



MAGNET
FORENSICS®

The Many Faces of Automation

C# Dev/Connect October 2015 Meeting

So Who is this Guy?

Indoctrinated through
Waterfall

Survived the tribulations of
the Burndown Chart

Too agile for Agile

Ascended to He Who
Walks the Path of
Automation





Why This Talk?

We talk about these a lot:

Unit tests

Integration tests

System tests

TDD/BDD/ATDD

UI tests

Why This Talk?

Where does development stop?

How does testing fit into Continuous Delivery?

Why IT Operations matter



Continuous Everything

Continuous Integration

Feedback

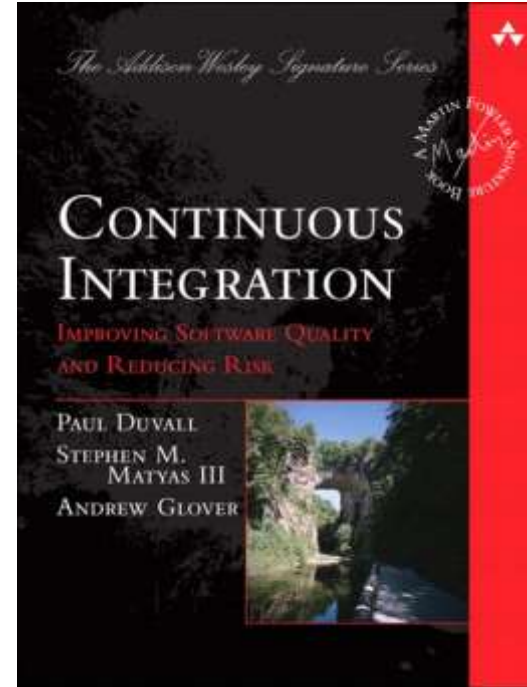
Version control

Automated build

Code inspection

Automated tests

Automated deployment



Continuous Delivery



Continuous Delivery



Continuous Delivery

Configuration management

Continuous Integration

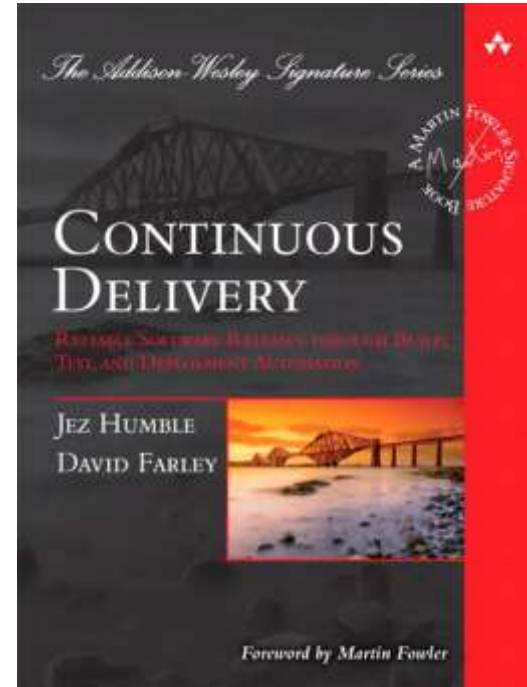
Automated tests

Automated build and deployment

Deployment pipeline

See also Kent Beck's talk on Software G
Forces:

<https://www.youtube.com/watch?v=KIkUWG5ACFY>



Continuous Deployment

An extension of Continuous Integration

Focused on deploying changes

Continuous <noun>

Why is the continuous pattern so popular?

