DevOps: What, Why & How?

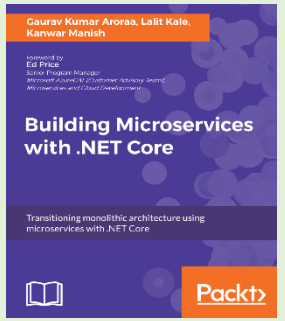
Vidya Vrat Agarwal

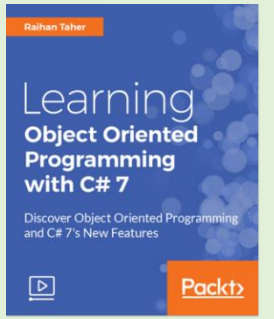
www.MyPassionFor.Net | @dotNetAuthor

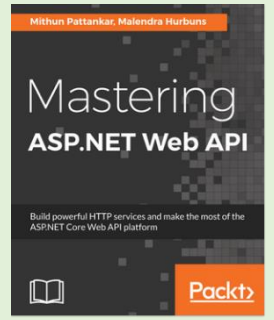
https://www.linkedin.com/in/vidyavrat/

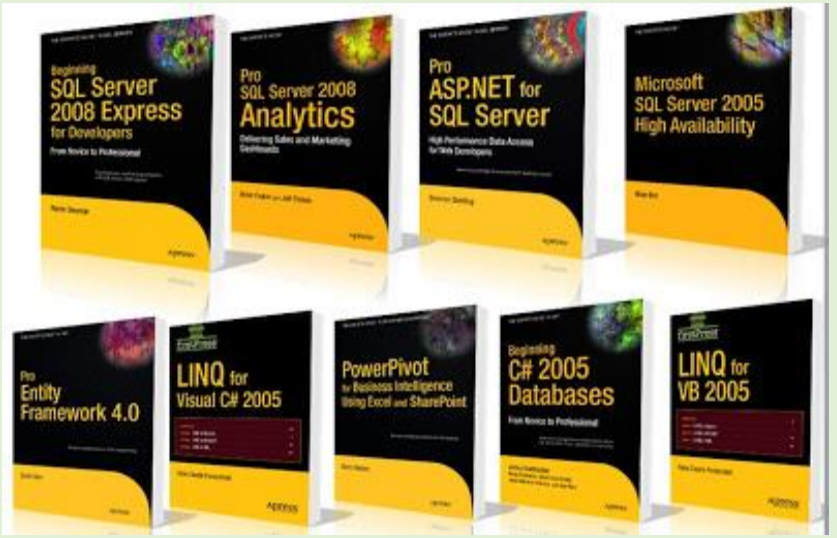
About Me

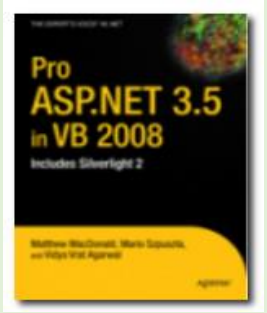
- 18+ years of industry experience
- Principal Architect with T-Mobile
- Microsoft MVP Since 2014 | C# Corner MVP Since 2013
- TOGAF Certified Architect
- Certified Scrum Master (CSM)
- Certified Associate in Project Management (CAPM)
- Microsoft Certified (MCT, MCSD / MCAD .NET, MCTS etc.)
- Published Author (5), and Technical Reviewer (over a dozen)

