

DevOps: What, Why & How?

Vidya Vrat Agarwal

www.MyPassionFor.Net | [@dotNetAuthor](#)
<https://www.linkedin.com/in/vidyavrat/>

Agenda

- DevOps & It's Key Values
- DevOps Workflow
- Delivery Cycles
- Agile & DevOps
- DevOps Practices
- With / Without DevOps
- DevOps Metrics
- DevOps Automation
- DevOps Tool Chain

Industry Trends for IT



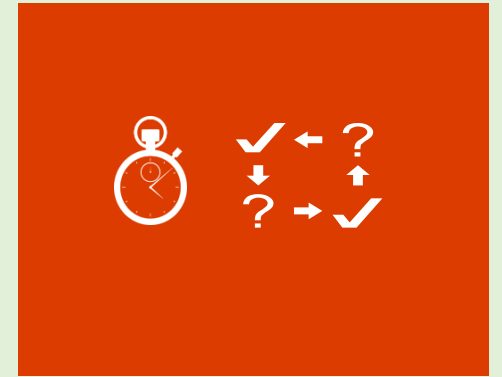
Heterogeneous
devices and
platforms



Hybrid IT

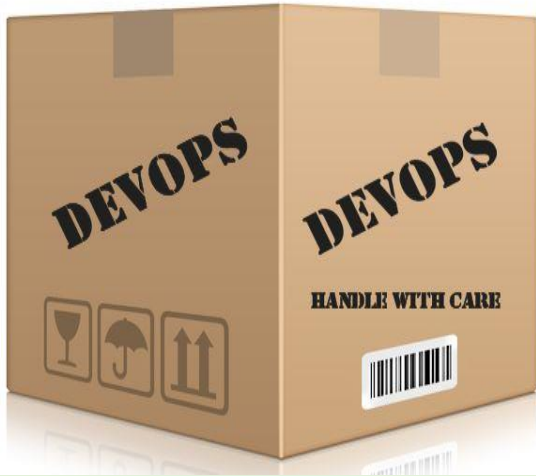


Distributed teams

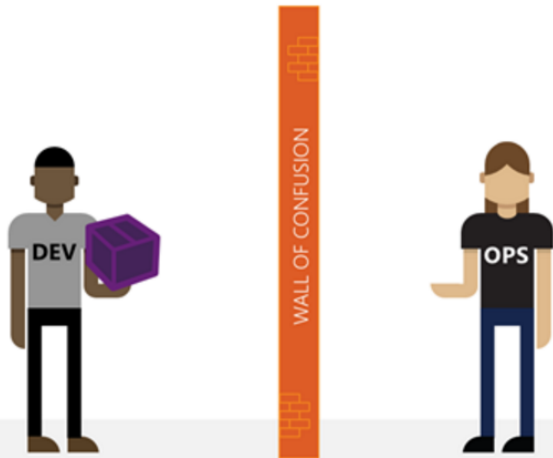


Delivery agility

DevOps & It's key values



- You cannot buy DevOps and install it.
- "DevOps is the union of people, process, and products to enable "continuous delivery" of value to our end users." – *Donovan Brown, MSFT*



- Breaks walls of confusion between teams & fosters better communication and collaboration throughout the application development lifecycle.
- Delivers software more frequently & produces higher quality software.
- Shortens lead time and software delivery cycles.

*“Continuous Delivery is a software development discipline where you **build software** in such a way that the software can be **released** to production at **any time**”*



