

INTRODUCTION TO DEVOPS

Dan Piessens

Twitter: @dpiessens

ABOUT ME

- Senior Consultant for Centare
- 13 Years Experience as Developer, IT Consultant, Architect, Trainer, Coach
- 2008 — 2013 MS Patterns & Practices Champion



WHAT IS DEVOPS?



IT, DOES THIS DESCRIBE YOU?

- We always seem to be reacting to application problems . . .
- Deployments take forever and we need to work overtime to finish them . . .
- There's not enough time to handle development's needs and the rest of the business . . .

THE REAL VALUE OF SOFTWARE



“Every software feature is an experiment until feedback is obtained from end users in production.”

WHAT DEVOPS IS

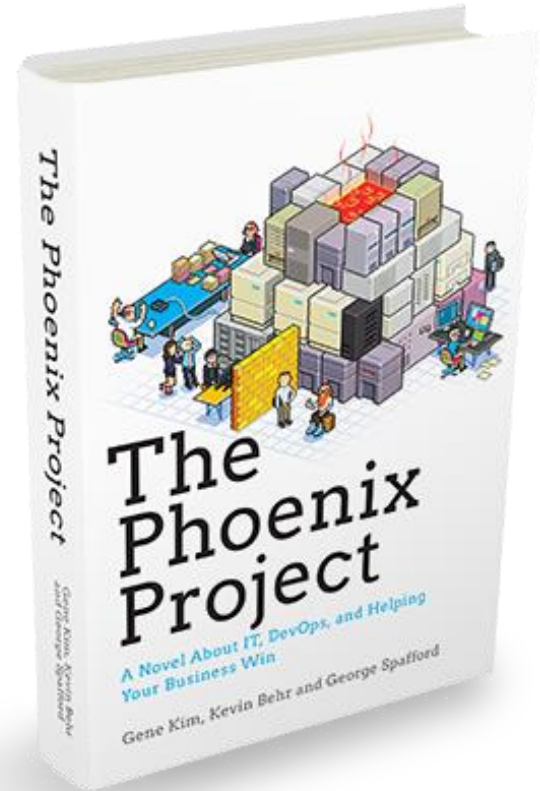
- Coordination of development and IT operations to reduce impediments around application deployment, maintenance and feedback.
- Focus on becoming proactive and recognizing the value of working software in production.

WHAT DEVOPS IS NOT

- Does not require the creation of a new team
- Does not imply giving up control to development
- Not an excuse to ignore security, controls, etc. to move faster.

"THE GOAL" FOR DEVOPS AND IT

- Written by Gene Kim, Kevin Behr and George Spafford
- Explains principles of DevOps as a story
- Illustrates dysfunctions between the business / development / IT

