IBM SolutionsConnect 2013
Turning Opportunity into Outcomes.

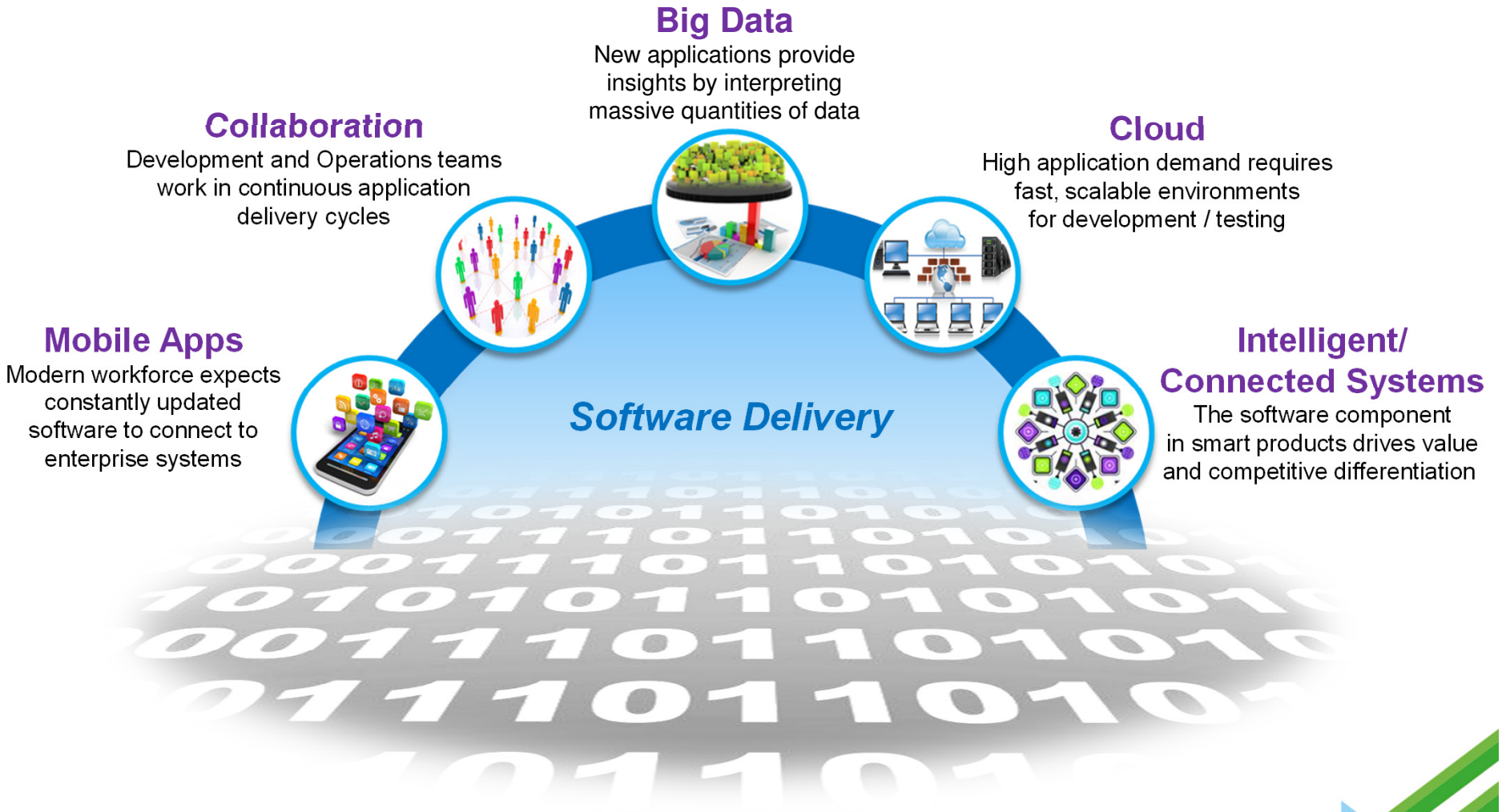
Increase your agility with continuous delivery of software

Alexandre Abi Khaled

Agenda

1. Promise of DevOps
2. Practices in Continuous Delivery
3. Plan to utilize the Cloud

Software delivery is at the heart of today's top technology trends



It’s about gaining competitive advantage through software innovation

Differentiated and engaging customer experiences

Build customer loyalty and increase market share by continuously obtaining and responding to customer feedback

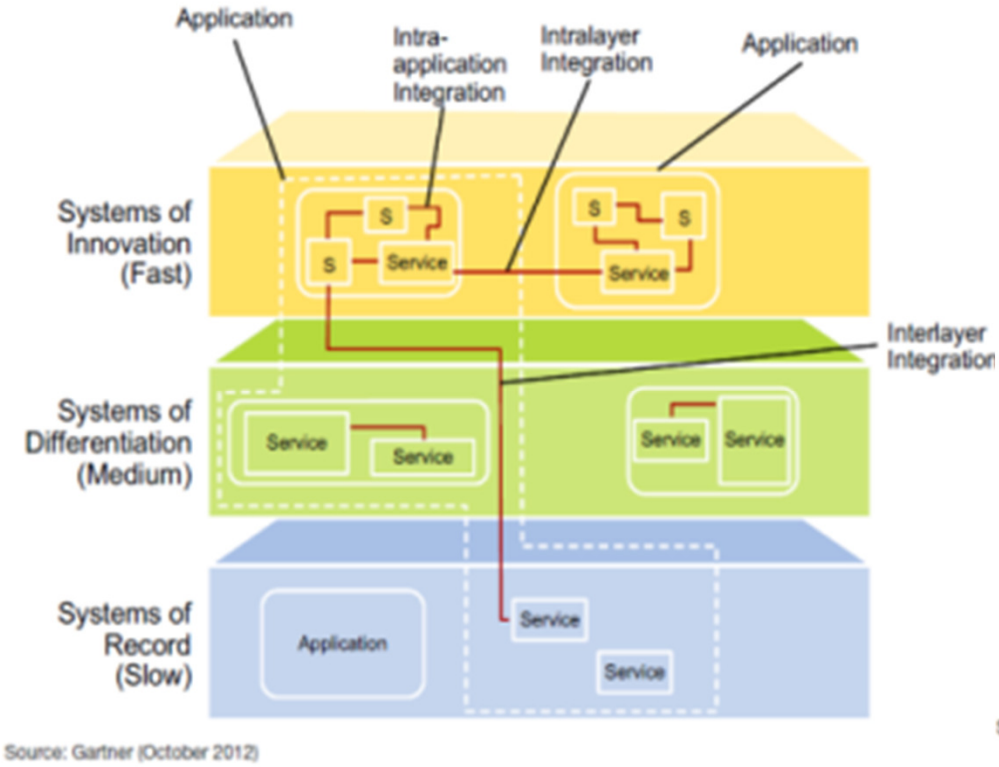
Quicker time to value

Obtain fast-mover advantage and capture markets with software-based innovation, with improved predictability and success