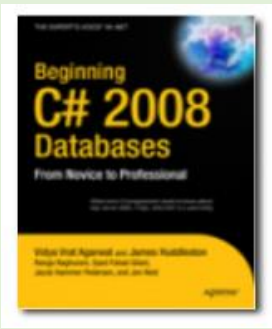


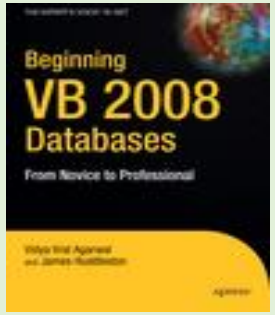


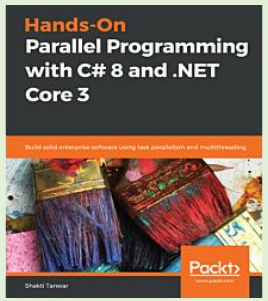


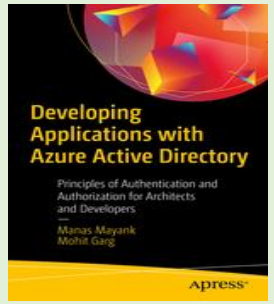






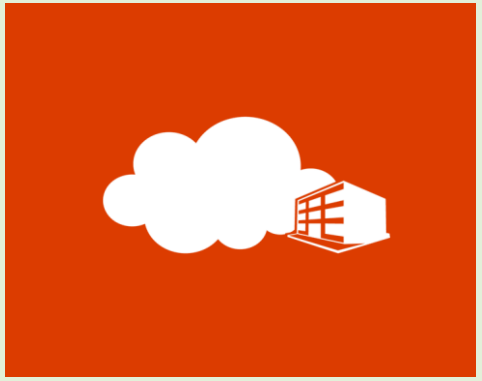






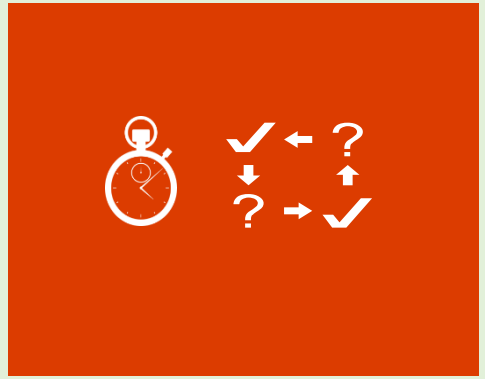
Industry Trends for IT

Heterogeneous devices and platforms



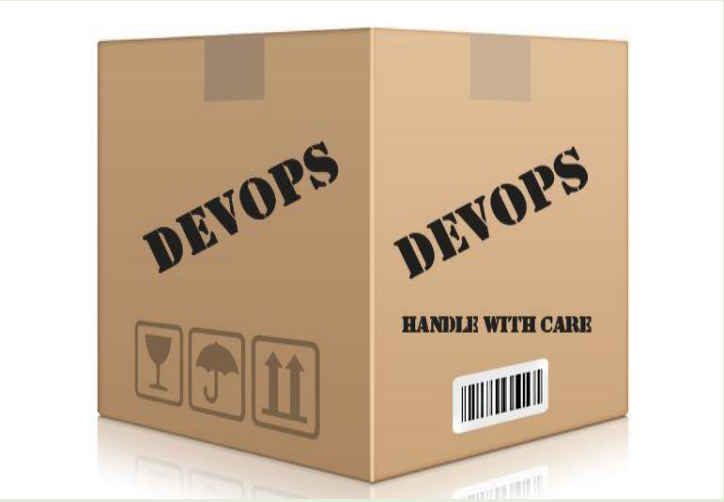