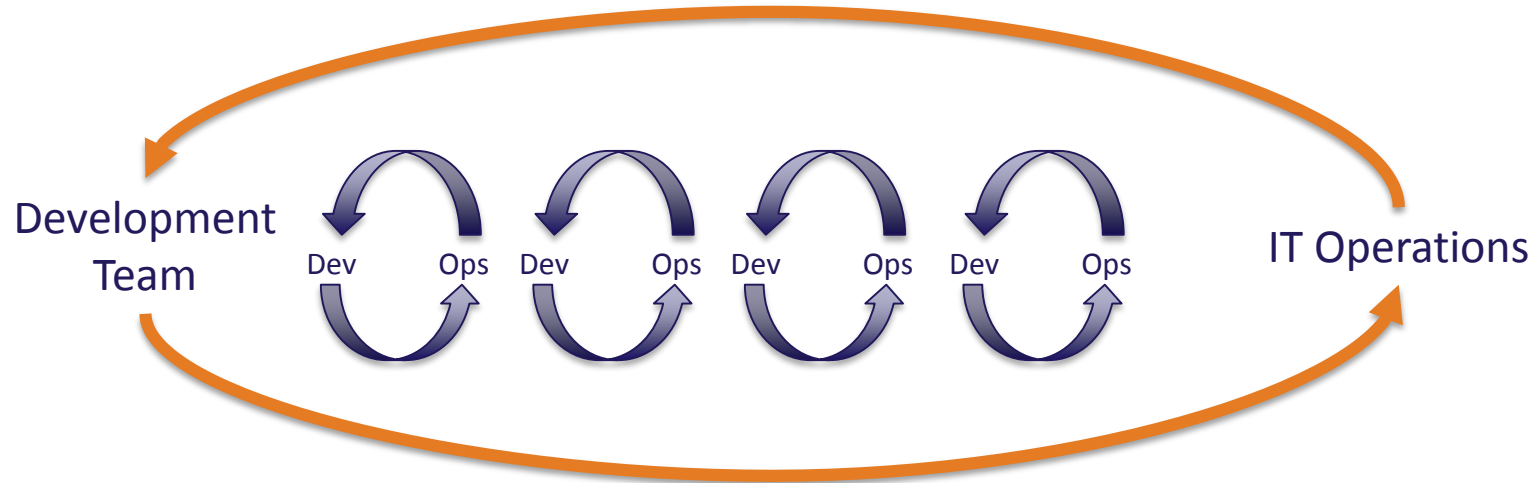


TYPES OF WORK



- Business/Development Projects
- IT Projects
- Changes / Patches
- Unplanned Work!

THE THREE WAYS



Gene Kim, "The Phoenix Project"

EMBRACING CHANGE



- Your experiments will fail!
- Fail Fast
- Iterate quickly
- Focus on the learning point, not the failure

CONTINUOUS EVERYTHING



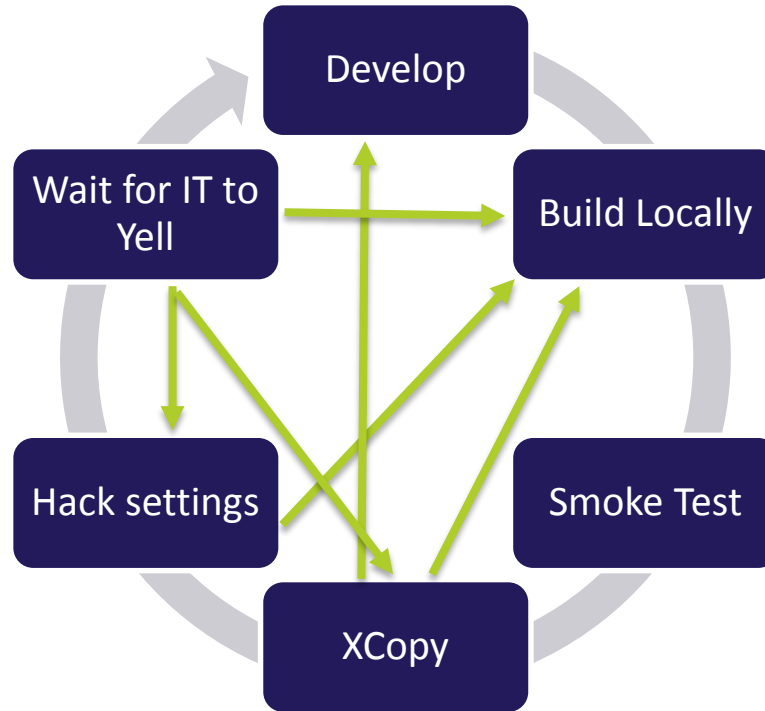
THE CONTINUOUS CONTINUUM

Continuous...	What it Does	Started By	Supporting Tools
Integration	Builds and Asserts Code Quality	No-One (Automated)	Bamboo Jenkins TeamCity TFS
Deployment	Manages Application Releases to an Environment	Anyone (Dev, Qa, Business)	Octopus Deploy MS Release Manager
Delivery	Makes New Functionality Available to Users	Business	Feature Toggles 3 Stage Updates Canary Servers