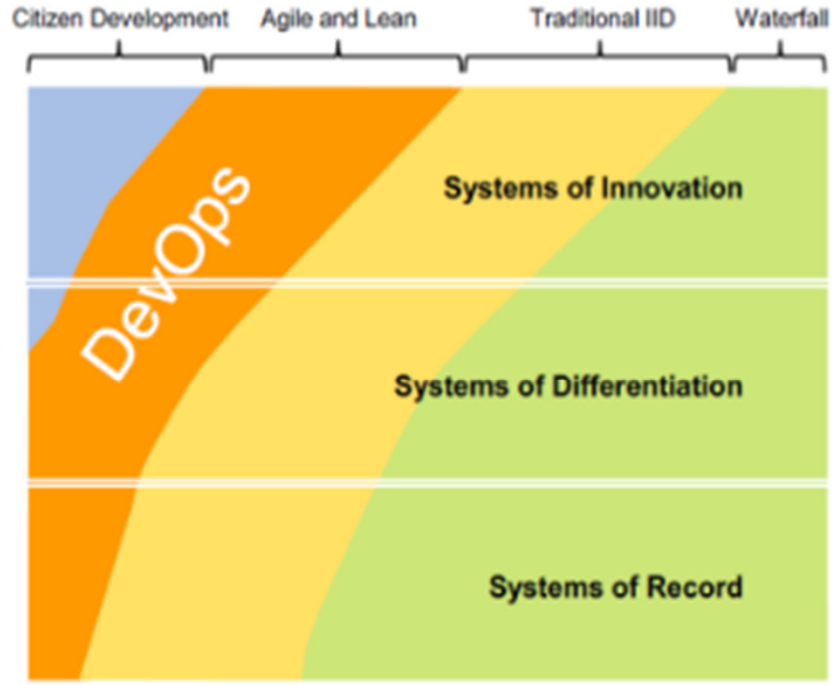
Increased capacity to innovate

Reduced waste and rework in software delivery and enables a shift of resources to higher value activities

Evolving customer and market expectations

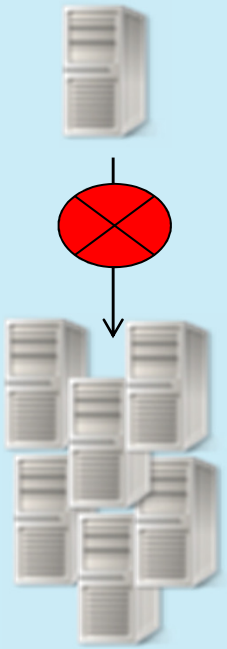
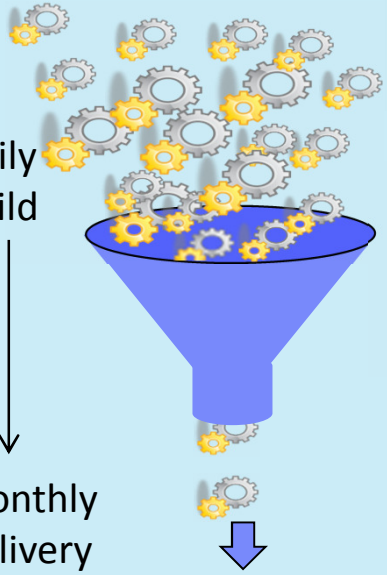
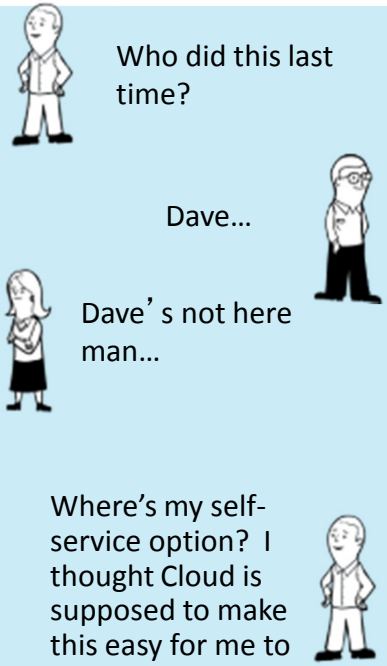
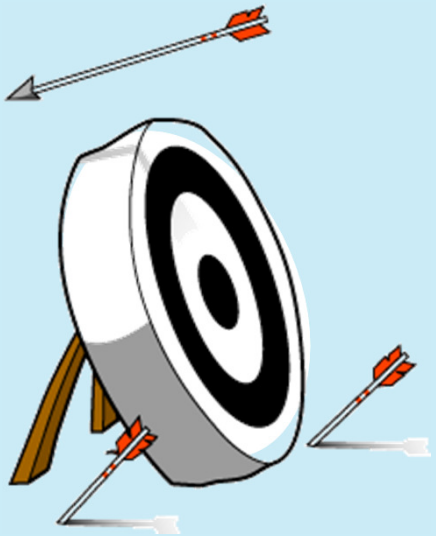


Source: Gartner (October 2012)

