IBM Software

Innovate2012

The Premier Event for Software and System Innovation

Next NOW!

DevOps: Extending Agile Development Discipline to Deployment

Chandra Venkatapathy
Sr. Product Manager, Rational

Please note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Delivery Challenges

Today's business and technical needs are pushing traditional delivery approaches to the breaking point

People

Process

Information

Customers

Desire for fast and continuous innovation

Line of Business

Requirements

1st Gap

Dev & Test Teams

Code & Tests

2nd Gap

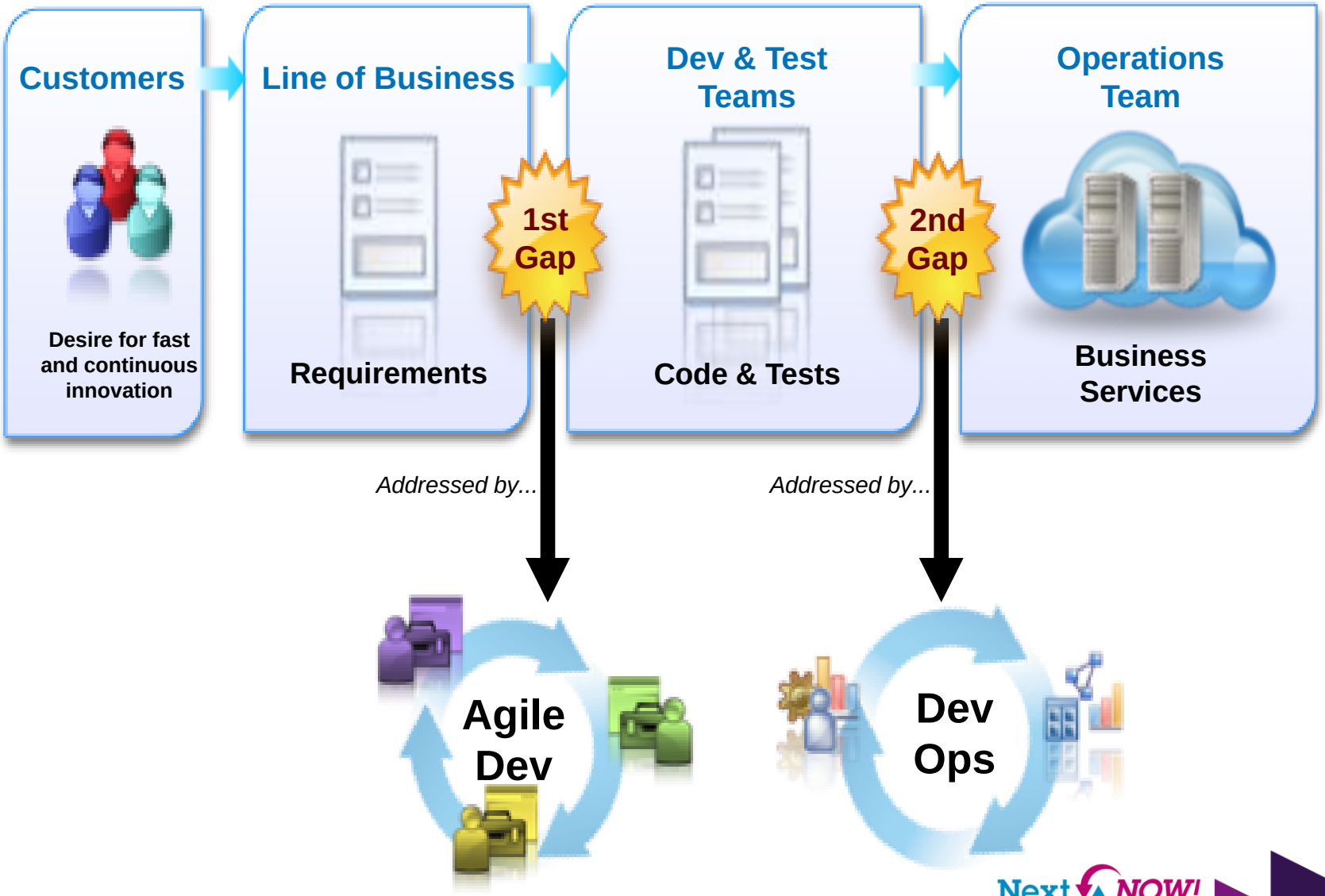
Operations Team

Business Services

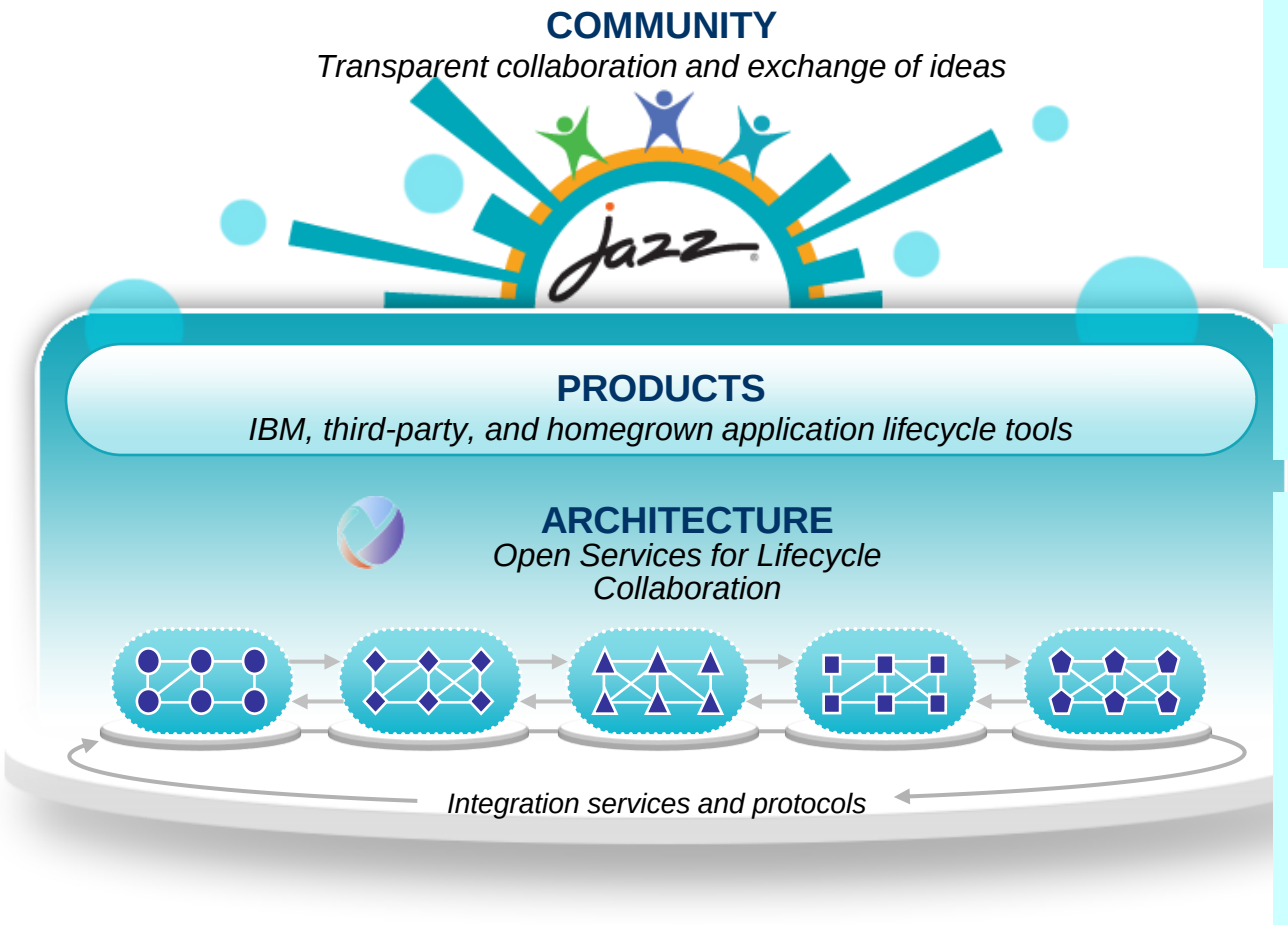
"At some point, you take a step back, and you realize you have an awful lot of **siloed systems** that are **limiting transparency** across strategic projects."

