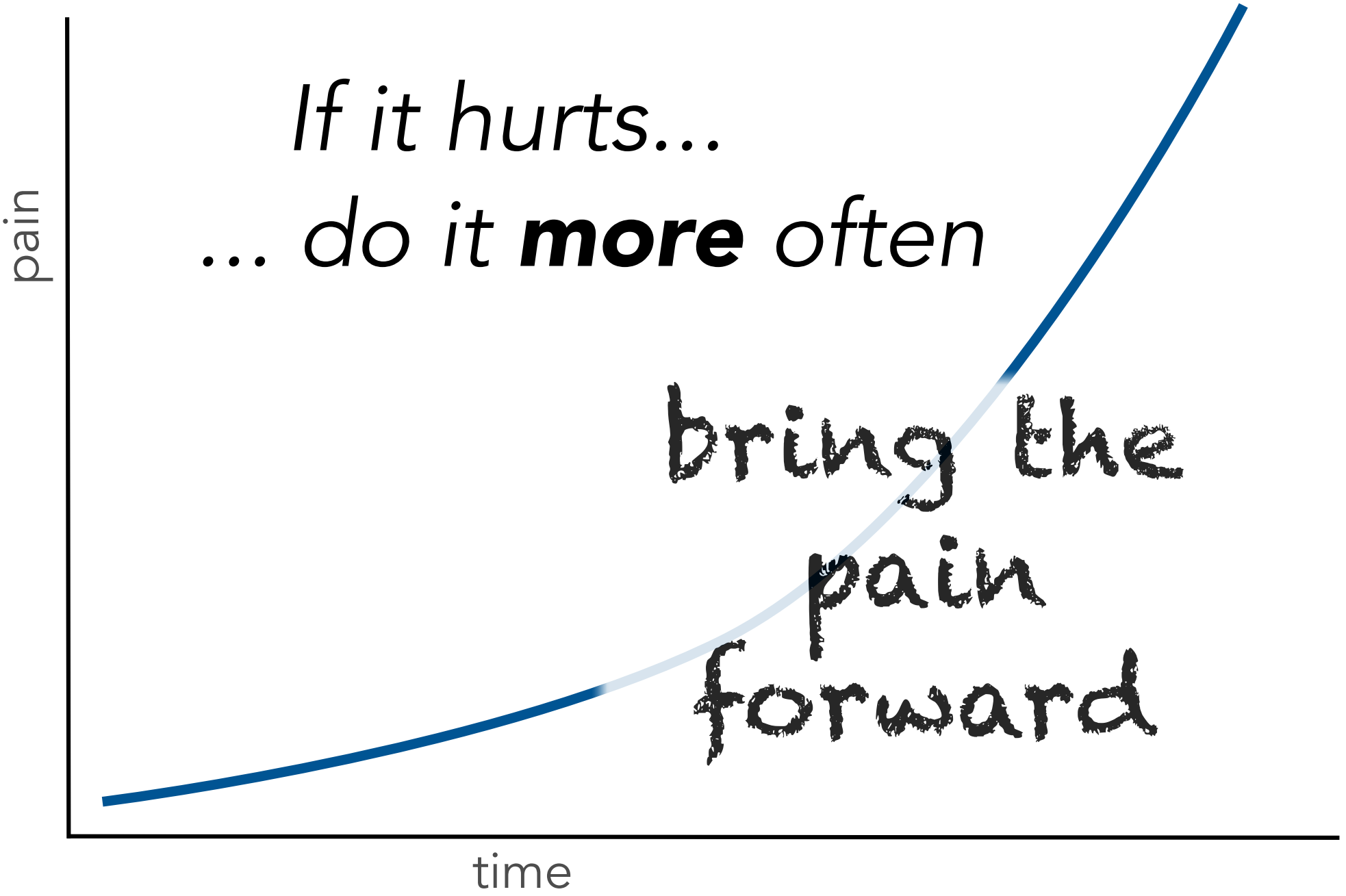
technical documentation