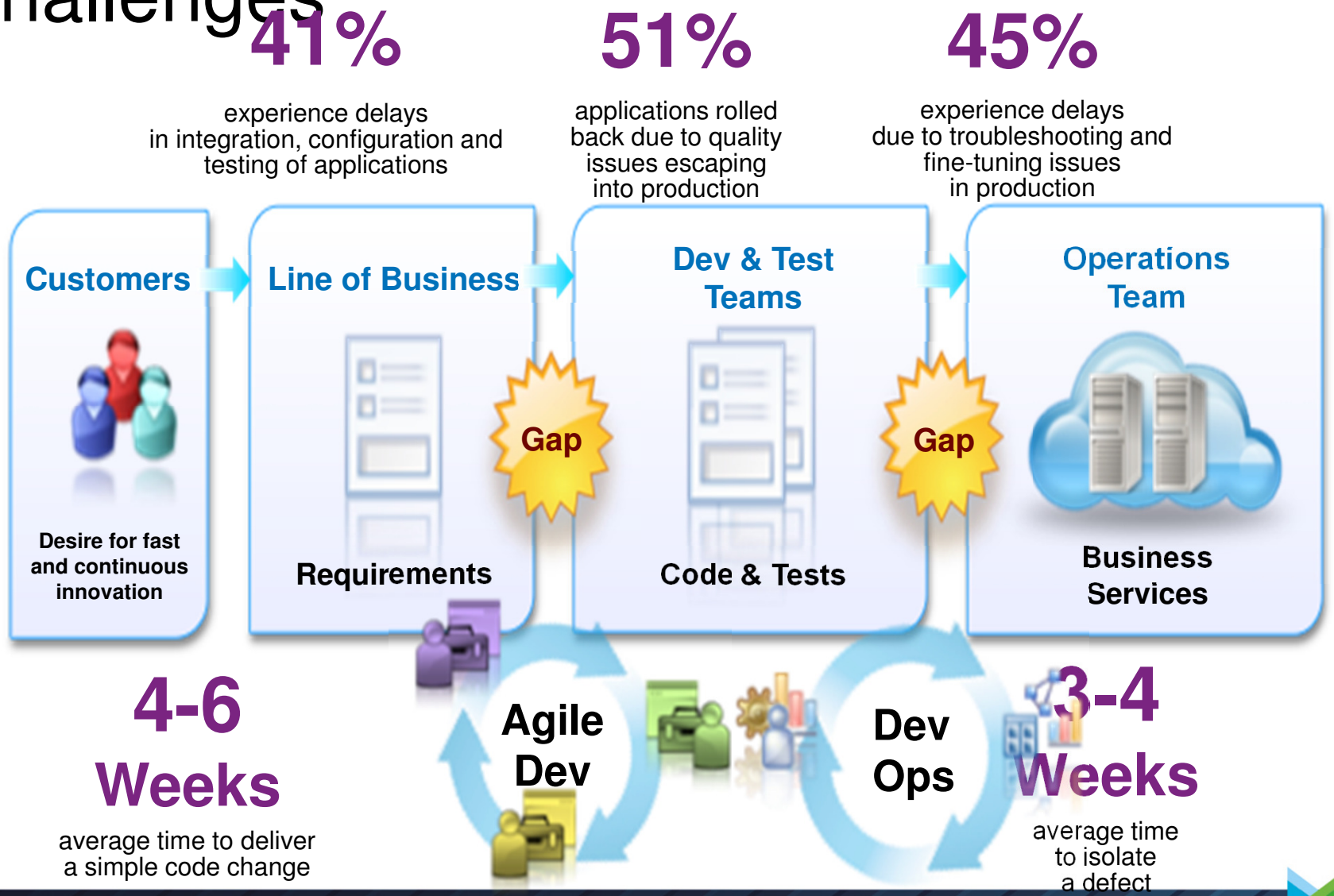


Source: Gartner (October 2012)

Patterns of challenges from client experiences

Differences in dev and ops environments cause failures	Backlog of agile releases that Ops cannot handle	Manual tribal processes for release lack repeatability/ speed	Lack of stakeholder feedback leads to missed opportunities
			

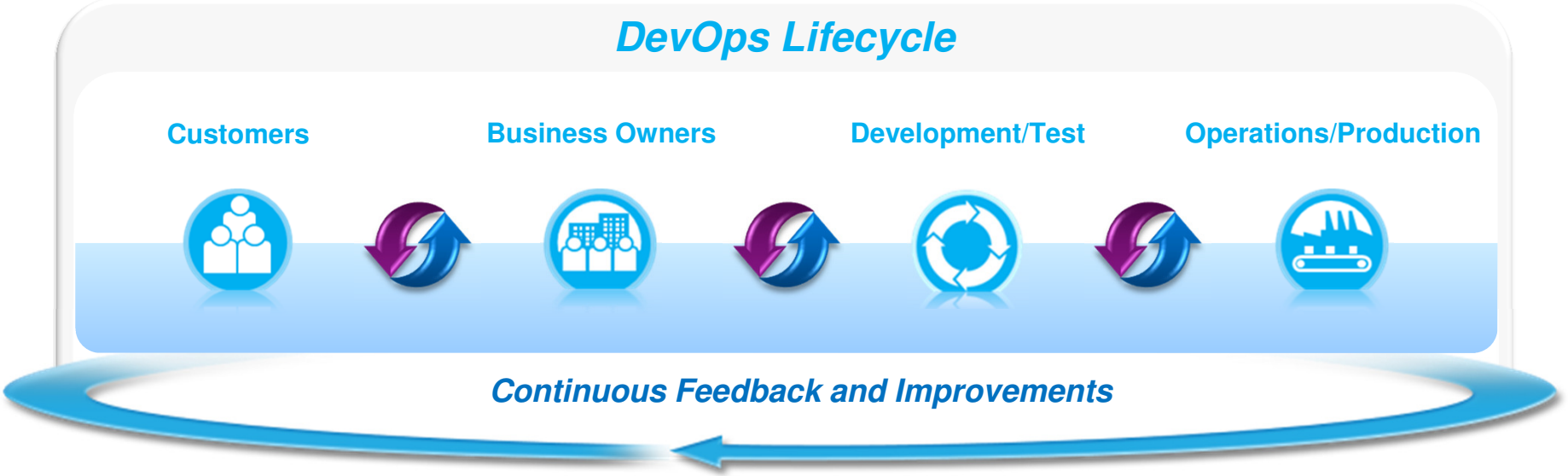
DevOps answers the following challenges



DevOps: A blueprint for continuous delivery

devops *noun* \dev-aps\

Enterprise capability for continuous software delivery that enables clients to seize market opportunities and reduce time to customer feedback.



- Accelerated software delivery
- Reduced time to gather and analyze customer
- Improved governance across the lifecycle
- Balanced quality, cost and speed

Agenda

1. Promise of DevOps
2. Practices in Continuous Delivery
3. Plan to utilize the Cloud

jazz Building better software

Search jazz.net | Register | Log in

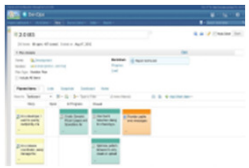
Products | Downloads | Community | Our Story | MY STUFF | GET HELP | EXTEND | LIBRARY | FORUM | BLOGS | JAZZHUB

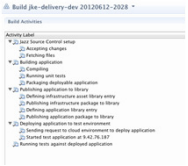
IBM SmartCloud Continuous Delivery

Extend agile development to enable enterprise DevOps

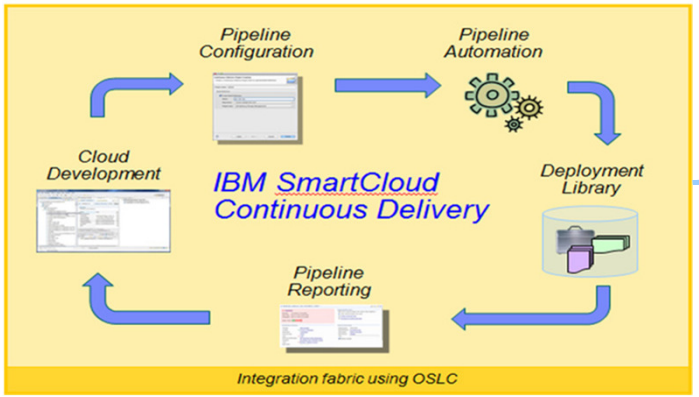
Download 2.0 Release
October 30, 2012

Standardize | **Plan & Track** | **Manage Changes** | **Automate Delivery** | **Feedback**