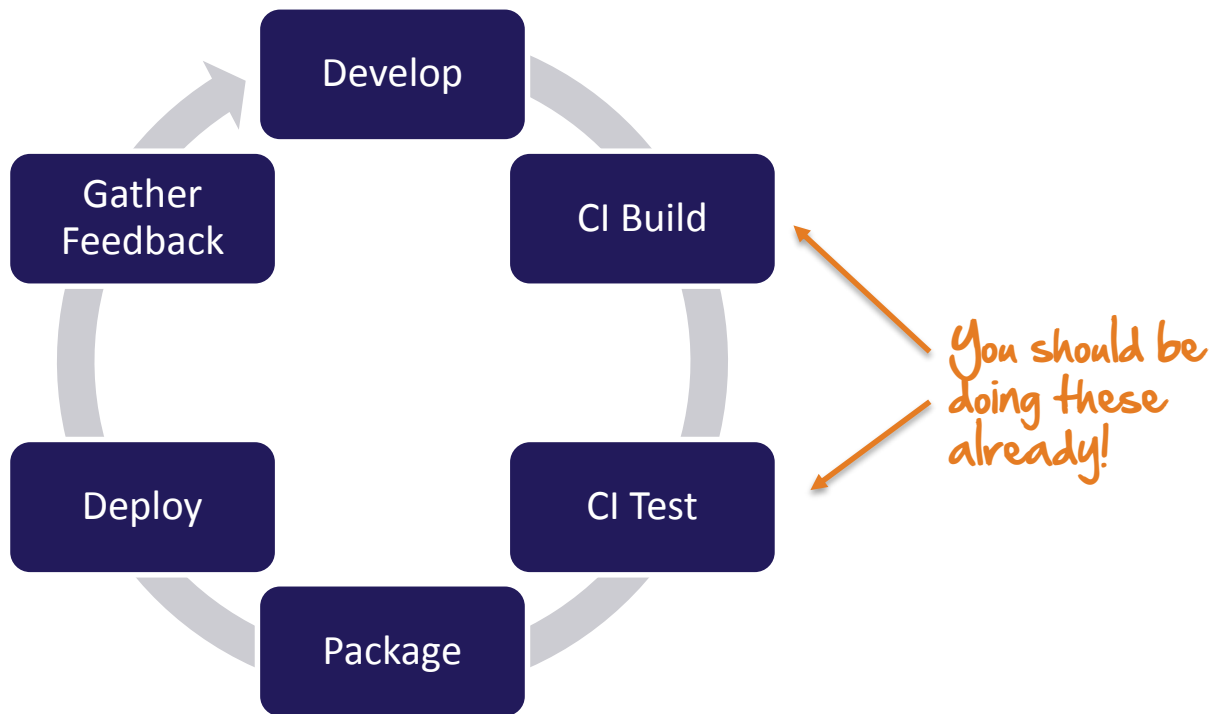
WHY DO DEPLOYMENTS FAIL?

- Large Work Batches
- Large Batches \neq Deployment Size
- How often do you deploy your software?

MOST DEPLOYMENTS TODAY



THE IDEAL DEPLOYMENT CYCLE



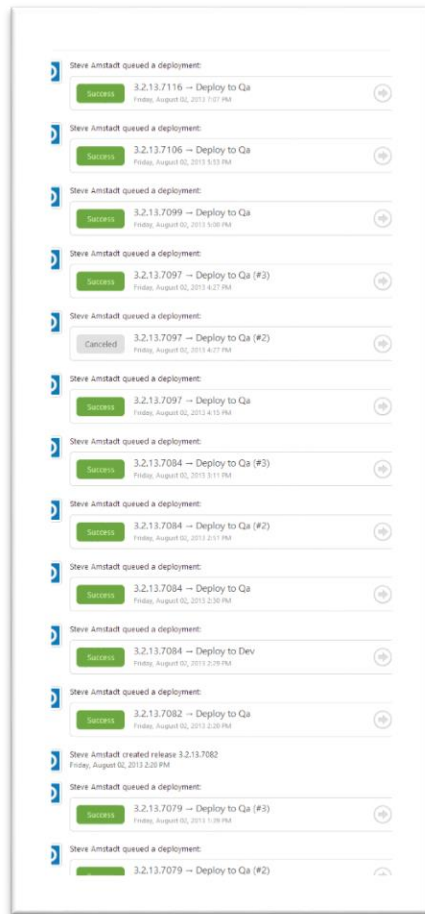
WHAT'S THE GOAL?