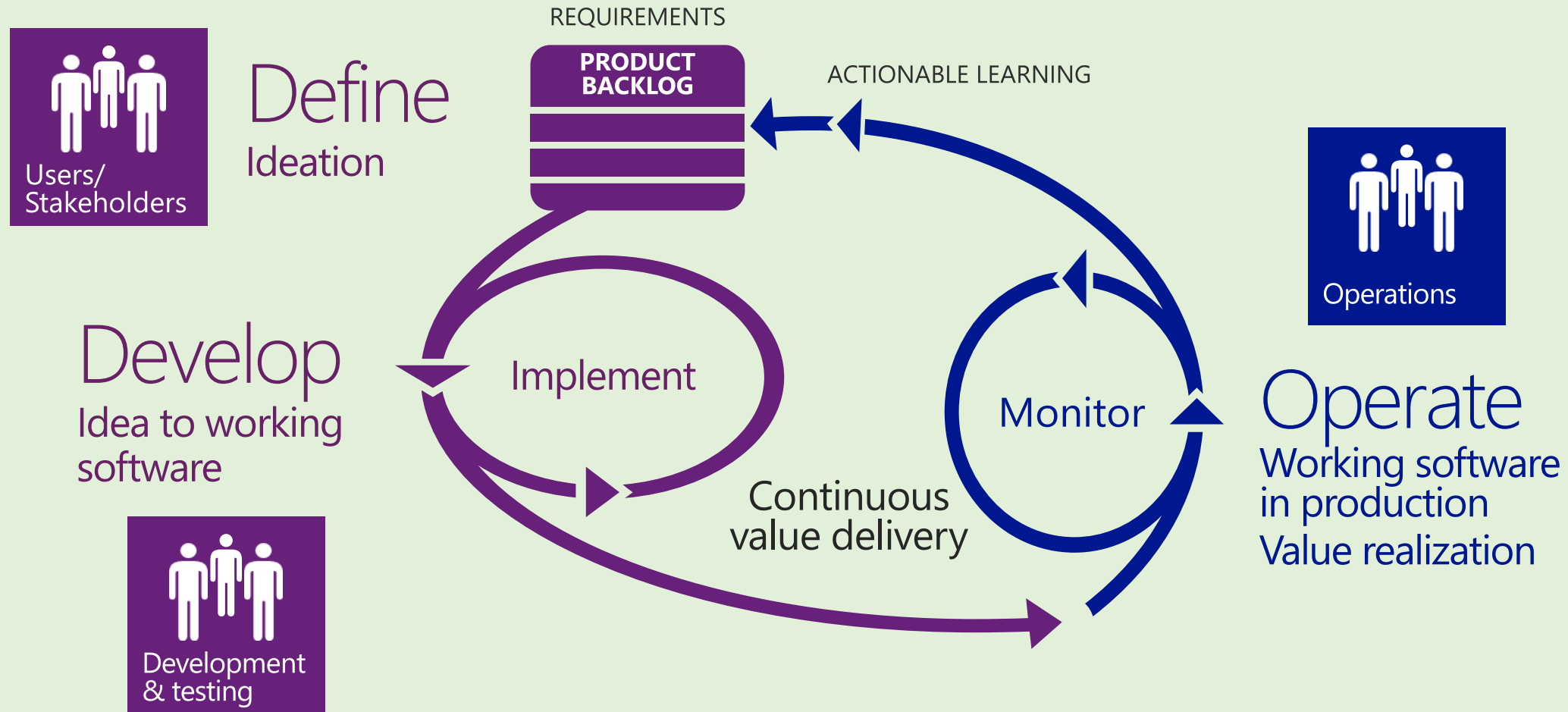
Martin Fowler, ThoughtWorks

DevOps Benefits & ROI

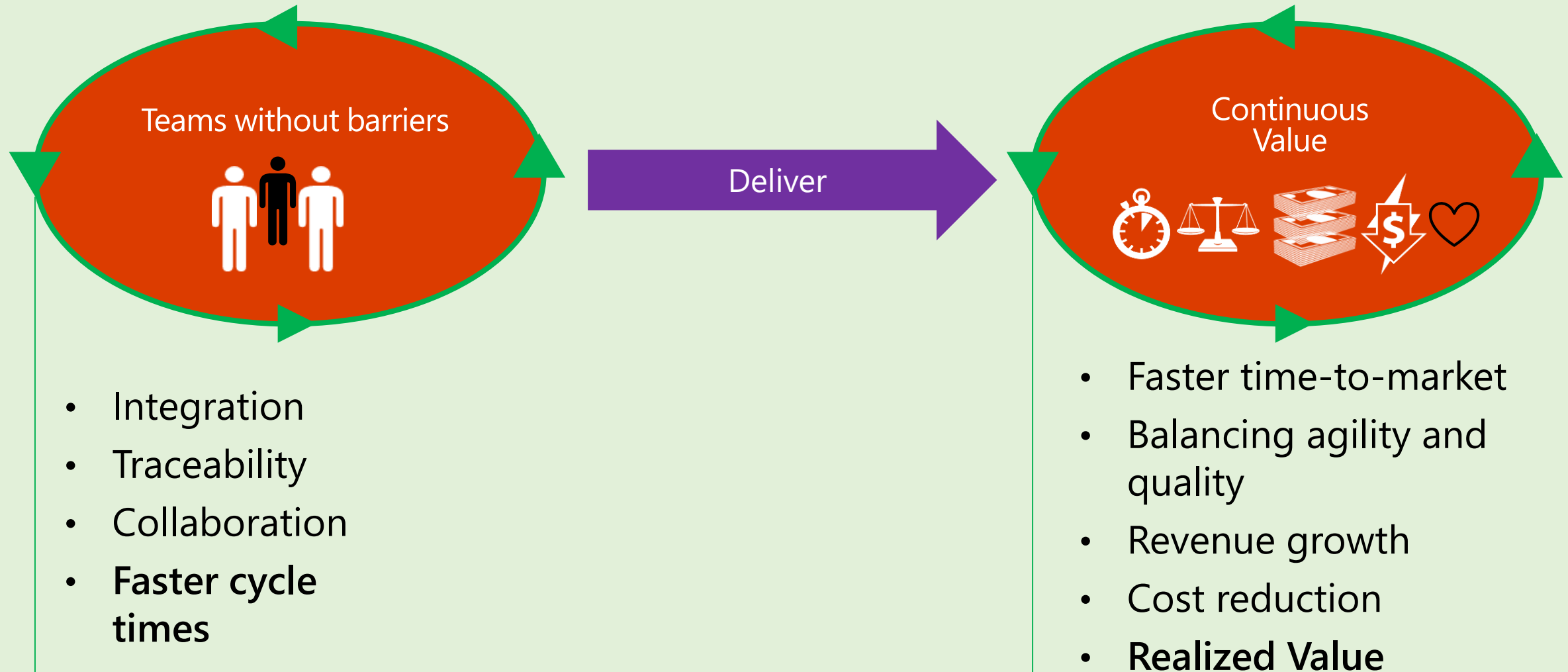
Source- Forrester Research Inc.

- **Improved IT operations efficiency.** saving 50% of allocated IT operations support.
- **Improved developer productivity with faster, automated release processes.** Developers saw a 15% improvement in productivity.
- **Improved tester productivity with faster, automated release processes.** 20% productivity gain with their deployment of the Microsoft DevOps solution.
- **Faster recovery from failures and reduction in release risk.** reduce time-to-resolution per incident by 2 hours.
- **Reduced cycle time by as much as 99% and more frequent deployments.** increased customer satisfaction and also gained new business.
- **Faster speed-to-market for new features, products, and services for customers.** 20x more, leading to increased sales.
- **Improved release quality.** successful releases with a reduction in errors ,increase in release reliability, less time on remediation of release errors.