Rational Team Concert



IBMSmartCloud Provisioning

IBM Workload Deployer

IBM PureApplication Systems

Agile Development

Deployment of Virtual Systems

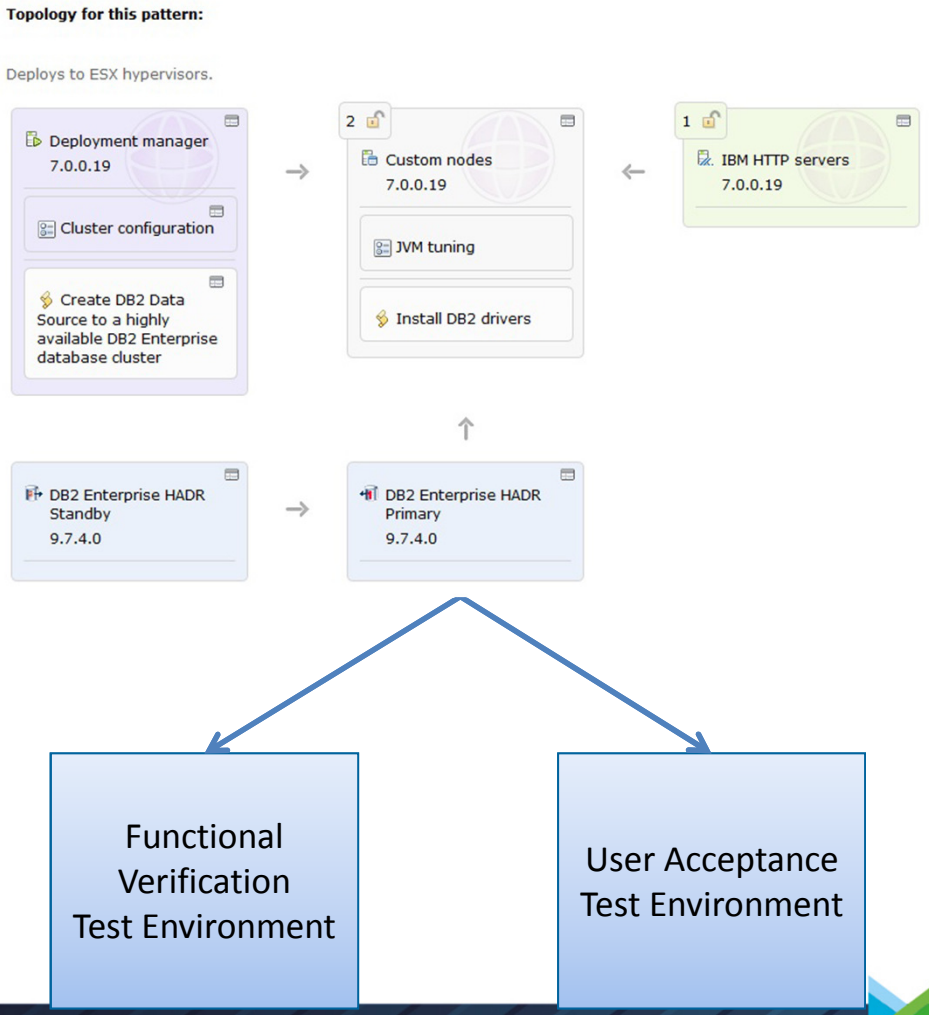
Standardize: *Communicate and share environment patterns*

Issues

- Dev environments different from production environments
- Difficult to provision test environments
- Environment provisioning creates new "snowflakes"

Solution

- Define standard Virtual System Patterns for environments
- Share patterns between Dev and Ops to improve communications
- Repeatable and reliable process to provision base environments (rubber stamps)



Plan and Track Changes:

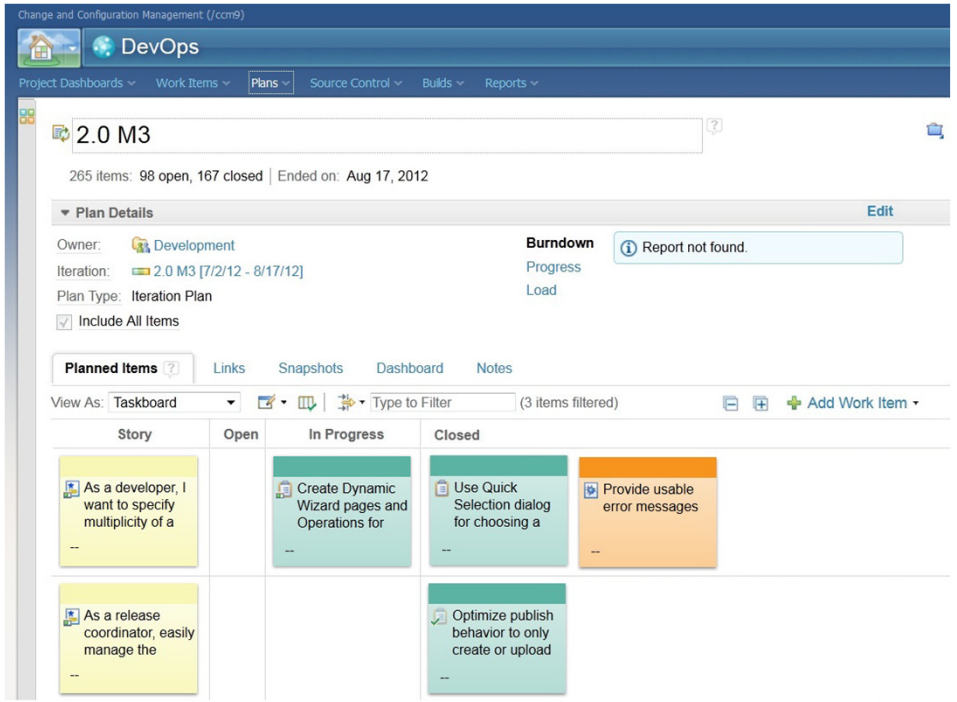
Common tools across Dev and Ops

Issues

- Configuration tasks are not planned or tracked
- Different tools/processes for tracking work

Solution

- Use collaborative development tools to track and plan application and configuration changes
- Ensures work is properly planned in the proper order
- Ensures audit trail for planned changes to environments



