The ABCDs of Database Development: Always Be Continuously Delivering

Elizabeth Ayer (@ElizAyer)
Redgate Software
Goals

- Understand the technology and **process requirements** to work towards automation step-by-step in your release pipeline.
- Learn about the **organizational changes** necessary to support process modifications.
- Appreciate why these changes are necessary in support of **modern development and deployment methodologies**.
Elizabeth Ayer
Product Manager, Redgate Software

@ElizAyer

in/elizabethhayer
ALM – and where the database fits in

Three core processes in Application Lifecycle Management:

- Governance
  - Requirements
- Development
  - Design, build and test
- Operations
  - Deployment and monitoring
Natural friction across pipeline

Development → Operations
Natural friction across pipeline

Development → Operations

Operations → Development

Why?
Development focus is on speed
- Agile
- Scrum
- Lean
- Feature-driven Development
- Iterative
Operations focus is on production protection
Monitoring
 Deployment
 Integrity
 Data Management
Databases as a bottleneck

3 reasons why databases have traditionally slowed down deployments:

1. **Odd languages**
   - SQL
   - Cubes
   - X-Query

2. **Data persistence**
   - Data outlives applications
   - Data can’t be replaced

3. **DBA paranoia**
   - Frankly…
A quick primer on continuous delivery

Release pipeline

Development  Test  Production

Continuous integration  Continuous delivery  Continuous deployment

Approval gate
The goals of continuous delivery for databases

• Faster feedback on changes made
  • Continuously integrate team changes
  • Automated testing
  • Releases rehearsed in testing environments before deployed to Production

= repeatability, repeatability and a strong team process for changes
Focus on the pipeline

Think of the Toyota production system...

- Start with Lean
  - Focus on the customer, eliminate waste
  - Continuously Improve
  - Empower the team
  - Optimize the whole
  - Plan for change
  - Automate processes
  - Build quality in
Four key stages of the deployment pipeline

- Source Control
- Continuous Integration
- Automated Testing
- Automated Deployment
Create development environment for automation

**Empower Development**
- Environment should let them work at their speed

**Take part in Development**
- Instead of stopping bad deployments, stop bad development

**Automate your build process**
- Supply clean production data or supply good sample data

**Get started on writing tests**
- Build your library
Automate **testing**

Different types of testing for different stages of the pipeline:

- **Development**
  - Unit tests

- **Integration Testing**
  - Integration tests
  - Automated tests

- **QA**
  - Deployment validation

- **Pre-Production/Staging**
  - Behaviour validation

- **Production**
  - Other validations
Always Be Continuously Delivering
Every delivery is practice
Changes to philosophy

DBAs must work with Devs…

Yes, you must protect the data for the business, but that must be tempered with helping deliver functionality.
Changes to philosophy

Devs must work with DBAs…

**Yes**, you may know more about business needs, so educate rather than isolate.
Changes to philosophy

Project Management must think of operations as part of development...

**Yes:**
- Deployment is part of development
- Release 1.1 and on are part of development
  - And planning for 1.1 is not premature optimization
- Data retention is part of development
Changes to workplace

Technology is only half the story...
Changes to workplace for continuous database delivery

• Empower developers to do more
• Bring DBAs into development teams
• Management must buy-in
• Management must get out of the way
How are other companies getting started?
Case Study: **Boxon**

- Global packaging and labelling company
- Based in Sweden; sub-division in China
- Three business areas: profitable packing solutions; intelligent marking; customized big-bags solutions for bulk handling
- One developer responsible for ASP.NET application with SQL Server backend, enabling customers to control consistent printing of labels wherever the print shop is located in the world
Boxon – Initial Process

• Deploying to production was scary...
• Database changes not version-controlled
• 2-hour window - at night - to complete deployments to production – difficult to find quiet period with a global customer base
Boxon – Improved Process

- Deployments are less scary now...
- Tickets are logged and prioritized in Unfuddle (unfuddle.com)
- Ticket numbers are logged in SVN and TeamCity to track items
- Now use SQL Source Control and Subversion to version DB changes
  - Source of authority on database build
- TeamCity (CI build server) is triggered on check-in of changes
  - Fast response – changes checked in frequently
- Notification if build fails
- Build packaged into a Nuget file for deployment
Case Study: Move with Us

- Risk management solution and sales and marketing channel provider for residential property businesses in the UK
- Located near Cambridge, UK
- Customers put a heavy load on databases, so updates need to be efficient
Move with Us – Initial Process

- Database code often not checked in to VCS due to risk of conflicts – changes were communicated by email instead
- Hot fixes sometimes done directly on integration server
- 1 day needed to release updates – a lot of work in building and testing
- Problems with merging code led to a lot of firefighting
Move with Us – Improved Process

- Each developer has a local copy of database
- Source Control on Subversion used to check in database changes to version control
- TeamCity (CI build server) automatically compiles changes on check-in
- Successful builds packaged into a Nuget file automatically
  - Fast getting here – multiple database changes checked in per day
- Nuget package deployed to Test on demand
- If problems detected, return to Dev to fix
Case Study: Practice Fusion

- Web-based electronic health record (EHR) company
- Founded in 2005
- Based in San Francisco
- Hosts over 50 million patient records
- Used in all 50 states and by 150k+ physicians
- 3 DBA engineers; 1 data architect; 10+ developers
Practice Fusion – Initial Process

• Hand-crafted SQL scripts - inconsistent
• Database changes not always version controlled
• Changes released by opening dozens of files in SSMS
• Long-winded team comms if further changes needed
• Smoke-testing deployments showed some objects had been deployed straight to Prod and weren’t in deployment script
Practice Fusion – Improved Process

- Database source control on Subversion to version DB changes
  - Source of authority on database build
- Jenkins (CI build server) is triggered on check-in of changes
  - More time to develop; less time managing scripts
- Build failed by Jenkins if problems detected
  - Fewer issues deploying to production
- Jenkins, SQL Compare and SQL Data Compare used to deploy changes to environments
- Custom scripts for replication
Summary

• Eliminate or mitigate friction
• Adopt lean methodologies and focus
• A-B-C-D
• Change your philosophy
• Change your workplace
Goals

- Understand the technology and **process requirements** to work towards automation step-by-step in your release pipeline.
- Learn about the **organizational changes** necessary to support process modifications.
- Appreciate why these changes are necessary in support of **modern development and deployment** methodologies.
Documentation and resources

- Continuous Delivery by Jez Humble and David Farley (Addison Wesley)
- The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win by Gene Kim, Kevin Behr and George Spafford (IT Revolution Press)
- The Goal: A Process of Ongoing Improvement by Eliyahu M. Goldratt and Jeff Cox (Gower Publishing Ltd.)
- Agile Organization by the agile admin (theagileadmin.com)

Further resources:
- Database Delivery Learning program: [www.red-gate.com/delivery](http://www.red-gate.com/delivery)
  - [Patterns and practices](http://www.red-gate.com/delivery) on Simple-Talk
  - Tutorials in [Red Gate training academy](http://www.red-gate.com/delivery)
- [www.youtube.com/user/RedGateVideos](http://www.youtube.com/user/RedGateVideos) - for recorded seminars