DevOps / Agile Transformation
Creating a Strategic Roadmap

Peter Eeles
Executive IT Architect
IBM DevOps Global Tiger Team
peter.eeles@uk.ibm.com
Agenda

• The Business / IT Context
• Agile, DevOps and the Evolution of Delivery Practices
• Transformation Best Practices
• Case Studies
• Getting Started
• Summary
Years till 50 Million Users

From Bank 3.0 – Brett King
Technology Trends

Big Data
Insights on new products by more efficiently interpreting massive quantities of data

Cloud
Demand for apps requires fast, scalable environments for dev and test, as well as production

Social Business
Broader set of stakeholders collaborates to deliver continuous innovation and value

Mobile
Modern workforce expects constantly updated software to connect to enterprise systems

Instrumented Products
Industry requirements demand faster response to regulations and standards, with traceability and quality

Intelligent/Connected Systems
Software component in smart products driving increased value and differentiation
Agile, DevOps
And the Evolution of Delivery Practices
The Evolution of Delivery Practices

Traditional

- Multiple Views
- Quality Attribute-Driven Development
- Component-Based Development
- Asset Reuse
- Decision Capture
- Architecture Proving
Iterations allow for mid course corrections

As knowledge increases, the Product Owner uses iterations to guide the project towards enhanced goals.
The Evolution of Delivery Practices

<table>
<thead>
<tr>
<th>Traditional</th>
<th>Iterative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple Views</td>
<td>Iterative Development</td>
</tr>
<tr>
<td>Quality Attribute-Driven Development</td>
<td>Risk-Value Lifecycle</td>
</tr>
<tr>
<td>Component-Based Development</td>
<td>Shared Vision</td>
</tr>
<tr>
<td>Asset Reuse</td>
<td>Use Case-Driven Development</td>
</tr>
<tr>
<td>Decision Capture</td>
<td>Release Planning</td>
</tr>
<tr>
<td>Architecture Proving</td>
<td></td>
</tr>
</tbody>
</table>

InterConnect 2016
Agile Practices

- **Test-Driven Development (TDD)**
  - Creating tests that are a specification of what the code should do first
- **Continuous Integration**
  - Encourages frequent the integration and testing of programming changes
- **Refactoring**
  - Changing an existing body of code in order to improve its internal structure

- **Whole Team**
  - A focus on the value of highly-collaborative teams as exemplified by Scrum’s daily standup meeting. Instills of sense of collective ownership and responsibility
- **User Story-Driven Development**
  - Capture requirements in a lightweight manner. Encourages collaboration with the relevant stakeholders throughout a project
- **Team Change Management**
  - Supports the logging of defects or new requirements, by any member of the team, that are within the scope of the current iteration
# The Evolution of Delivery Practices

<table>
<thead>
<tr>
<th>Traditional</th>
<th>Iterative</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Multiple Views</td>
<td>• Iterative Development</td>
<td>• Test-Driven Development</td>
</tr>
<tr>
<td>• Quality Attribute-Driven</td>
<td>• Risk-Value Lifecycle</td>
<td>• Continuous Integration</td>
</tr>
<tr>
<td>Development</td>
<td>• Shared Vision</td>
<td>• Refactoring</td>
</tr>
<tr>
<td>• Component-Based Development</td>
<td>• Use Case-Driven Development</td>
<td>• Whole Team</td>
</tr>
<tr>
<td>• Asset Reuse</td>
<td>• Release Planning</td>
<td>• User Story-Driven Development</td>
</tr>
<tr>
<td>• Decision Capture</td>
<td></td>
<td>• Team Change Management</td>
</tr>
<tr>
<td>• Architecture Proving</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

InterConnect 2016
Disciplined Agile Delivery / Scaled Agile

Team size
- Under 10 developers
- 1000’s of developers

Compliance requirement
- Low risk
- Critical, audited

Geographical distribution
- Co-located
- Global

Domain Complexity
- Straight-forward
- Intricate, emerging

Enterprise discipline
- Project focus
- Enterprise focus

Organization distribution (outsourcing, partnerships)
- Collaborative
- Contractual

Organizational complexity
- Flexible
- Rigid

Technical complexity
- Homogenous
- Heterogeneous, legacy

InterConnect 2016
<table>
<thead>
<tr>
<th>Traditional</th>
<th>Iterative</th>
<th>Agile</th>
<th>Scaled Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple Views</td>
<td>Iterative Development</td>
<td>Test-Driven Development</td>
<td>Measured Performance</td>
</tr>
<tr>
<td>Quality Attribute-Driven Development</td>
<td>Risk-Value Lifecycle</td>
<td>Continuous Integration</td>
<td>Formal Change Management</td>
</tr>
<tr>
<td>Component-Based Development</td>
<td>Shared Vision</td>
<td>Refactoring</td>
<td>Concurrent Testing</td>
</tr>
<tr>
<td>Asset Reuse</td>
<td>Use Case-Driven Development</td>
<td>Whole Team</td>
<td></td>
</tr>
<tr>
<td>Decision Capture</td>
<td>Release Planning</td>
<td>User Story-Driven Development</td>
<td></td>
</tr>
<tr>
<td>Architecture Proving</td>
<td></td>
<td>Team Change Management</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
WORKED FINE IN DEV
OPS PROBLEM NOW
A Typical Deployment Landscape