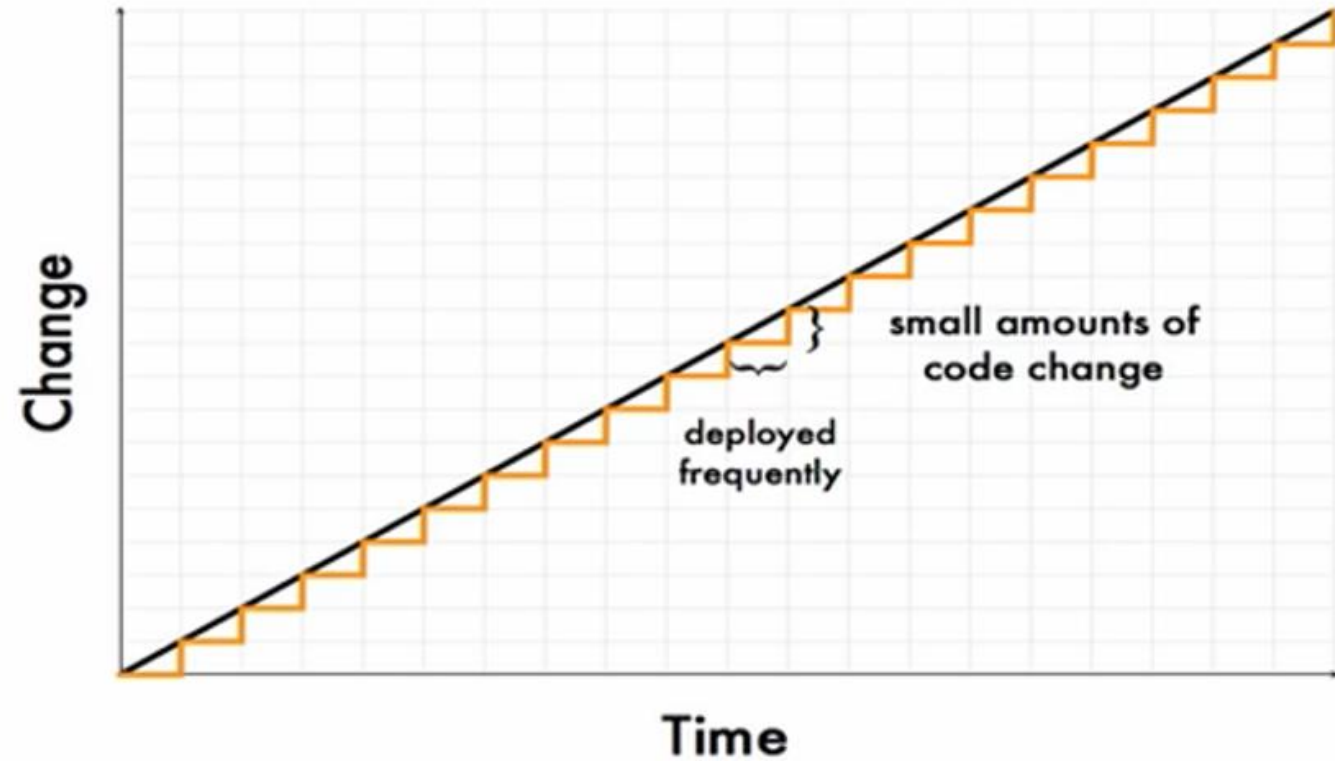
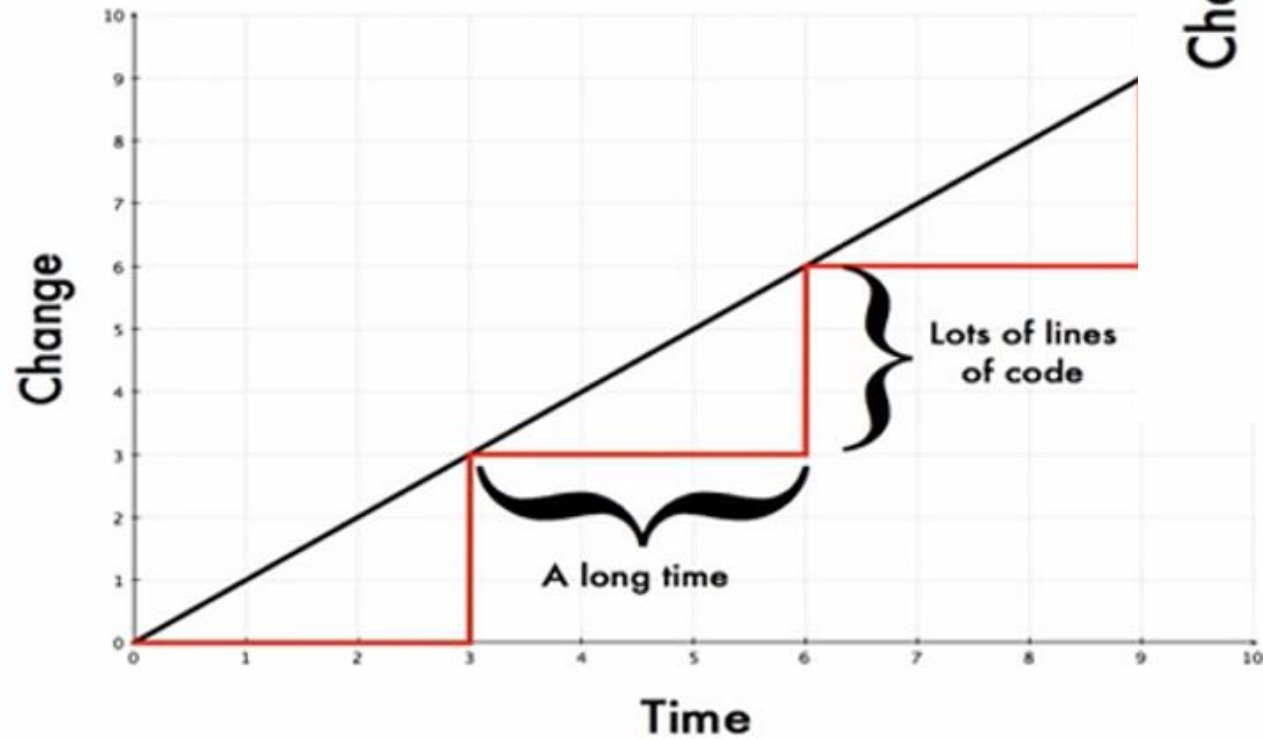
DevOps Workflow from Planning to Release