■ “No Brainer” Deployments

- Done with thought, just not work
- 10+ Deployments Per Day

■ Fail Fast!

15
Deploys
In One
Day!



WHAT MAKES THIS WORK?

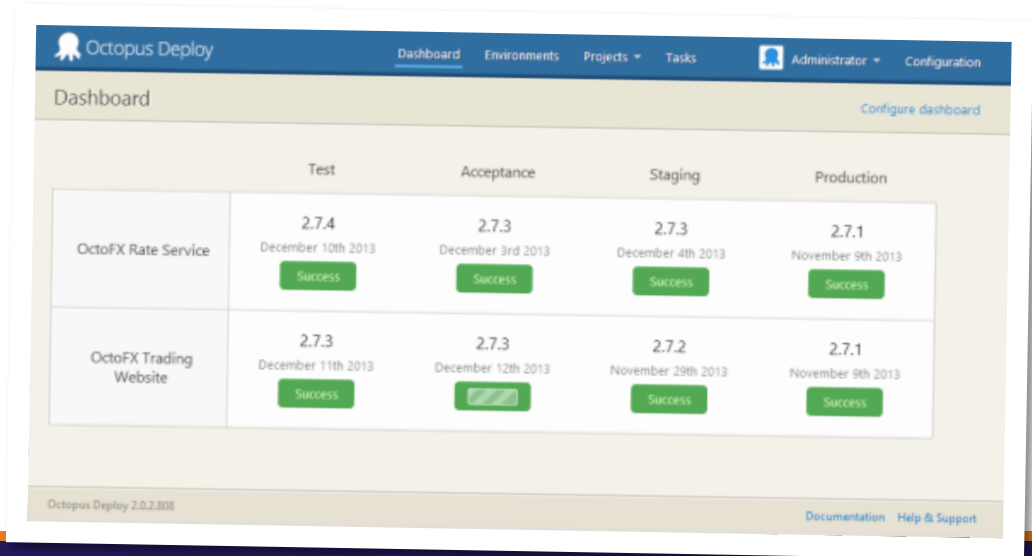
- Small Work Batches
- Automated Quality Gates
 - Unit Tests, Code Coverage, Quality Checkers (FxCop, etc.)
- Repeatable Process
 - Goal: No Manual Steps During Deployment

CONTINUOUS INTEGRATION

- Ensures consistent builds / quality
 - Eliminates “works on my machine”
- Creates a shared sense of responsibility
- Must run **quickly** to be effective
- Package what you test

DEPLOYMENT: OCTOPUS DEPLOY

- Flexible Deployment Tool
- Focused on Windows Infrastructure
 - Web Applications
 - Windows Services
 - Click Once Apps
 - Databases
 - PowerShell



The screenshot shows the Octopus Deploy dashboard. At the top is a navigation bar with the Octopus Deploy logo, a menu (Dashboard, Environments, Projects, Tasks), and user information (Administrator, Configuration). Below the navigation bar is a 'Dashboard' section with a 'Configure dashboard' link. The main content area displays a table of deployment results for two services: 'OctoFX Rate Service' and 'OctoFX Trading Website'. The table has four columns representing environments: Test, Acceptance, Staging, and Production. Each cell in the table shows the version number, the deployment date, and a status indicator (a green button labeled 'Success' or a green button with a diagonal line).