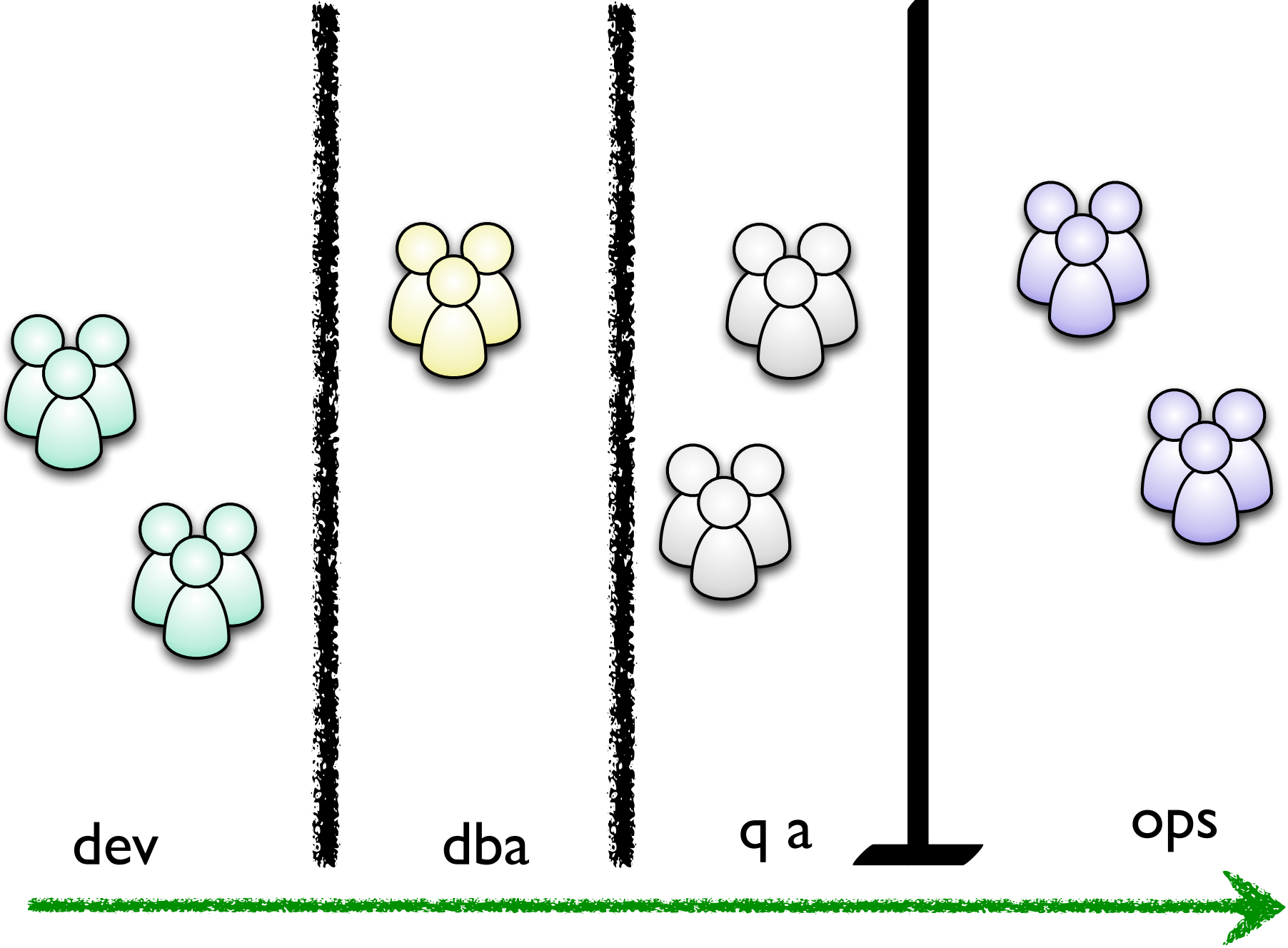
database creation, manipulation, & initialization scripts