- Development Director
Temenos, Inc.

Addressing Application Lifecycle Management gaps



Jazz provides open collaboration across the software and systems lifecycle



Jazz.net – A place where stakeholders collaborate

Enables visibility & influence into the evolution of the Jazz architecture and products

Lifecycle tools that support the Jazz architecture

Specifications for linked lifecycle data via Open Services for Lifecycle Collaboration (OSLC)

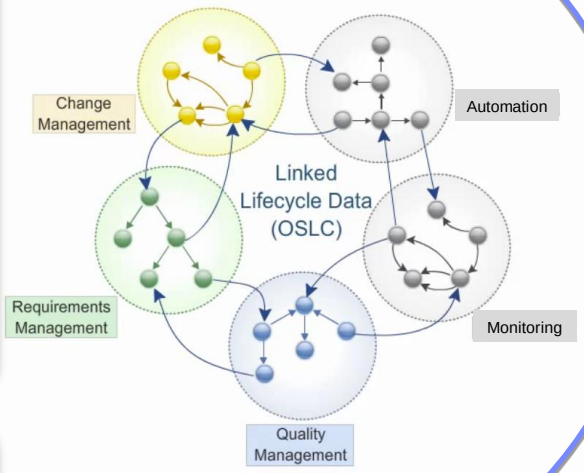
Integration services & protocols for implementing common lifecycle patterns (“the Jazz platform”)

Open Services for Lifecycle Collaboration (OSLC)

Working to standardize the way software lifecycle tools share data

Open Services for Lifecycle Collaboration
Lifecycle integration inspired by the web

- Community Driven – @ **open-services.net**
- Specifications for numerous disciplines
  - Such as, ALM, PLM and DevOps
  - Defined by scenarios – solution oriented
- Inspired by Internet architecture
- A different approach to industry-wide proliferation
- Based on W3C® Linked Data



Inspired by the web | **Free to use and share** | **Changing the industry**

GET INVOLVED AND CONTRIBUTE!

OSLC is expanding

Specifications

Core and common

Core	v2	Scope	Draft	Converge	**Finalize**	Wiki →
Reporting	v1	Scope	Draft	**Converge**	Finalize	Wiki →

Application lifecycle management

Change Management	v2	Scope	Draft	Converge	**Finalize**	Wiki →
Quality Management	v2	Scope	Draft	Converge	**Finalize**	Wiki →
Requirements Management	v2	Scope	Draft	Converge	**Finalize**	Wiki →
Asset Management	v2	Scope	Draft	Converge	**Finalize**	Wiki →
Architecture Management	v2	Scope	Draft	Converge	**Finalize**	Wiki →
Software Configuration Management	v1	Scope	Draft	Converge	**Finalize**	Wiki →
Automation	v1	Scope	**Draft**	Converge	Finalize	Wiki →

Software project management

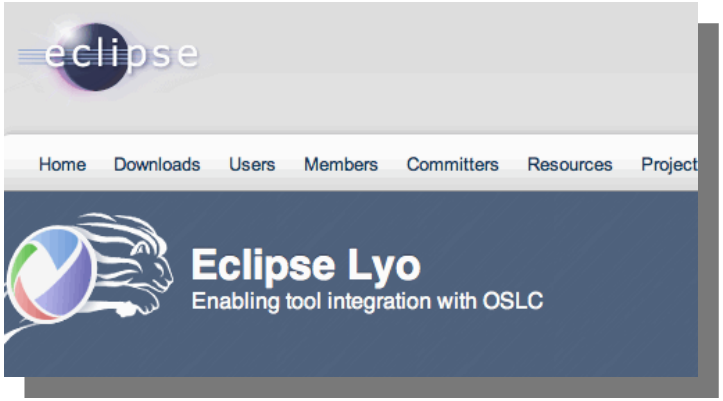
Estimation and Measurement	v2	Scope	Draft	**Converge**	Finalize	Wiki →