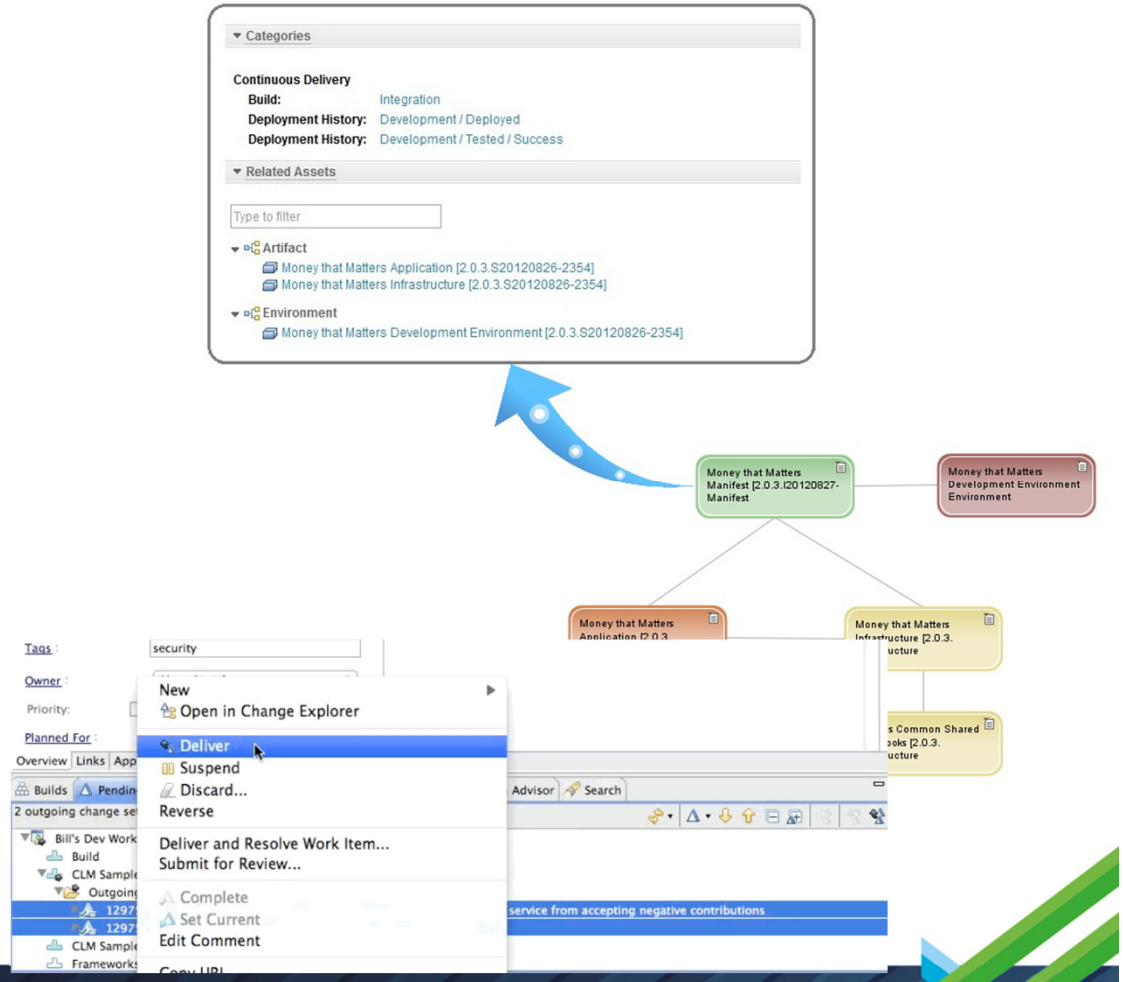
Manage Changes: *Application and configuration changes versioned*

Issues

- Configuration changes vary between environments
- Configuration changes are not well managed
- Release artifacts are randomly created and applied

Solution

- Version control configuration changes as code in an SCM close to the application changes
- Version and track release artifacts in an Artifact Library
- Track release dependencies



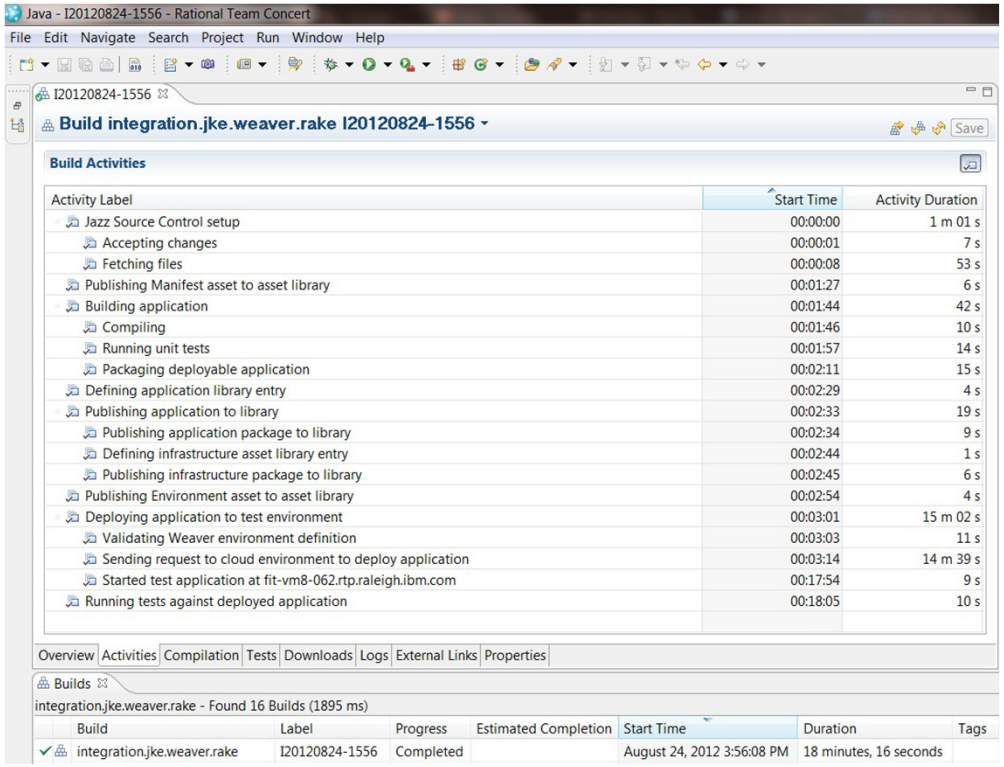
Automate Delivery: *Repeatable and reliable processes*

Issues

- Application deployment is generally an unreliable, manual process
- Difficult to replicate
- Slow to execute
- Long delays between deployments increases size of change sets making problem determination difficult

Solution

- Tooling to define and configure an automated delivery process
- Reliable and repeatable delivery process
- Deploy early and often to “fail fast” and resolve even faster