	Test	Acceptance	Staging	Production
OctoFX Rate Service	2.7.4 December 10th 2013 Success	2.7.3 December 3rd 2013 Success	2.7.3 December 4th 2013 Success	2.7.1 November 9th 2013 Success
OctoFX Trading Website	2.7.3 December 11th 2013 Success	2.7.3 December 12th 2013 Success	2.7.2 November 29th 2013 Success	2.7.1 November 9th 2013 Success

Octopus Deploy 2.0.2.808

[Documentation](#) [Help & Support](#)

IT'S NOT JUST ABOUT SOFTWARE

- Automate Your Infrastructure
 - New Development Environments
 - Automated Testing
 - Disaster Recovery / Scalability
- Tooling
 - Puppet
 - Chef
 - PowerShell Desired State

CONSIDER CLOUD OPTIONS

■ Microsoft Azure

- Full Automation Available
- Secure
- Scalable
- Cost Effective

■ No “Big Bang” Approach Needed

- Hybrid solutions
- Private VPN to datacenter

FLEXIBLE RELEASES



FEATURE TOGGLES

- A mechanism to switch between features at runtime
- Separates delivery from deployment
- Typically done at the UI / Service layer
- Scary?

You already have this in your application... User Login and Authorization!

MAKING TOGGLES EXPLICIT

Brittle!

```
HomePageModel homePageModel;

if (ConfigurationManager.AppSettings["ToggleMyFeature"] == "true")
{
    homePageModel = new HomePageModel
    {
        WelcomeMessage = "Well this is different...",
        SubMessage = "Something changed, not sure what",
        Title = "Base Page"
    };
}
else
{
    homePageModel = new HomePageModel
    {
        WelcomeMessage = "Welcome to my page!",
        SubMessage = "Now this is cool :)",
        Title = "Home Page"
    };
}
```

MAKING TOGGLES EXPLICIT

Easier to Refactor

```
HomePageModel homePageModel;  
  
if (ToggleManager.IsEnabled<ToggleMyFeature>())  
{  
    homePageModel = new HomePageModel  
    {  
        WelcomeMessage = "Well this is different...",  
        SubMessage = "Something changed, not sure what",  
        Title = "Base Page"  
    };  
}  
else  
{  
    homePageModel = new HomePageModel  
    {  
        WelcomeMessage = "Welcome to my page!",  
        SubMessage = "Now this is cool :)",  
        Title = "Home Page"  
    };  
}
```

It's a Class!

TOGGLES AND DEPLOYMENT

- Toggles do stuff! = Has a performance impact
- Must correlate changes to runtime feedback
- Flip toggles via deployments
- Nuget package: FeatureSwitch

3 STAGE UPDATES

Orders	
Id	Number
1	10001
2	10002
3	10003



3 STAGE UPDATES

STEP 1: DEPLOY NEW FUNCTIONALITY DISABLED

Orders		
Id	Number	Urgent
1	10001	F
2	10002	F
3	10003	F



3 STAGE UPDATES

STEP 2: TOGGLE NEW SERVICE CODE

Orders		
Id	Number	Urgent
1	10001	F
2	10002	F
3	10003	F



Web Service

No Urgent Code
Uses Urgent Code

3 STAGE UPDATES

STEP 3: REMOVE OLD FUNCTIONALITY

Orders		
Id	Number	Urgent
1	10001	F
2	10002	F
3	10003	F



GETTING FEEDBACK



RUNTIME FEEDBACK

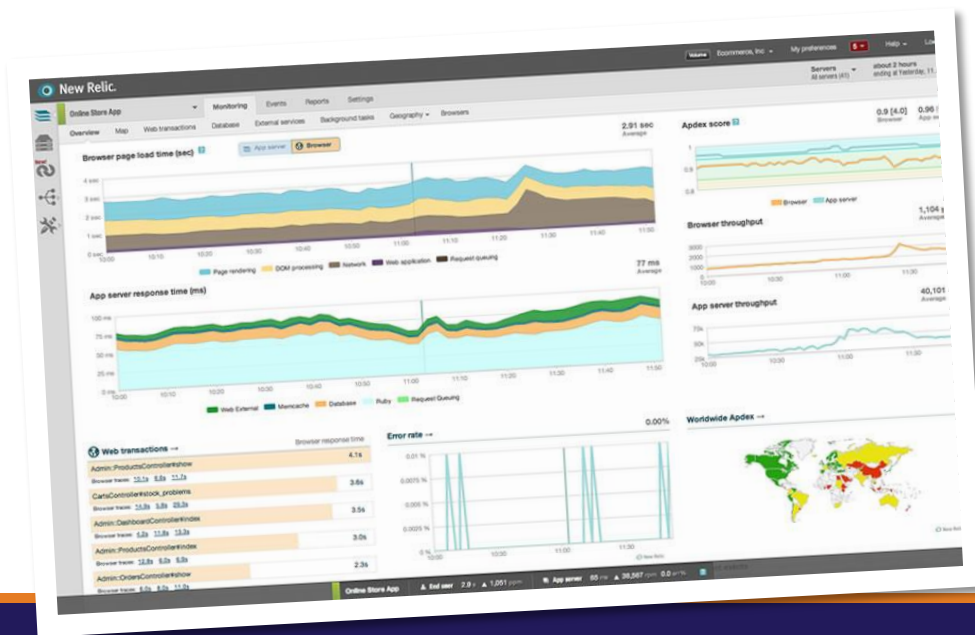
- Instrument your applications at runtime!