Development
- Developer
  - Build
  - Unit Test

Build
- Build Engineer
  - Integration
  - Build
  - Deploy
  - Component Test

QA
- QA Team
  - Deploy
  - QA Test

SIT
- Integration Tester
  - Deploy
  - SIT

UAT
- User
  - Deploy
  - UAT

Production
- Operations Engineer
  - Deploy
  - Monitor

Release Manager
Apply Lean Principles

1. Get ideas into production fast
2. Get people to use it
3. Get feedback

Non-Value-added waste
Value-added production work

16
http://ibm.co/devopsfordummies
InterConnect 2016
Bottlenecks

8 / sec → 4 / sec → 4 / sec
# The Evolution of Delivery Practices

<table>
<thead>
<tr>
<th>Traditional</th>
<th>Iterative</th>
<th>Agile</th>
<th>Scaled Agile</th>
<th>DevOps</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Multiple Views</td>
<td>• Iterative Development</td>
<td>• Test-Driven Development</td>
<td>• Measured Performance</td>
<td>• Collaborative Development</td>
</tr>
<tr>
<td>• Quality Attribute-Driven Development</td>
<td>• Risk-Value Lifecycle</td>
<td>• Continuous Integration</td>
<td>• Formal Change Management</td>
<td>• Continuous Testing</td>
</tr>
<tr>
<td>• Component-Based Development</td>
<td>• Shared Vision</td>
<td>• Refactoring</td>
<td>• Concurrent Change Management</td>
<td>• Continuous Release</td>
</tr>
<tr>
<td>• Asset Reuse</td>
<td>• Use Case-Driven Development</td>
<td>• Whole Team</td>
<td>• Concurrent Testing</td>
<td>• Continuous Monitoring and Optimisation</td>
</tr>
<tr>
<td>• Decision Capture</td>
<td>• Release Planning</td>
<td>• User Story-Driven Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Architecture Proving</td>
<td></td>
<td>• Team Change Management</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Diagrams

- Traditional Practices
- Iterative Practices
- Agile Practices
- Scaled Agile Practices
- DevOps Practices

**InterConnect 2016**
Transformation Best Practices
Transformation Best Practices

IBM
- Consider all elements of a delivery ecosystem
- Implement a center of excellence
- Plan improvements around capabilities
- Adopt capabilities incrementally
- Embrace principles of organizational change

Kotter
- Establish a sense of urgency
- Create the guiding coalition
- Develop a vision and strategy
- Communicate the change vision
- Empower employees for broad-based action
- Generate short-term wins
- Consolidate gains and produce more change
- Anchor new approaches in the culture
Consider all Elements of a Delivery Ecosystem

Cross-cutting Concerns
Functionality, qualities, constraints

Method
Roles, work products, tasks, processes, standards, guidelines

Tools
Development tools & their integrations

Infrastructure
Locations, nodes & connectivity

Enablement
Training curriculum & courses

Organization
Organizational roles & units

Adoption
Adoption plan, organizational change, metrics
Implement a Center of Excellence

Center of Excellence Creates & Maintains Delivery Environment

Supports

Delivery Project Creates & Maintains Application
Plan Improvements around Capabilities

- **Capability**
  - Requirements Definition & Mgt.
  - Method
  - Tools
  - Organization
  - Infrastructure
  - Adoption

- **Practice**
  - Shared vision
  - Use case-driven development
  - Requirements management
  - Evolutionary architecture
  - Evolutionary design
  - Component software architecture
  - Continuous integration
  - Test-driven development

- **Development Environment Component**
  - Iterative development
  - Two-level project planning
  - Whole team approach

- **Architecture & Design**

- **Construction**

---

Large-grained

Fine-grained
Adopt Capabilities Incrementally

Current State
- Solution Context
  - Solution Definition
    - Solution Deployment
      - Solution Management

Increment 1
- Solution Context
  - Solution Definition
    - Solution Deployment
      - Solution Management

Increment 2
- Solution Context
  - Solution Definition
    - Solution Deployment
      - Solution Management

Increment 3
- Solution Context
  - Solution Definition
    - Solution Deployment
      - Solution Management