application scripts

libraries

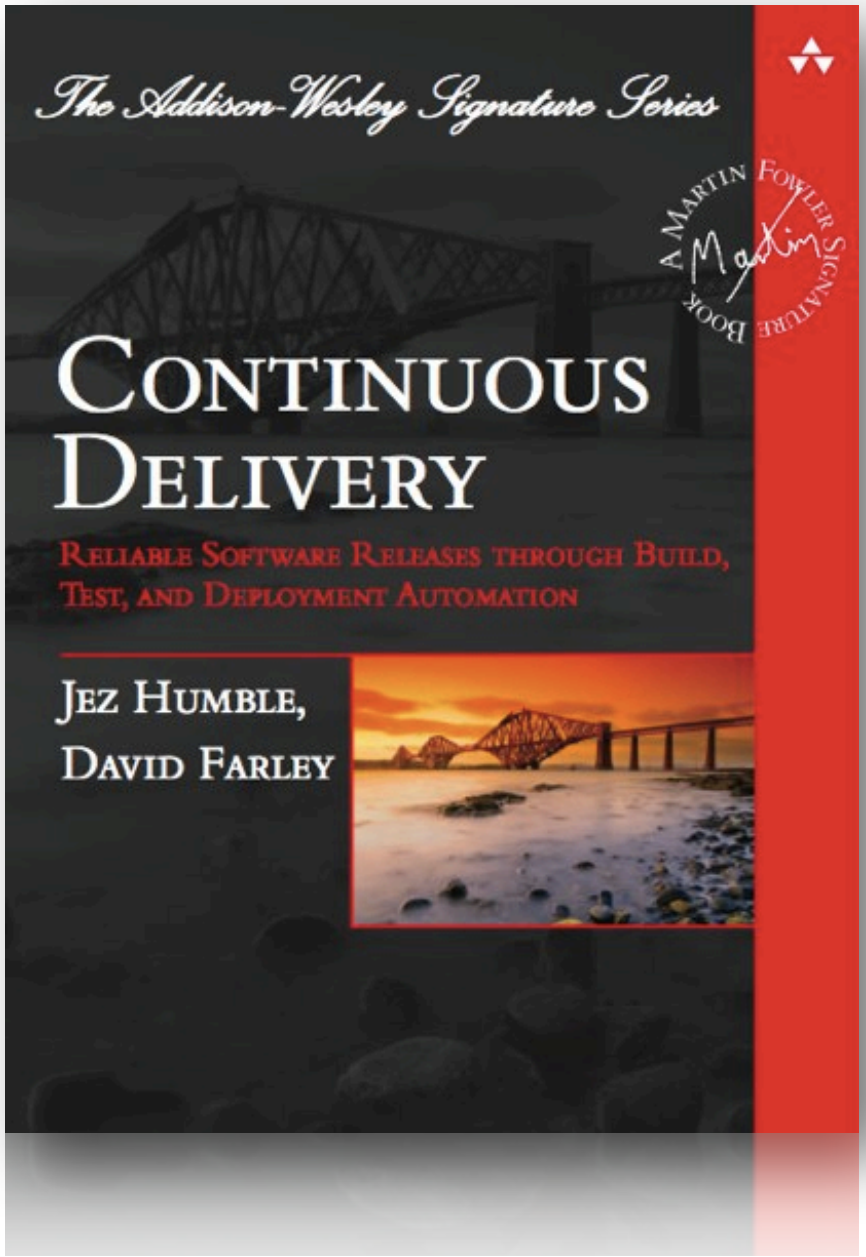
tool chains


If it hurts...
... do it **more** often
bring the pain forward
pain
time



everyone is responsible for delivery

Continuous Delivery



Jez Humble
David Farley

more to come
about architectural
impacts of continuous
delivery…

?'S

Mark Richards
Independent Consultant
Hands-on Enterprise / Integration Architect
Published Author / Conference Speaker

http://www.wmrichards.com
http://www.linkedin.com/pub/mark-richards/0/121/5b9

Published Books:
Java Message Service, 2nd Edition
97 Things Every Software Architect Should Know
Java Transaction Design Strategies

Neal Ford
Director / Software Architect /
Meme Wrangler

ThoughtWorks®

2002 Summit Blvd, Level 3, Atlanta, GA 30319, USA
T: +1 40 4242 9929 Twitter: @neal4d
E: nford@thoughtworks.com W: thoughtworks.com