Teams without barriers realize business value.



Delivery Cycles



Agile Manifesto

Agile manifesto and principles have an influence on DevOps. Let's review agile manifesto:

"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

How Agile principles apply on DevOps

- Our highest priority is to satisfy the customer through an early and continuous delivery of valuable software.

When a team adapts the DevOps culture then team works together to deliver a high-quality software to the end users consistently over and over.

- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

DevOps promotes short delivery cycles over long delivery cycles. Short delivery cycles have small amount of change which is far easier to deploy and test and detect impact of changes.

- Business people and developers must work together daily throughout the project.

DevOps enable the business people to have look at the product early and provide candid feedback. DevOps also provides an easy and effective way of adding approval step in the CI/CD pipeline where business / product owner can approve changes for deployment to other environments.

- Working software is the primary measure of progress.

At the end business values working software having reliable feature delivered on time to end users. DevOps practices enable the software development teams to deliver working software on time and provided proper channel for feedback between various stages of the DevOps workflow.

DevOps Practices

- Configuration Management
- Release Management
- Continuous Integration
- Continuous Deployment
- Infrastructure as Code
- Test Automation
- Application Performance Monitoring

Any Language, Any Platform

Program in any language

.NET
Java
Python
Ruby
Nodejs

...

Develop on any OS

Linux
Mac
Windows

Deploy to any Platform

- Android
- iOS
- Windows Phone
- Docker
- Azure
- AWS
- Linux
- Mac
- ...