Hybrid IT

Distributed teams

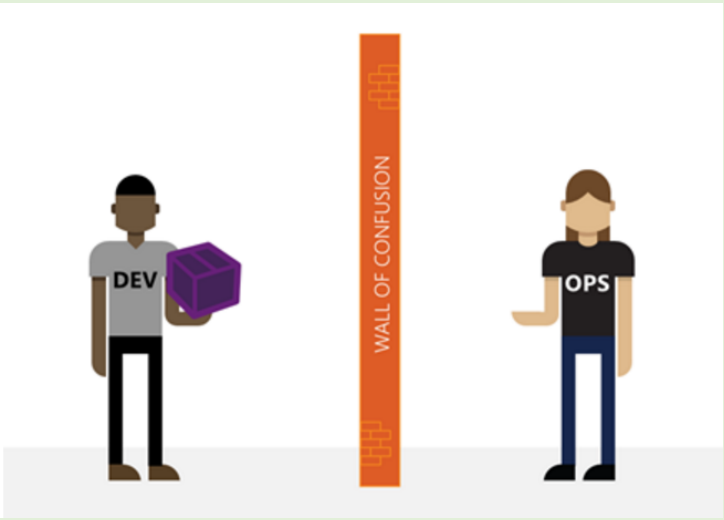


Delivery agility

DevOps & It's key values

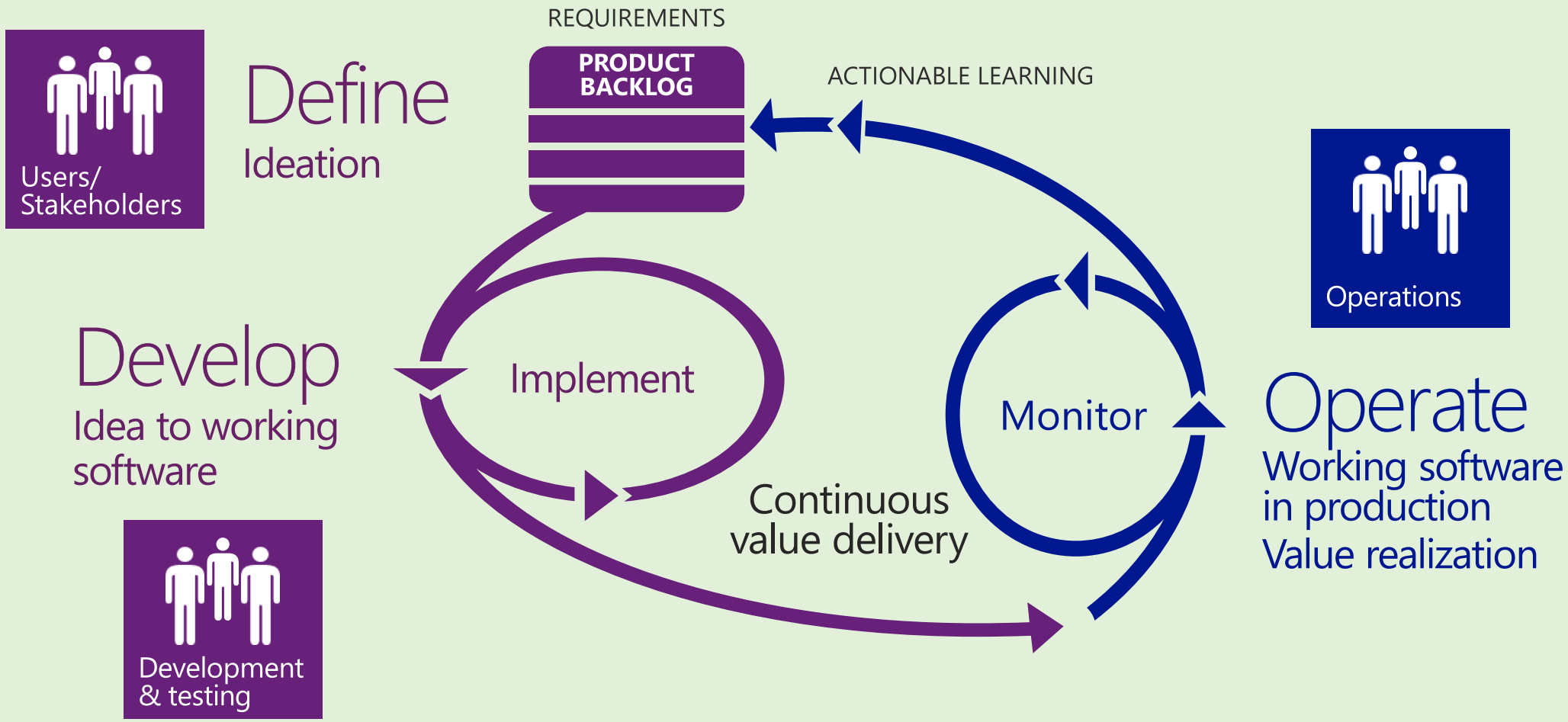


- You cannot buy DevOps and install it.
- "DevOps is the union of people, process, and products to enable "continuous delivery" of value to our end users." – *Donovan Brown, MSFT*