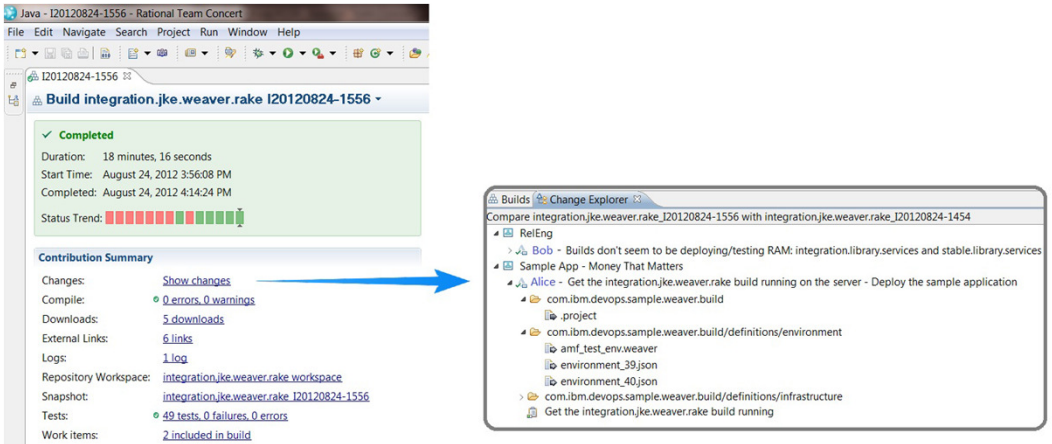
Feedback: *Immediate feedback of changes*

Issues

- Weeks, months pass before getting any feedback of a change (if any)
- Often cannot determine where/if a build was deployed
- Long delays in feedback makes problem determination more difficult

Solution

- Report results of automated delivery process directly in context of development tools
- Preserve traceability of delivery output with build process
- Report automated tests results in a timely fashion

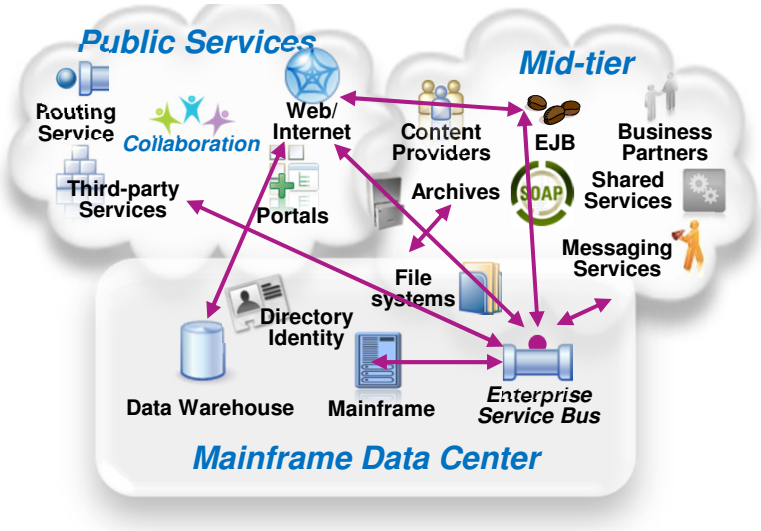


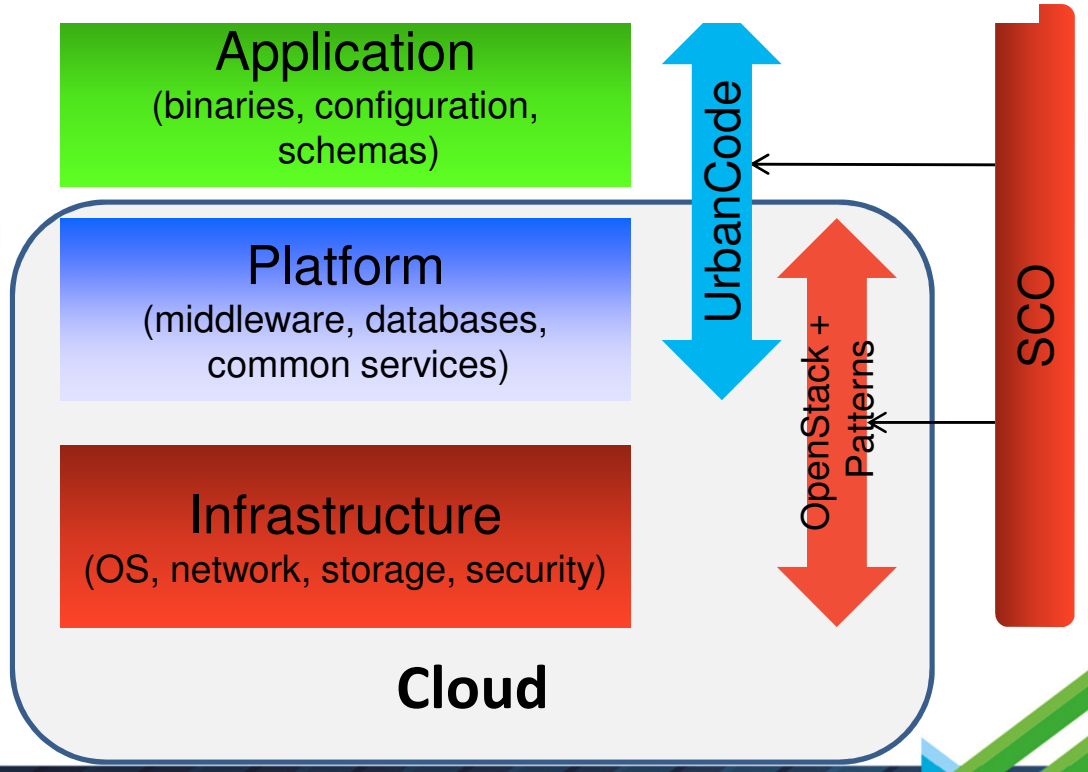
Agenda

1. Promise of DevOps
2. Practices in Continuous Delivery
3. Plan to utilize the Cloud

Cloud only gets you part of the way

- Cloud (software defined environment) drives speed and consistency of infrastructure & platform
- Applying application and configurations changes is the next complexity bottleneck

