Achieve Competitive Advantage with IBM DevOps

Bridge the Gap between Development and Operations

Smarter software for a smarter planet

Enabling Product and Service Innovation | Rational
Introduction

Break Down the Barriers with Improved Planning and Collaboration

Define, Manage, and Track your Requirements and Designs

Integrated Change, Build and Deploy is What Sets Us Apart

Quality is not Accidental

**Bridge the Gap between Development and Operations**

Put your Business in Motion - Become a Mobile Enterprise
Software delivery and IT operations always had (and still have) many challenges.

CHALLENGES

**Costly, error prone** manual processes delaying innovation while falling behind the competition

**Slow** deployment to development and test environments leave teams waiting and unproductive

**Upgrade risk** due to managing multiple application configurations and versions across servers

Software glitch costs trading firm Knight Capital $440 million in 45 minutes

New Zealand’s biggest phone company, Telecom paid out $2.7 million to some 47,000 customers who were overcharged after a software glitch

Software failures were behind 24% of all medical device recalls last year
Emergence and proliferation of Systems of Engagement made many of the known problems more severe

- **Line-of-business**
  Takes too long to introduce or make changes to services/products

- **Operations**
  Rapid app releases impacts system stability and compliance

>70% of resources devoted to maintaining existing systems and products

4-6 weeks to deliver application changes to customers

- **Development/Test**
  Speed mismatch between faster moving front office and slower moving back office systems, delaying time to get feedback

- **Suppliers**
  Delivery in the context of agile

>80% of applications rolled back

>50% of outsourced projects fail to meet objectives
IBM calls the end-to-end solution of these challenges “DevOps”

• DevOps is an overloaded term due to recent popularity

• IBM defines DevOps as:
  "Enterprise capability for continuous software delivery that enables clients to seize market opportunities and reduce time to customer feedback"

• Key capabilities to enable DevOps Approach
  – Plan and Measure
  – Develop and Test
  – Release and Deploy
  – Monitor and Optimize
Continuous software delivery with IBM DevOps solution

Connecting:
- Customers
- Business Owners
- Development & Test
- Operations & Production

Providing Capabilities to:
- Plan & Measure
- Develop & Test
- Release & Deploy
- Monitor & Optimize

Resulting in:
- Accelerated Software Delivery
  - Expanding collaboration to include customers, LOB and others to eliminate organization silos

- Balanced speed, cost, quality and risk
  - Automating manual processes across delivery lifecycle to eliminate waste/delays and compliance tracking

- Reduced time to customer feedback
  - Enabling a customer feedback loop for continuous improvement

© 2014 IBM Corporation
IBM is uniquely equipped to offer end-to-end DevOps capability.

**Plan and Measure**
- Continuous Business Planning
  - IBM Rational® Focal Point™
  - IBM DOORS NG

**Develop and Test**
- Collaborative Development
  - IBM DOORS NG
  - IBM Rational Team Concert™
  - IBM Rational Quality Manager
  - IBM Rational Software Architect
  - Design Manager
  - IBM Rational Lifecycle Integration
  - Adapters
  - IBM Rational Developer family

**Monitor and Optimize**
- Continuous Monitoring
  - IBM SmartCloud® Application Performance Management
  - IBM SmartCloud Monitoring - Application Insight
  - IBM SmartCloud Analytics - Log Analysis

**Release and Deploy**
- Continuous Release and Deploy
  - IBM UrbanCode Release
  - IBM UrbanCode Deploy
  - IBM SmartCloud® Orchestrator

**Continuous Customer Feedback and Optimization**
- IBM Digital Analytics
- IBM Tealeaf® CX
- IBM SmartCloud Control Desk

**Continuous Testing**
- IBM Rational Quality Manager
- IBM Rational Test Workbench
- IBM Rational Test Virtualization Server
- IBM Rational Development and Test Environment for System z
- IBM InfoSphere Optim Test Data Management
- IBM Security AppScan

IBM is uniquely equipped to offer end-to-end DevOps capability.
IBM UrbanCode is the premier release and deploy automation solution

Enabling clients to more rapidly deliver mobile, cloud, big data and traditional applications with high quality and low risk