Testing in Continuous Delivery

Testing in Continuous Delivery

- Automated testing gives feedback and answers known questions
- Human testing provides feedback and clarifies risk
- Support for all testing improves

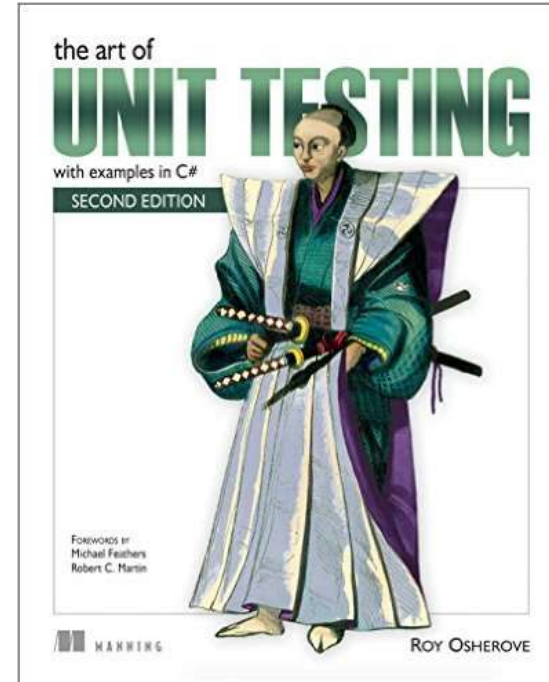
Automated Tests

Three pillars of unit automated tests:

Trustworthy

Maintainable

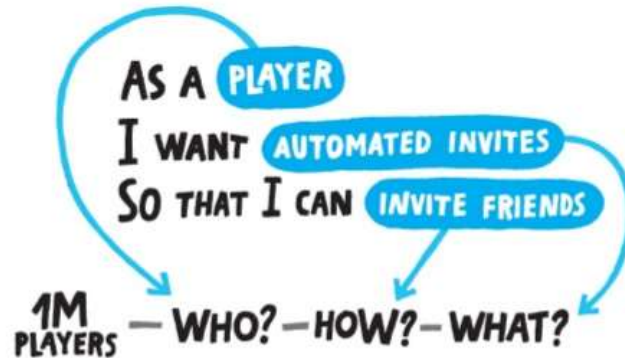
Readable



Impact Mapping

How to build an impact map

- Define business goal ("Why")
- Name key personas ("Who")
- Describe desired change in behavior ("How")
- Find features that allow desired changes ("What")



© 2012 Gojko Adzic

Acceptance Tests

- Business tests that support the team
- Automated tests that answer direct questions about risk:
 - Business acceptance criteria
 - System requirements
 - Compliance
 - Performance benchmarks
- Popular as a Given/When/Then format

Monitoring Tests

- Monitoring tests are a way to determine availability and responsiveness of a system
- Passive monitoring analyzes existing data, synthetic monitoring exercises the system

Monitoring Tests

GitHub:

```
{"status": "good", "last_updated": "2015-10-20T20:12:49Z"}
```

Facebook:

```
{  
  "current": {  
    "health": 1,  
    "subject": "Facebook Platform is Healthy"  
  },  
  "push": {  
    "status": "Complete",  
    "updated": "2015-10-19T15:52:17-07:00",  
    "id": 60668546  
  }  
}
```

Monitoring Tests

```
{  
  Version: "1.0.0",  
  Status: "good",  
  Metrics: {  
    RequestsReceived: 0,  
    ResponsesServed: 0,  
    AverageRoundTrip: 0,  
    Redis: {  
      Connection: "good"  
    }  
  },  
  LastUpdated: "2015-01-01 01:01:01 -0000"  
}
```


Manifest Validation

Is what's in the package what was deployed?

Which version of this library is in this release?

Do we clean up properly post-upgrade?

Manifest Validation

```
<manifest>
  <version>1.0.0</version>
  <jenkins>
    <jobUrl>
      http://jenkins/job/build/1
    </jobUrl>
  </jenkins>
  <assemblies>
    <assembly name="Dependency.dll">
      <checksum></checksum>
      <version>5.16.2</version>
    </assembly>
  </assemblies>
</manifest>
```

```
<?xml version="1.0" encoding="utf-8"?>
<packages>
  <package id="xunit" version="1.9.1"
targetFramework="net40" />
</packages>
```

Questions?

Thanks everyone!

You can find me at
@SpectacledBear on Twitter
<https://github.com/SpectacledBear>