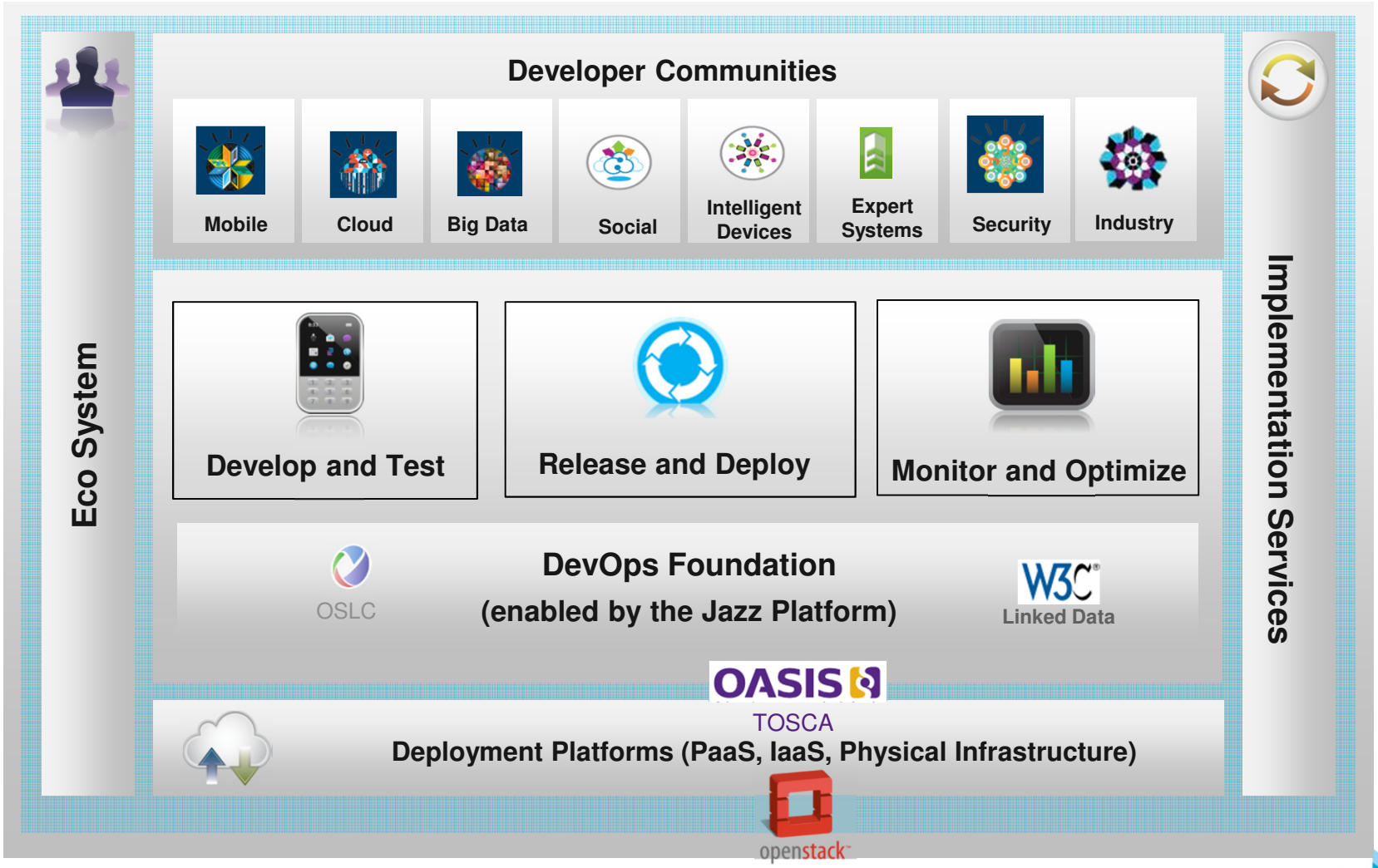




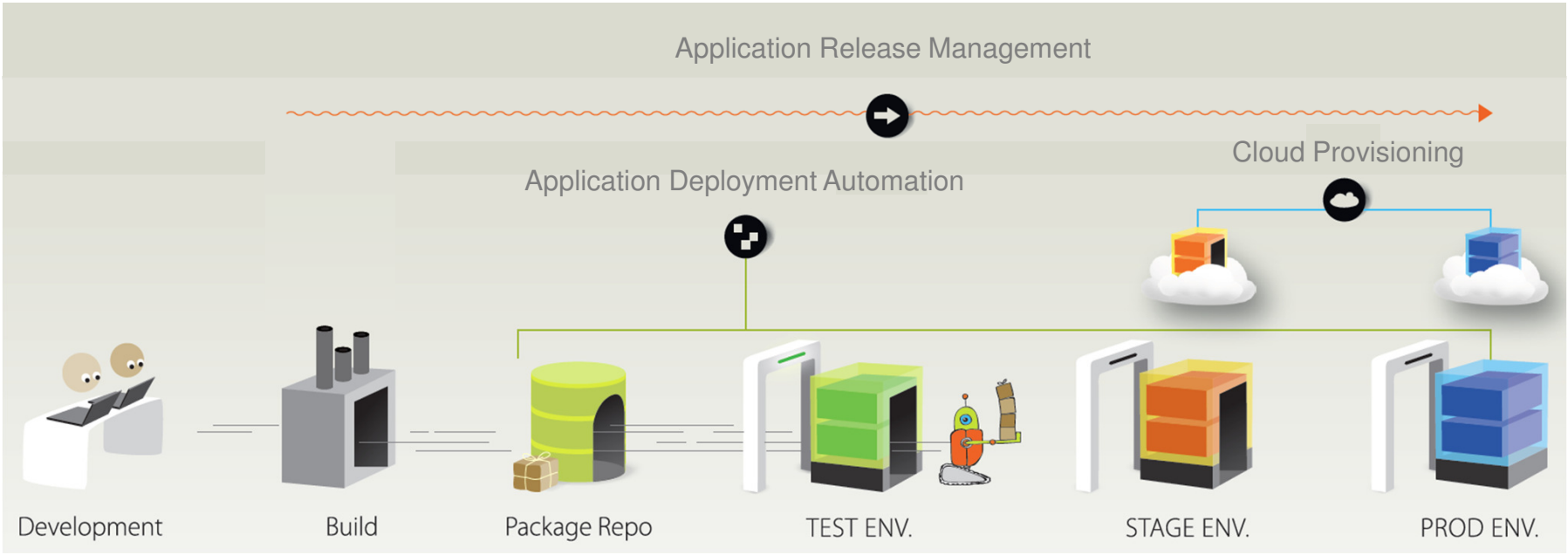
IBM provides expertise across the DevOps lifecycle

- Continuous feedback across heterogeneous environments
- Expertise in enterprise-scale data, security, analytics and instrumentation
- Open-standards-based innovation delivery platform that leverages existing investments
- End-to-end capabilities for managing all aspects of delivery
- Client partnership leveraging know-how and industry expertise to deliver quick ROI
- Solutions that support incremental adoption based on delivery process maturity