**IBM UrbanCode Deploy**
- Simplifies deployment automation for applications
- Manage configurations across environments
- Intelligent deployment artifact tracking
- Higher quality with repeatable, reliable, and governed processes
- Technology plugins supported out of the box

**IBM UrbanCode Release**
- Define, manage, and coordinate operational releases
- Aggregates applications into single release
- Phases to support quality and geo dispersed deployments
- Coordinate manual and automated tasks
- Replaces manual spreadsheets and documents
IBM UrbanCode is in the center of the continuous delivery solution

Products can be incrementally adopted when/if needed
IBM UrbanCode Deploy overview

*IBM UrbanCode Deploy automates a process of application and infrastructure deployments.*
IBM UrbanCode high level architecture

The service tier has a central server that provides a web server front-end and core services, such as workflow, agent management, deployment, inventory, security, and others.

- The data tier's relational database stores configuration and runtime data
- CodeStation is for non-structured data

An Agent runs on host and implements actual deployment.
IBM UrbanCode Deploy is a tool for automating application deployments through existing environments.
IBM UrbanCode Deploy automates deployment to IBM MobileFirst platform – Worklight

Automatically deploy Worklight applications as part of your deployment process
The plugin supports:
• Deploying Worklight adapter and application components to Worklight Server
• Deploying Worklight mobile application binaries to the Worklight Application Center for distribution and management
## Enterprise release common problems

<table>
<thead>
<tr>
<th>RELEASE</th>
<th>DEPLOY</th>
</tr>
</thead>
<tbody>
<tr>
<td>RELEASE PLANNING</td>
<td>DEPLOYMENT GOVERNANCE</td>
</tr>
<tr>
<td><strong>schedule and prepare releases for development</strong></td>
<td><strong>visibility to every aspect of a future release</strong></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### COMMON CHALLENGES

<table>
<thead>
<tr>
<th>Manual, labor-intensive process</th>
<th>Poor integration between systems</th>
<th>Assembling deployment plans requires many meetings and reviews</th>
<th>Labor intensive deployment effort</th>
</tr>
</thead>
<tbody>
<tr>
<td>No standard process / tooling across LOBs</td>
<td>Troubleshooting issues difficult because unclear which application versions are deployed to which environments.</td>
<td>Gaps and errors in plans cause production issues.</td>
<td>Long outage windows deployment errors &amp; risk</td>
</tr>
<tr>
<td>Limited visibility into dependencies leads to sub-optimal plans</td>
<td>Production deployment status only via conf. calls and manual reporting.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
IBM UrbanCode Release solves enterprise release common problems

The release pipeline contains phases

UrbanCode Release deals with applications

Deployment plan is the main part of the release. It can be scheduled or manually executed.
IBM UrbanCode application lifecycle integrations

*Providing richer linked data integrations across the entire application lifecycle*

Check out other lifecycle integrations at [http://plugins.urbancode.com](http://plugins.urbancode.com)
IBM UrbanCode Deploy provides continuous testing with virtualized services

Avoid testing bottlenecks due to dependencies on external services

- Automate setup and management of test virtualization server in the cloud
- Automates configuration of virtualized services for an application under test
- Automate setup of production-like test environments with low cost
Introducing IBM UrbanCode Deploy with Patterns

- Pattern designer
  - Design open, full stack application environments in a diagram or textual editor
- Design once, deploy anywhere
  - Deploy full stack environments to multiple clouds
- Environment lifecycle management
  - Manage infrastructure change and easily apply changes to existing environments
- Delivery process automation
  - Automated delivery process with integrated full stack environments
Overview of IBM UrbanCode Deploy with Patterns

Engine manages cloud infrastructure and provisioning resources from clouds

IBM UrbanCode Deploy with Patterns Engine:
- Cloud discovery service
- Heat engine
- Heat extensions

Provides interface for maintaining blueprints

IBM UrbanCode Deploy with Patterns Designer Server

Provides information about components

IBM UrbanCode Deploy Server

Rational License Server

OpenStack Keystone Server
- OpenStack Identity Server

Discovery service provides information about the available cloud resources

Provides authentication tokens to the OpenStack system
Improve delivery lifecycle with software defined environments