Microsoft Ecosystem

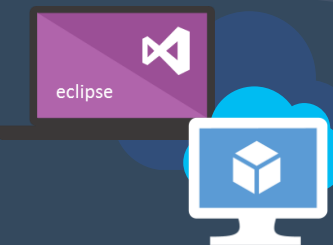
People | Process | Products



01

Develop

Developer Workstation



Team Collaboration

TFS / VSTS



Workstations - On-Premises | Hybrid | Cloud

02

Build & Test

Build/CI

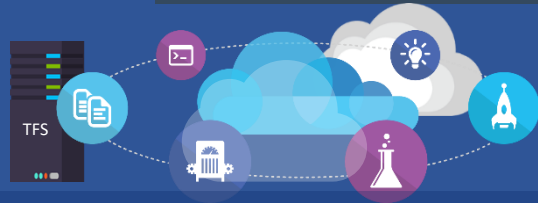
TFS / VSTS

Release Management for Visual Studio

Test

TFS / VSTS

Microsoft Test Manager



ALM Services - On-Premises | Hybrid | Cloud

03

Deploy

Release

Microsoft System Center

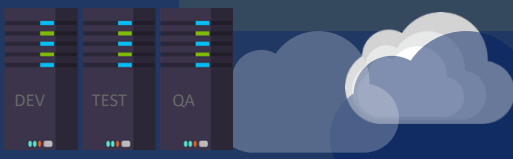
Release Management for Visual Studio

Automation Service

PowerShell | WAML

Azure Resource Management

xPlat Command Line



Environments - On-Premises | Hybrid | Cloud

04

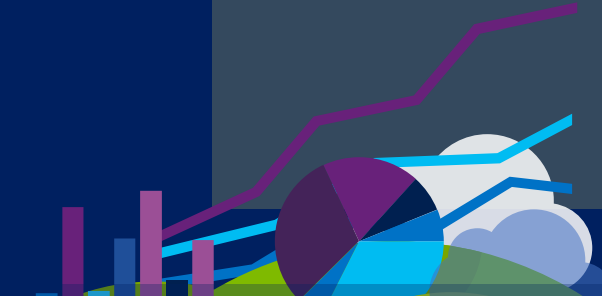
Monitor & Learn

Monitor

Microsoft System Center

VSTS

Application Insights



Monitoring - On-Premises | Hybrid | Cloud

Hybrid Ecosystem

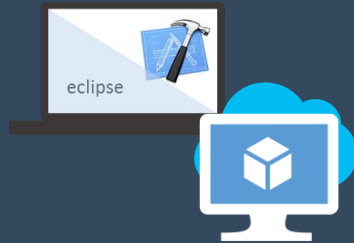
People | Process | Products



01

Develop

Developer Workstation



Team Collaboration

GitHub
CodePlex

02

Build & Test

Build/CI

gradle

GRUNT

Jenkins

Hudson

Test

gradle

GRUNT

03

Deploy

Configuration



Release

gradle

GRUNT

Jenkins

Hudson

VAGRANT

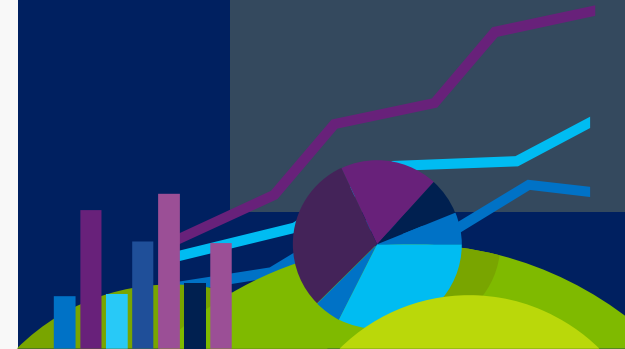
04

Monitor & Learn

Monitor

Nagios

ZABBIX

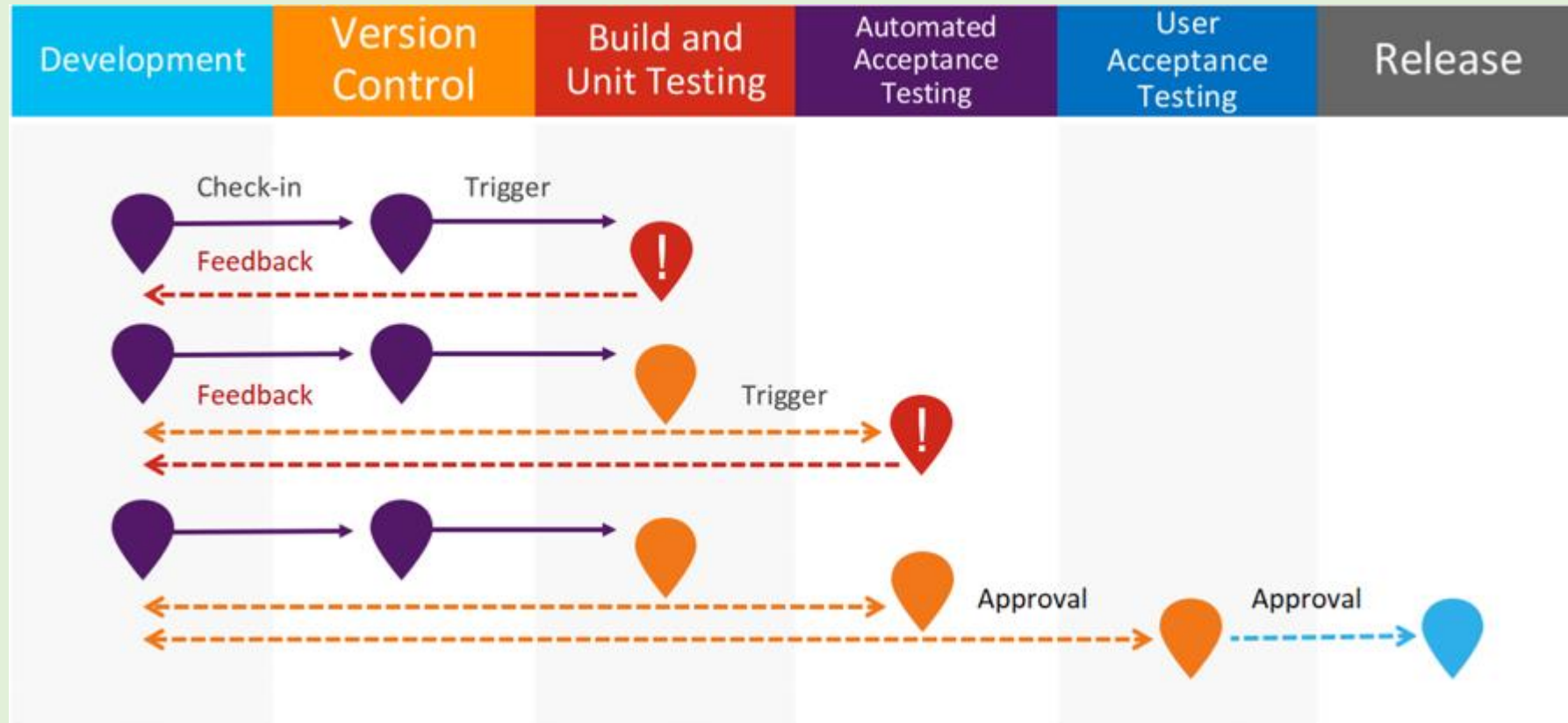