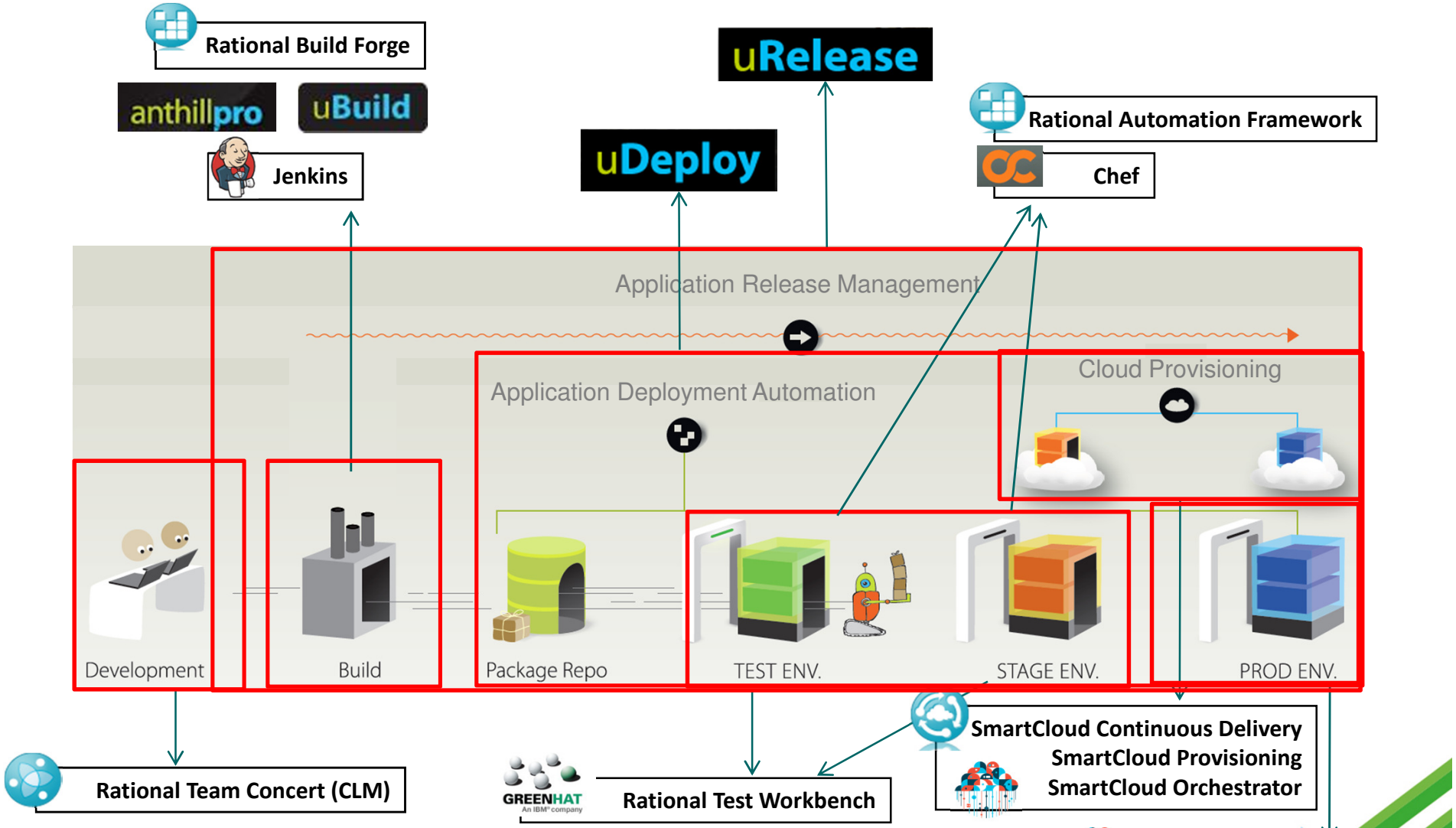
IBM DevOps Reference Architecture



DevOps Tool Chain



DevOps Tool Chain



Delivering real-world results

International Investment Firm
Driving Down Costs

Release process required considerable effort and delayed by manually-introduced errors

- Solution: Automated release process
- Results: Cost avoidance of over $2.3M/year, reduced release time from 2-3 days to 1-2 hours and virtually eliminated test team "down-time"

Online Retailer
Speeding Time-to-Market

Significant delays getting application changes to production

- Solution: Scaled up continuous deployment
- Results: Deployment time reduced by over 95% with easy scale and deploying to over 250 servers within 2 months of implementation

Higher Education
Speeding Time-to-Market

Agile development teams constrained by slow deployment to dev and test environments

- Solution: Accelerate deployment by enabling development teams to self deploy with automation
- Results: Deployments cut from hours to minutes and a greater number of servers with fewer resources

SaaS Software Provider
Reducing Risk

Difficulty managing multiple customer configurations and versions of software deployed across servers

- Solution: Automate managing configuration and version deployment
- Results: Execute customer specific releases, reduced deployment outages by over 90%

Agenda

1. Promise of DevOps
2. Practices in Continuous Delivery
3. Plan to utilize the Cloud

Reduce Cycle-time to deliver business changes across dev/test/ops lifecycles

	Dev	Test	Ops
Value	▪ Self-Service Deployment with reduced overheads to design and test changes ▪ Faster Production feedback with context and fix defects ▪ Early performance in-sights	▪Improved Test coverage ▪Reduced setup overheads ▪Heterogeneous environments	▪ Self-service to Dev/Test ▪ Faster time to deploy Apps ▪ Reduced time to stabilize and post-production defects ▪ Reduced setup and deployment overheads
Incremental Improvements and What can you do with it?	▪ More features delivered ▪ More time for Innovation and experimentation	▪ More test coverage and test iterations	▪ More Apps deployed supporting business ▪Supporting Dev to innovate more
Metric	▪ Avg time to deliver a story ▪ # of Sprints per release ▪ # of defects from test / production	▪ % of Test coverage ▪ # of test iterations ▪ # of defects opened from production	▪ # of Apps Deployed ▪ # of Dev/Test projects supported / person ▪ Time to Stability ▪ Time to fix hot defects
	Business Metrics: Time to deliver a change - Predictability and Variance, Quality metrics		

Who should be interested ?

	Primary interest	Extended interest
Development (VP Dev/Test, CTO)	**Continuous Integration** Target: Development teams using RTC to do continuous integration today, or to become more agile in their delivery of services. Pain: Struggling to quickly spin-up a test environment and deploy application code and associated middleware including configuration. Customer Value: Reduce cycle time from 3 weeks to a few hours	**Continuous Testing** Target: QA / Test teams using Rational Quality portfolio Pain: Teams struggling with setting up test environments for functional, integration and performance test coverage. Customer Value: Setup test environments 90% faster
Operations (CIO, Enterprise Architect)	**Continuous Deployment** Target: Operations or release teams using IBM private cloud or virtual environments Pain: Unable to keep up with Dev/Test requests to setup and configure environments – infrastructure, middleware and applications. •Customer Value: Deliver environments 3X faster	**Continuous Monitoring** Target: Operations teams experiencing a high rate of production rollbacks Pain: Application defects and performance impacting production release readiness and efficiencies Customer Value: 30 - 40% reduction in outages