SCM: RTC, AccuRev, CVS, Dimensions, Git, Mercurial, Perforce, SVN, TFS

Build Automation: UrbanCode Build, BuildForge, Hudson/Jenkins, JetBrains TeamCity, Microsoft TFS/TeamBuild

IBM UrbanCode Deploy

Deploy Automation

Pull changes

Publish build

Execute application deployment and manage settings across environments

IBM Endpoint Manager

Chef

IBM PureApplication System

SoftLayer

IBM UrbanCode Deploy with Patterns

Cloud Provisioning

Environments

DEV

QA

PROD
Cloud Hosts automate infrastructure delivery

- A Cloud Host is a public or private cloud capable of deploying virtual machines into a virtualized server environment

- Automatic deployment of virtual machines utilizes images, patterns, and scripts to create a pre-configured and reproducible running system of one or more virtual machines

- The virtual machines are a fully configured stack with virtualized processors, memory, and storage, operating system, middleware, and application code

- Virtual Systems Patterns are particularly useful for DevOps
What is a Virtual System Pattern?

• The defined architecture of an application solution
• For each server component of the application
  – Virtual machine with defined resources
  – Installed operating system
  – Integrated middleware
  – Application code
  – Pre-configured and tuned
  – May include pre-configured monitoring, security, and test frameworks
• In a form that can be automatically deployed to a virtualized server, resulting in repeatable, consistent deployment
Virtual System Patterns define explicit topology

- This pattern defines two nodes, i.e. separate Virtual Machines
- Core OS 1.0 is an image used as the base for each deployable node in the pattern
- Scripts for each node provide further configuration when the nodes are deployed as Virtual Machines
IBM offers three cloud host solutions that support Virtual System Patterns

• IBM Workload Deployer
  – A hardware appliance that can deploy Virtual System Patterns to Intel, Power, and zLinux hosting platforms
  – A component of IBM SmartCloud Orchestrator

• SmartCloud Orchestrator (contains IWD)
  – Can deploy Virtual Systems Patterns to Intel, Power, and zLinux hosting platforms

• PureApplication System
  – Hardware, software, and a management appliance in a pre-built rack
  – PureApplication Manager can deploy Virtual System Patterns to it’s own Intel and Power servers
Resource Templates are models for a group of resources

- Hierarchy of resources, groups, and agent prototypes
- Starting point for creating new resources
- Virtual system patterns on a cloud system can be imported as resource templates.
- Resource templates can be created and provisioned to OpenStack or Amazon Elastic Compute Cloud through IBM UrbanCode Deploy with Patterns.
Environments can be modeled in reusable “Blueprints” with UrbanCode Deploy

- Blueprints specify how a template or pattern for an environment should be configured for a particular application
  - Including which components are deployed on which parts
- Provide instructions on how to configure the environment
  - Application component mapping to resource templates
  - Resources needed
  - Agents to deploy
Environments can be modeled in reusable “Blueprints” with UrbanCode Deploy with Patterns

Leverage the palette at the right side of the designer to work with the blueprint:

- The palette is adjusted depending on specific cloud system
- Add the components from IBM UrbanCode Deploy to the blueprint
- Drag virtual images to the blueprint
- The Network drawer shows networks that are available on the cloud.
- The Storage drawer shows cloud storage volumes.
- The Security drawer provides security settings which can be applied to the compute nodes.
- The Policies drawer includes resources like load balancers and auto-scaling groups.
- The Blueprint drawer shows other blueprints that can be added to the current blueprint.
Summary

IBM DevOps solution provides continuous delivery

- Increasing reliability, reducing risks and time to market, and decreasing costs

UrbanCode Release and Deploy with IBM Cloud Hosts make software delivery more efficient

- By automating software delivery in virtualized cloud environments in real time

IBM UrbanCode Deploy and Release integrate with Rational CLM

- Providing continuous delivery of software to agile projects
- Improving quality of delivered applications by integrating with functional, integration and performance testing tools
DEMO 3: UrbanCode Deploy integration with PureApplication System

- IBM UrbanCode Deploy allows establishing connection with the PureApplication private cloud.
- IBM UrbanCode Deploy allows importing a PureApplication VSP and save it as a resource template.

The goal of this demo is to demonstrate the UrbanCode Deploy ability to import environment definition from a PureApplication VSP and create the corresponding resource template.