This graphic shows OSS and partner products that are integrated with the Microsoft DevOps solution

DevOps Metrics



DevOps Automation



With / Without DevOps

Category	Without DevOps	With DevOps
Quality of code check-ins	Unknown	Validated through unit tests
Environment creation	Manual	Automated
Deployment Frequency	1-2 times a month	Several times per day
Deployment Process	Requires meeting/planning	Push button deployment
Deployment Validation	Manual	Automated
Monitoring	Manual to None	Health & Performance Monitoring
Dev & Ops Relationship	Culture of Blame	Culture of Trust

DevOps Toolchain

DevOps is technology agnostic and any development environment on any platform can fully adopt DevOps culture and can continuously deliver quality software to their customers.

Planning and Analysis

Capturing and tracking (TFS, VSTS, JIRA, ServiceNow).

Documentation or Wiki page (Microsoft Teams, SharePoint, Confluence).

Collaboration (Slack, HipChat, Microsoft Teams).

Design and Development

SCM (TFS, VSTS, Subversion, Git, Mercurial).

IDE (Eclipse, IntelliJ, Visual Studio).

Build and Release (CI/CD)

Repository management (Artifactory, Nexus).

Build tools (MSBuild, Jenkins, Bamboo).

Configuration management (Chef, Puppet, Ansible).

Cloud (AWS, Azure, OpenStack).

Containers (Docker).