- Many tools available

- New Relic
- Application Insights
- Splunk
- Raygun.io

- Include User Analytics

- Google
- All above tools

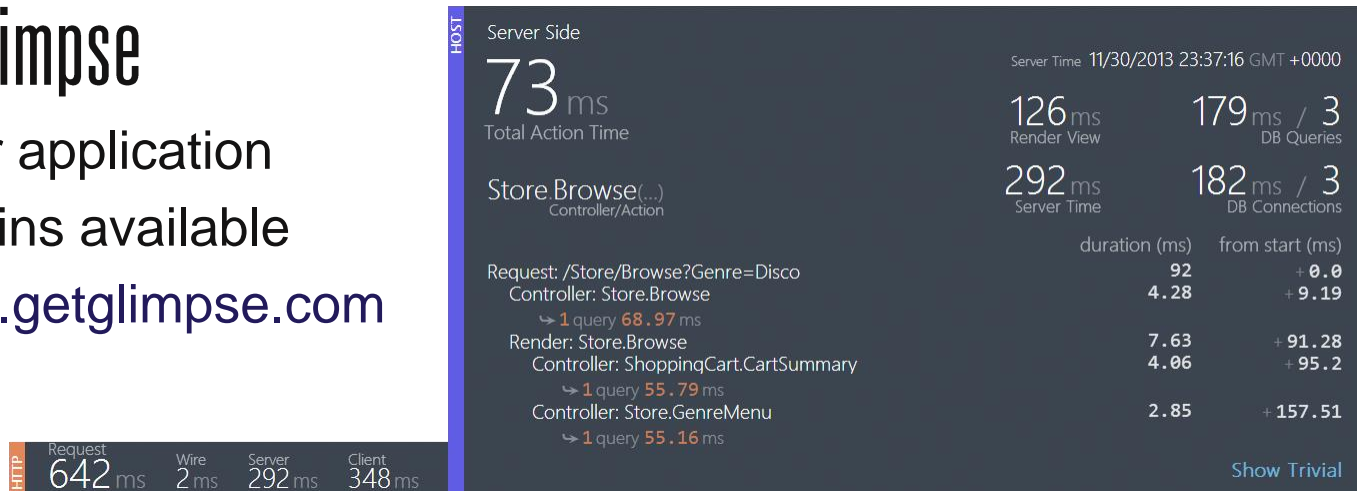


DIG DEEPER FOR DATA

- Monitoring tools have APIs for tracing / logging
- Instrument key transactions in your system
- Track performance end-over-end for deployments
- Choose a logging framework and leverage it!
 - Log to a common location

ANALYZE EARLIER

- Find errors during development
- Make performance reviews part of your DoD
- Example: Glimpse
 - Trace your application
 - Many plugins available
 - <http://www.getglimpse.com>



TAKE AWAY CONCEPTS

- Automate, Automate, Automate
- Deploy in small batches
- Provide fast feedback
- Separate deployments from releases
- Plan for failure



QUESTIONS?

Thank You!

"Deployments are like exercise, the more you do them the less it hurts"

-Dan Piessens