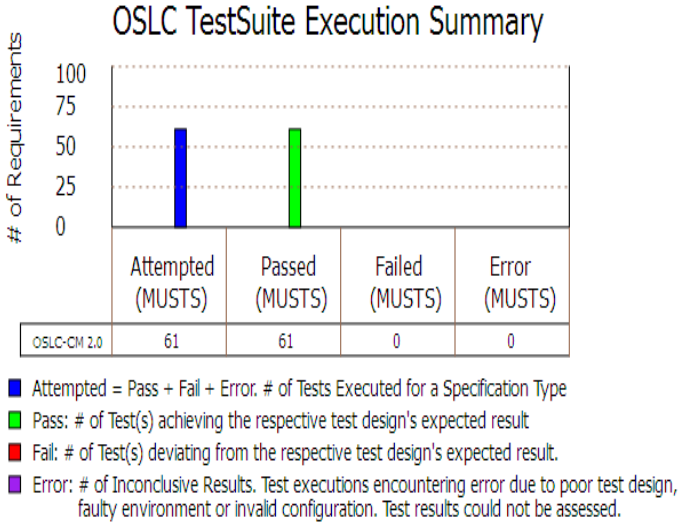
Product lifecycle management

Product Lifecycle Management	v1	**Scope**	Draft	Converge	Finalize	Wiki →

Integrated service management

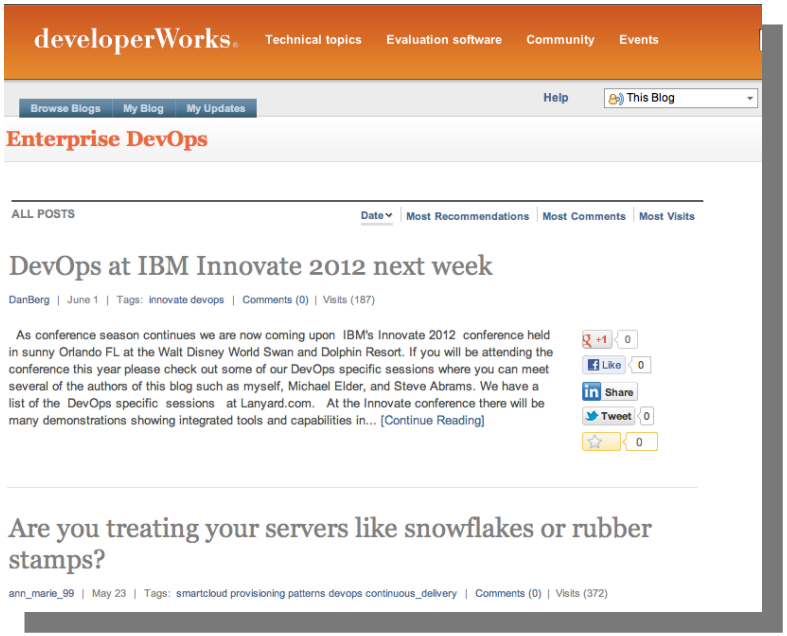
Performance Monitoring	v1	**Scope**	Draft	Converge	Finalize	Wiki →