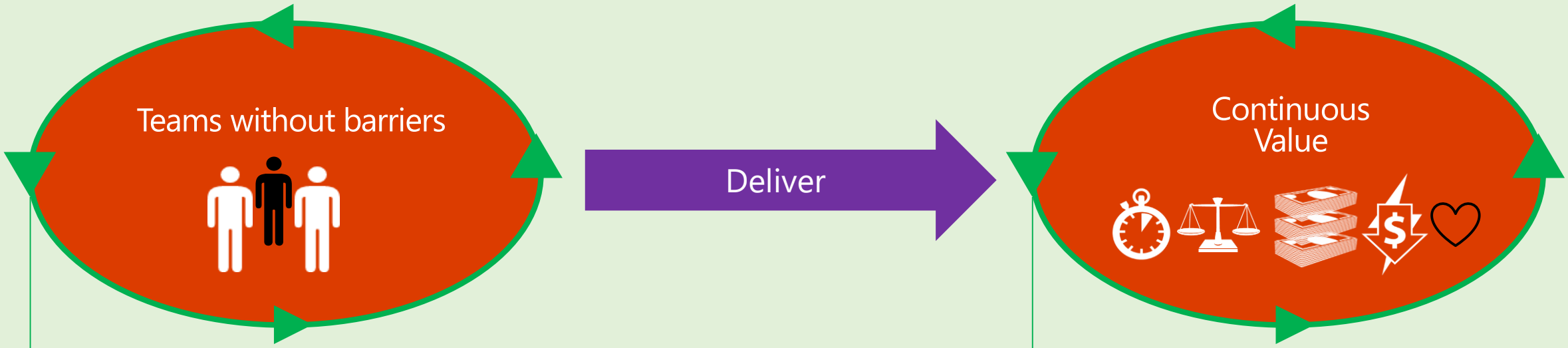


- Breaks walls of confusion between teams & fosters better communication and collaboration throughout the application development lifecycle.
- Delivers software more frequently & produces higher quality software.
- Shortens lead time and software delivery cycles.

DevOps Workflow from Planning to Release



Teams without barriers realize business value.