InterConnect 2016
Different Roadmaps to the Same Destination

Wave 1
Wave 2
Wave 3
Wave 4
Wave 5
Content of Increments

• What business drivers will each increment contribute to?
• What is the potential to deliver short-term wins?
  – Some capabilities will take longer to deploy than others
• What is a sensible sequence?
  – Are there dependencies between capabilities?
• Will projects be available to pilot the new capability?
• Do capabilities need to be subdivided?
  – E.g. requirements definition and requirements management
  – E.g. into practices such as Shared Vision, Whole Team
Case Study: Danske Bank
A large-scale distributed Agile improvement effort

Danske Bank IT Group
2000+ developers
6 business units
Global SW Dev Teams (20% India and increasing)

DKK 59 billion income
DKK 29 billion cost
4.5 m personal and
350,000 business
customers
22,000 employees

"We intend to enhance the efficiency of our IT
development process by 10% and reduce the time to
market from approximately 14 months to an average of
nine months. The first business deliveries will even be
provided in the course of just four months."

PETER RASMUSSEN, SENIOR VICE PRESIDENT
IT DEVELOPMENT PROCESSES & TOOLS, DANESKE BANK

How Agile at Danske Bank helps
deliver their improvement strategy

Scope
Projects (50%)
System management areas (90%)

Flexible
development model
& organisation
Adding another lifecycle,
providing approach for
system management

Efficiency
Productivity increase 10%

Time-to-market
Produce potentially shippable
products after each increment

Quality
Frequent user and
acceptance test

Customer and
employee
satisfaction
Ability to change scope and
plans, motivated development team
## Lifecycle selection framework (summary)

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Brief Description</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Management</td>
<td>Management Influences</td>
<td>2.0</td>
</tr>
<tr>
<td>Stakeholder</td>
<td>Stakeholder Influences</td>
<td>4.1</td>
</tr>
<tr>
<td>Project Team</td>
<td>Project Team Influences</td>
<td>3.3</td>
</tr>
<tr>
<td>Technology</td>
<td>Technology Influences</td>
<td>3.0</td>
</tr>
<tr>
<td>Solution</td>
<td>Solution Influences</td>
<td>3.0</td>
</tr>
</tbody>
</table>

### Management Influences
- Business Flexibility
- Empowered Teams

### Stakeholder Influences
- Acceptance of Agile
- Number of Stakeholders
- Stakeholder Responsiveness

### Project Team Influences
- Team Skills
- Embracing Change
- Co-located Teams
- Team Stability
- Team Roles
- Agile Disciplines

### Technology Influences
- Development Environment
- Execution Environment

### Solution Influences
- Requirements Churn
- Solution Complexity
- Time-To-Market
- Dependencies
- Release Frequency
- Demonstrability

---

InterConnect 2016
### Initial metrics

<table>
<thead>
<tr>
<th></th>
<th><strong>Business-related</strong></th>
<th><strong>Agile-related</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cycle time reduction</strong></td>
<td>§ Time spent from project initiation to delivery of first increment</td>
<td>§ Sprint velocity</td>
</tr>
<tr>
<td></td>
<td>§ Time spent from project initiation to project closure</td>
<td>§ Blocking work items</td>
</tr>
<tr>
<td><strong>Quality</strong></td>
<td>§ Defects (severity 1 and 2) in production per 100 FPs</td>
<td>§ Defect trend</td>
</tr>
<tr>
<td><strong>Continuous optimisation</strong></td>
<td>§ Process maturity level</td>
<td>§ Adoption of agile practices</td>
</tr>
<tr>
<td><strong>Productivity</strong></td>
<td>§ Function points per man year</td>
<td>§ Sprint burndown chart</td>
</tr>
<tr>
<td></td>
<td></td>
<td>§ Release burndown chart</td>
</tr>
</tbody>
</table>
“Gates” and “trains” ensure successful adoption
IBM CIO Office (internal apps): Strategy Steps

- 8,000 staff (dev, test, PMs, BAs, operations)

**Share Experience & Improve**
- Process Changes and Guidance
- Service Offerings/Questionnaire
- Develop and Deliver Education
- Execute Projects

- Agile, Iterative, Agile ERP, Waterfall
- Project and Team characteristics determine approach
- Agile Guidance for Services
- Negotiation process / SOW
- Process Simplification
- 4 Day Coaching Bootcamp (Coaches and Champions)
- 2 Day Disciplined Agile Workshop
- Select Service Offering
- Assign Coaches
- Monitor Progress
- Gather Lessons Learned and Measurements
- Streamline Deployment Process
- Developer Techniques - Test Driven Development
IBM CIO Office: Agile Business Value Metrics