Integration and Testing

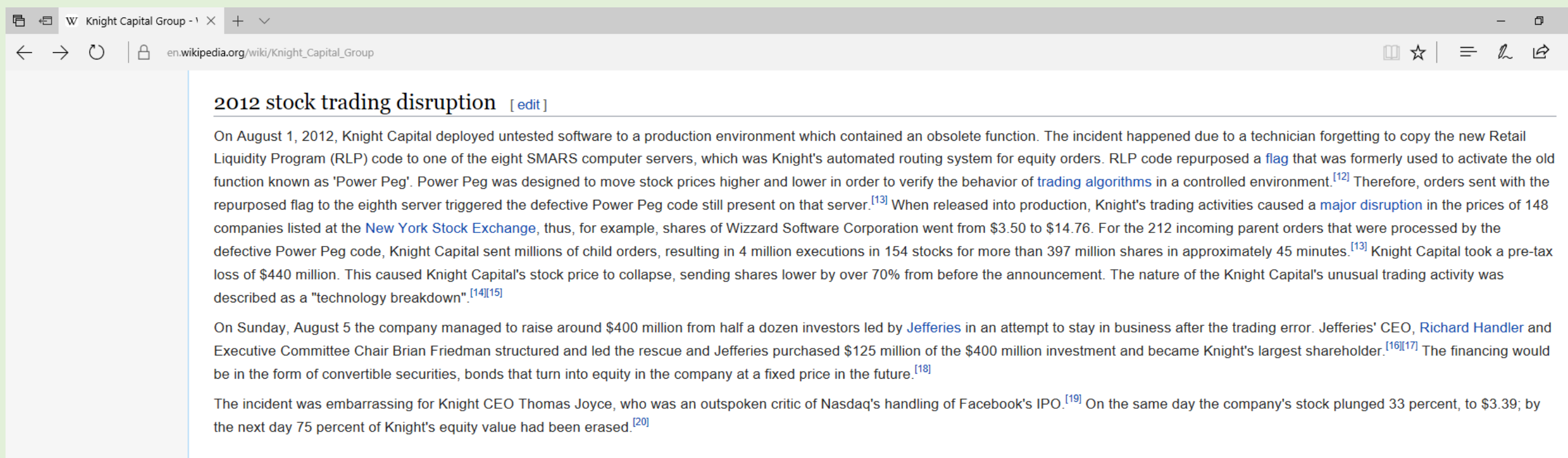
Source code verification (SonarQube).

Security testing (HP Fortify).

Functional testing (MSTest, NUnit, JUnit, Cucumber, Selenium).

Performance testing (SOASTA, Apache Test Bench, Microsoft Load and Performance Test).

Cost of NOT Deploying to just ONE server



The screenshot shows a web browser window with a single tab titled 'Knight Capital Group'. The address bar displays 'en.wikipedia.org/wiki/Knight_Capital_Group'. The article title is '2012 stock trading disruption' with an '[edit]' link. The text describes a major trading error by Knight Capital on August 1, 2012, where untested software caused a significant stock price disruption. It mentions the 'Power Peg' function, the repurposed 'flag', and the resulting loss of \$440 million. The article also notes the company's recovery efforts on August 5 and the impact on the company's stock price.

2012 stock trading disruption [edit]

On August 1, 2012, Knight Capital deployed untested software to a production environment which contained an obsolete function. The incident happened due to a technician forgetting to copy the new Retail Liquidity Program (RLP) code to one of the eight SMARS computer servers, which was Knight's automated routing system for equity orders. RLP code repurposed a [flag](#) that was formerly used to activate the old function known as 'Power Peg'. Power Peg was designed to move stock prices higher and lower in order to verify the behavior of [trading algorithms](#) in a controlled environment.^[12] Therefore, orders sent with the repurposed flag to the eighth server triggered the defective Power Peg code still present on that server.^[13] When released into production, Knight's trading activities caused a [major disruption](#) in the prices of 148 companies listed at the [New York Stock Exchange](#), thus, for example, shares of Wizzard Software Corporation went from \$3.50 to \$14.76. For the 212 incoming parent orders that were processed by the defective Power Peg code, Knight Capital sent millions of child orders, resulting in 4 million executions in 154 stocks for more than 397 million shares in approximately 45 minutes.^[13] Knight Capital took a pre-tax loss of \$440 million. This caused Knight Capital's stock price to collapse, sending shares lower by over 70% from before the announcement. The nature of the Knight Capital's unusual trading activity was described as a "technology breakdown".^{[14][15]}

On Sunday, August 5 the company managed to raise around \$400 million from half a dozen investors led by [Jefferies](#) in an attempt to stay in business after the trading error. Jefferies' CEO, [Richard Handler](#) and Executive Committee Chair Brian Friedman structured and led the rescue and Jefferies purchased \$125 million of the \$400 million investment and became Knight's largest shareholder.^{[16][17]} The financing would be in the form of convertible securities, bonds that turn into equity in the company at a fixed price in the future.^[18]

The incident was embarrassing for Knight CEO Thomas Joyce, who was an outspoken critic of Nasdaq's handling of Facebook's IPO.^[19] On the same day the company's stock plunged 33 percent, to \$3.39; by the next day 75 percent of Knight's equity value had been erased.^[20]

Thank You

Vidya Vrat Agarwal

www.MyPassionFor.Net | [@dotNetAuthor](#)

<https://www.linkedin.com/in/vidyavrat/>