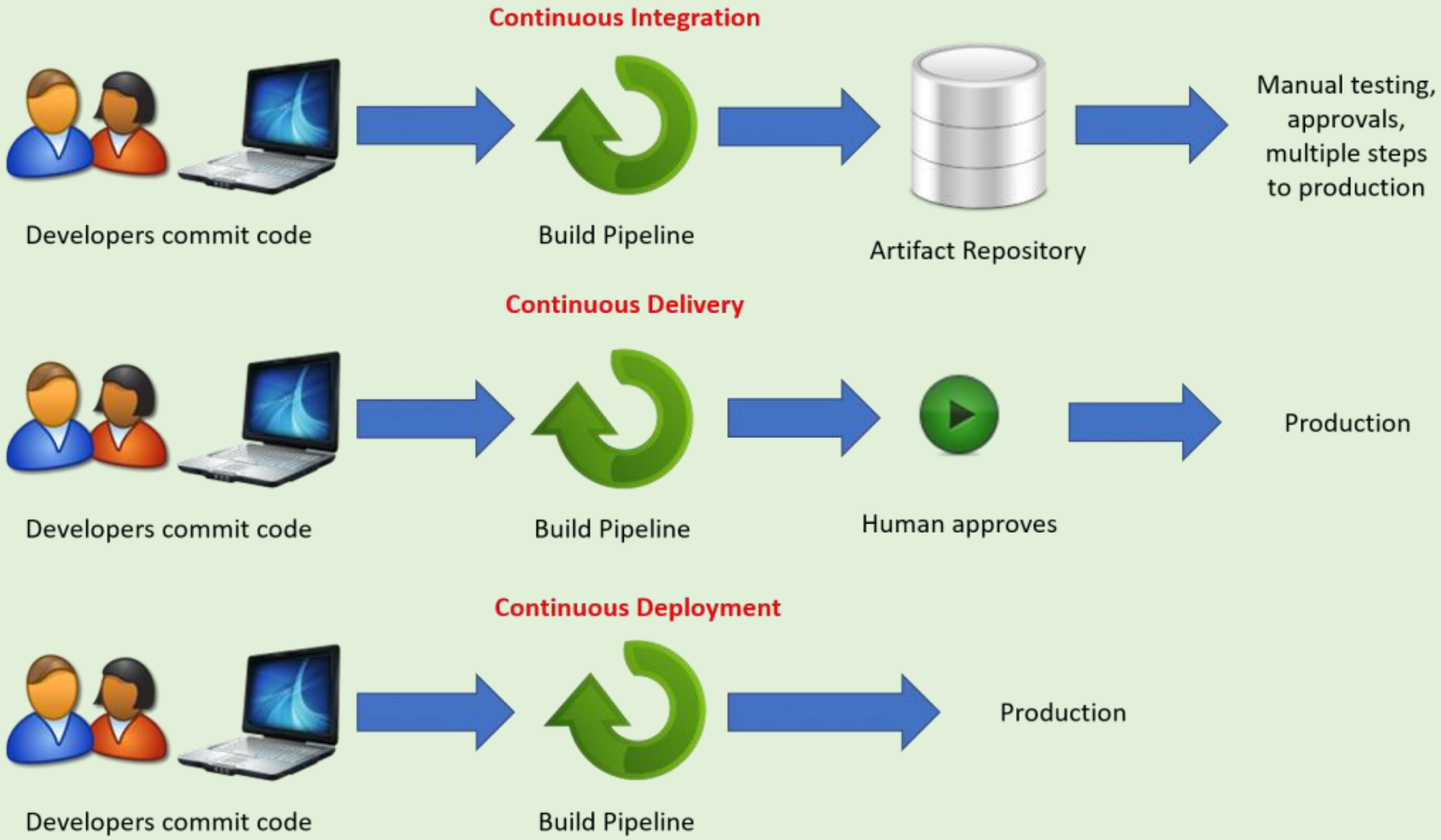


- Integration
- Traceability
- Collaboration
- **Faster cycle times**

- Faster time-to-market
- Balancing agility and quality
- Revenue growth
- Cost reduction
- **Realized Value**

DevOps Practices

- Continuous Integration
- Continuous Delivery
- Continuous Deployment
- Infrastructure as Code
- Test Automation
- Application Performance Monitoring


Continuous Integration
Developers commit code
Build Pipeline
Artifact Repository
Manual testing, approvals, multiple steps to production
Continuous Delivery
Developers commit code
Build Pipeline
Human approves
Production
Continuous Deployment
Developers commit code
Build Pipeline
Production