In one year there were 81 Agile projects yielding $41.5 million in savings

<table>
<thead>
<tr>
<th>Business Value</th>
<th>Supporting Measurements</th>
<th>Definition</th>
<th>Baseline (Waterfall)</th>
<th>Target (Agile)</th>
<th>YTD Results (Strategic Agile)</th>
<th>YTD Results (All Agile)</th>
</tr>
</thead>
<tbody>
<tr>
<td>✅ Speed to Client Value</td>
<td>1. Cycle Time Reduction</td>
<td>Days to deliver 100 function points (Normalized to a Business Component) - Story points are being gathered as well</td>
<td>88 Days</td>
<td>70 Days</td>
<td>36 Days (59% Reduction) 26 Days – Weighted by FP</td>
<td>$33M</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Deployment Time Reduction</td>
<td>Days to deploy to production once gold code is declared. (Enhancements Only)</td>
<td>23 Days</td>
<td>18 Days</td>
<td>12 Days (48% Reduction)</td>
<td>$2M</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>✅ Improved Visibility</td>
<td>3. Defect Reduction</td>
<td>Number of defects found during Qualify Phase. Projects using Test Driven Development accelerator (e.g. more bugs found earlier in FIT, FVT, customer acceptance &amp; less delivered in function).</td>
<td>15.5 Defects per 100 FPs</td>
<td>12.4 Defects per 100 FPs</td>
<td>8.1 Defects (48% Reduction)</td>
<td>$835K</td>
</tr>
<tr>
<td>✅ Increased Efficiency (Capital reallocation)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>✅ Increased Flexibility</td>
<td>4. Cost avoidance of Change Activity</td>
<td>Earlier Identification of Change activity for Iterative. Change requests eliminated for Agile</td>
<td>N/A</td>
<td></td>
<td></td>
<td>$5.7M</td>
</tr>
<tr>
<td>✅ Decreased Risk</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$41.5 m</td>
</tr>
</tbody>
</table>

Results for 30 Agile, 51 Iterative projects (Strategic roll out)
Getting Started
1. Business Outcomes

2. Idea/Feature/Bug Fix/Enhancement

3. Line of Business
   - Customers
   - PMO
   - Requirements/Analyst
   - Feedback

4. Development
   - Build
   - QA
   - SIT
   - UAT
   - Prod

   - Developer
   - Build Engineer
   - QA Team
   - Integration Tester
   - User/Tester
   - Operations

5. Artifacts Repository
   - Code Repository
   - Artifact Repository
   - CMDB

6. Metrics - Reporting/Dashboarding
An Example Whiteboard

**Business**
- Increase Revenue
- Increase Market Share

**IT**
- Client Facing (App + UI)
- SaaS Office = HR
- Ops (sys) - ops scheduling
- IT = crow store
- Dev tools = code maintenance
- X cheaper

**K Initiatives**
- Auto-test after every deploy
- Automate build/deploy

**UX Design**
- Design
- Branding

**Provider**
- AWS
- Cloud

**Team**
- Dev
- Test
Summary

• Agile / DevOps transformation considers
  – Method
  – Tools
  – Enablement
  – Organization
  – Infrastructure
  – Adoption
• Well-understood principles of organizational change apply
• Apply best practices to ensure success!
Notices and Disclaimers

Copyright © 2016 by International Business Machines Corporation (IBM). No part of this document may be reproduced or transmitted in any form without written permission from IBM.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IN NO EVENT SHALL IBM BE LIABLE FOR ANY DAMAGE ARISING FROM THE USE OF THIS INFORMATION, INCLUDING BUT NOT LIMITED TO, LOSS OF DATA, BUSINESS INTERRUPTION, LOSS OF PROFIT OR LOSS OF OPPORTUNITY. IBM products and services are warranted according to the terms and conditions of the agreements under which they are provided.

Any statements regarding IBM’s future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer’ s responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer’ s business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer is in compliance with any law.
Notices and Disclaimers Con’t.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM’s products. IBM EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.

IBM, the IBM logo, ibm.com, Aspera®, Bluemix, Blueworks Live, CICS, Clearcase, Cognos®, DOORS®, Emptoris®, Enterprise Document Management System™, FASP®, FileNet®, Global Business Services®, Global Technology Services®, IBM ExperienceOne™, IBM SmartCloud®, IBM Social Business®, Information on Demand, ILOG, Maximo®, MQIntegrator®, MQSeries®, Netcool®, OMEGAMON, OpenPower, PureAnalytics™, PureApplication®, pureCluster™, PureCoverage®, PureData®, PureExperience®, PureFlex®, pureQuery®, pureScale®, PureSystems®, QRadar®, Rational®, Rhapsody®, Smarter Commerce®, SoDA, SPSS, Sterling Commerce®, StoredIQ, Tealeaf®, Tivoli®, Trusteer®, Unica®, urban{code}®, Watson, WebSphere®, Worklight®, X-Force® and System z® Z/OS, are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at: www.ibm.com/legal/copytrade.shtml.
Thank You

Your Feedback is Important!

Access the InterConnect 2016 Conference Attendee Portal to complete your session surveys from your smartphone, laptop or conference kiosk.