Any Language, Any Platform

Program in any language

- .NET
- Java
- Python
- Ruby
- Nodejs
- …

Develop on any OS

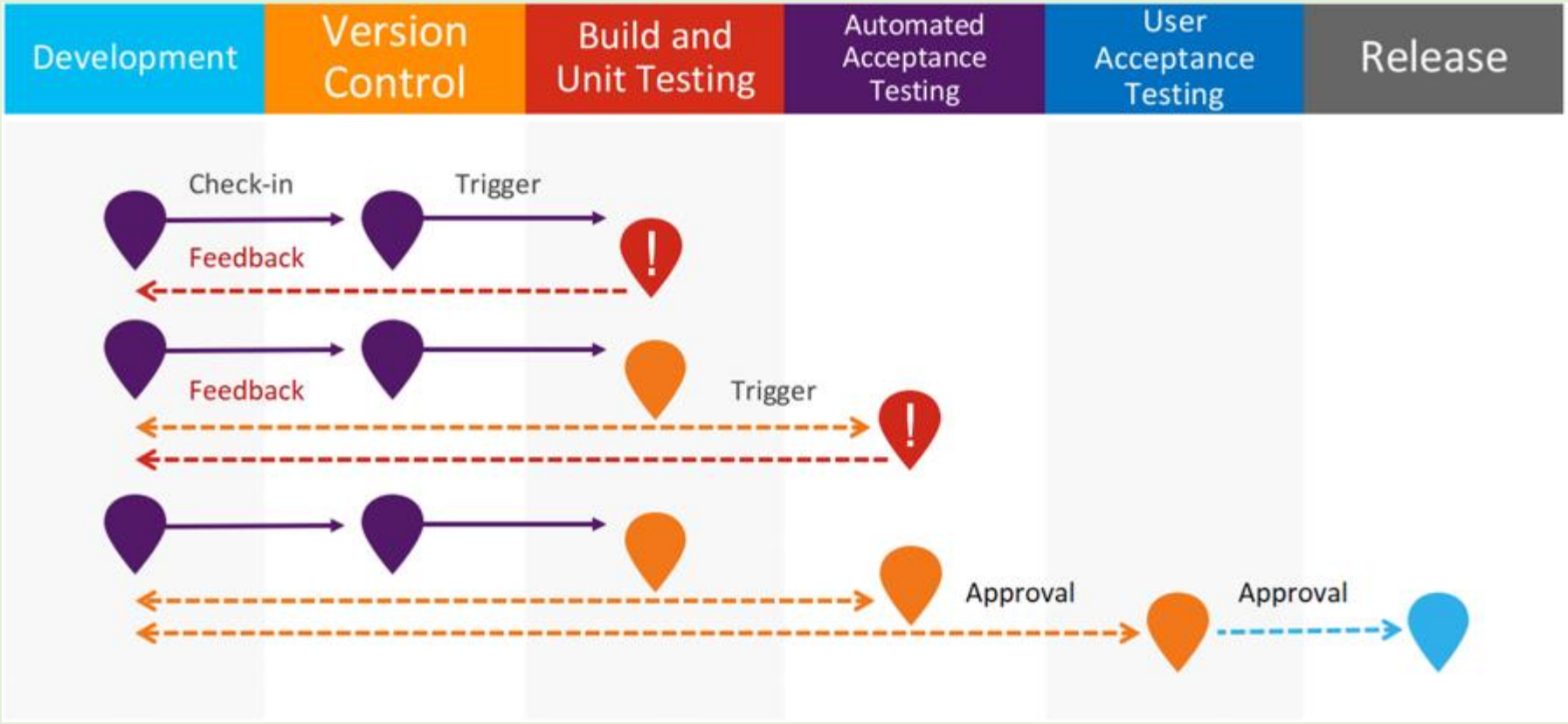
- Linux
- Mac
- Windows

Deploy to any Platform

- Android
- iOS
- Docker
- Kubernetes
- Azure
- AWS
- Serverless
- APIs
- Web Apps

DevOps Metrics

DevOps Automation



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 (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).

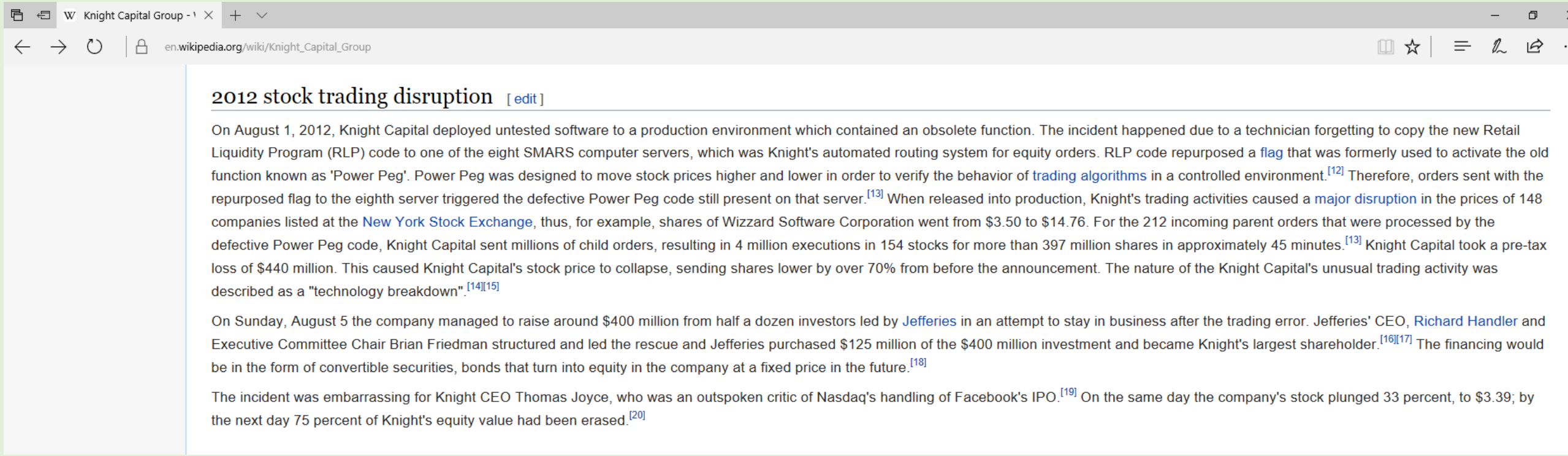
Integration and Testing

Source code verification (SonarQube).
Security testing (HP Fortify).
Functional testing (MSTest, NUnit, JUnit, Cucumber, Selenium).
Performance testing (Apache Test Bench, Microsoft Load and Performance Test).

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

Cost of NOT Deploying to just ONE server



https://www.bugsnag.com/blog/bug-day-460m-loss

Thank You

Vidya Vrat Agarwal

www.MyPassionFor.Net | @dotNetAuthor

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