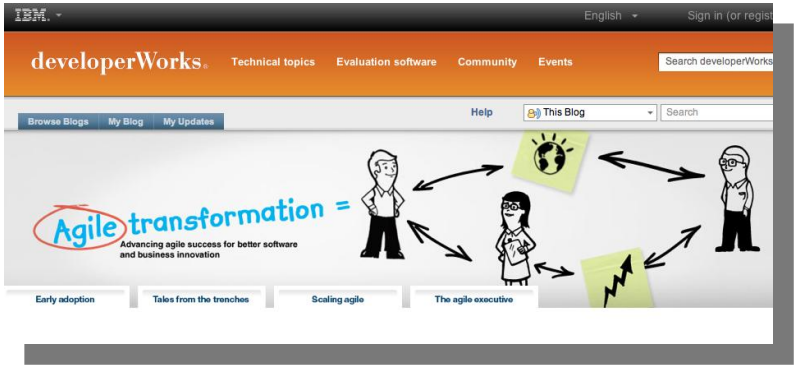




Communities

- Transparent development
- Thought leadership
- Collaboration and support



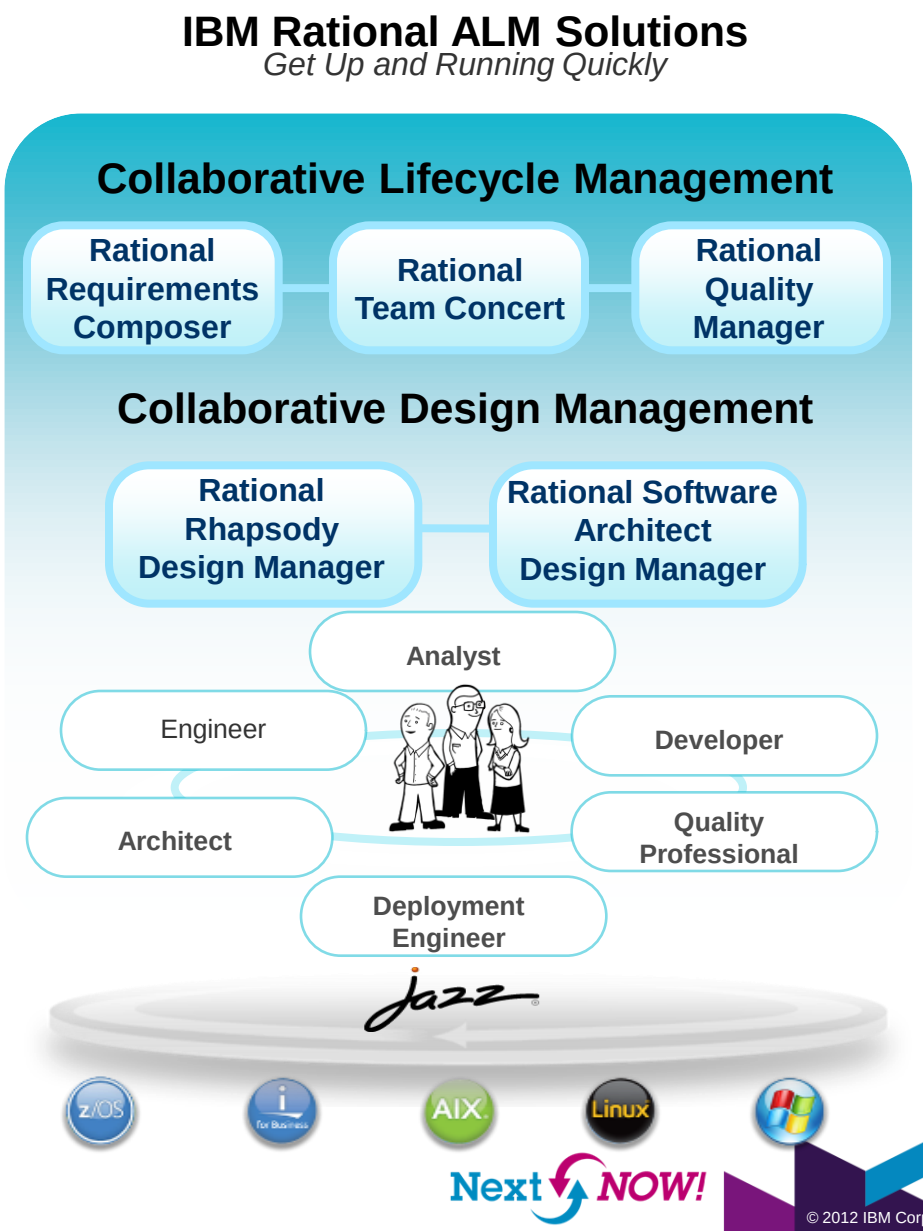


Next NOW!

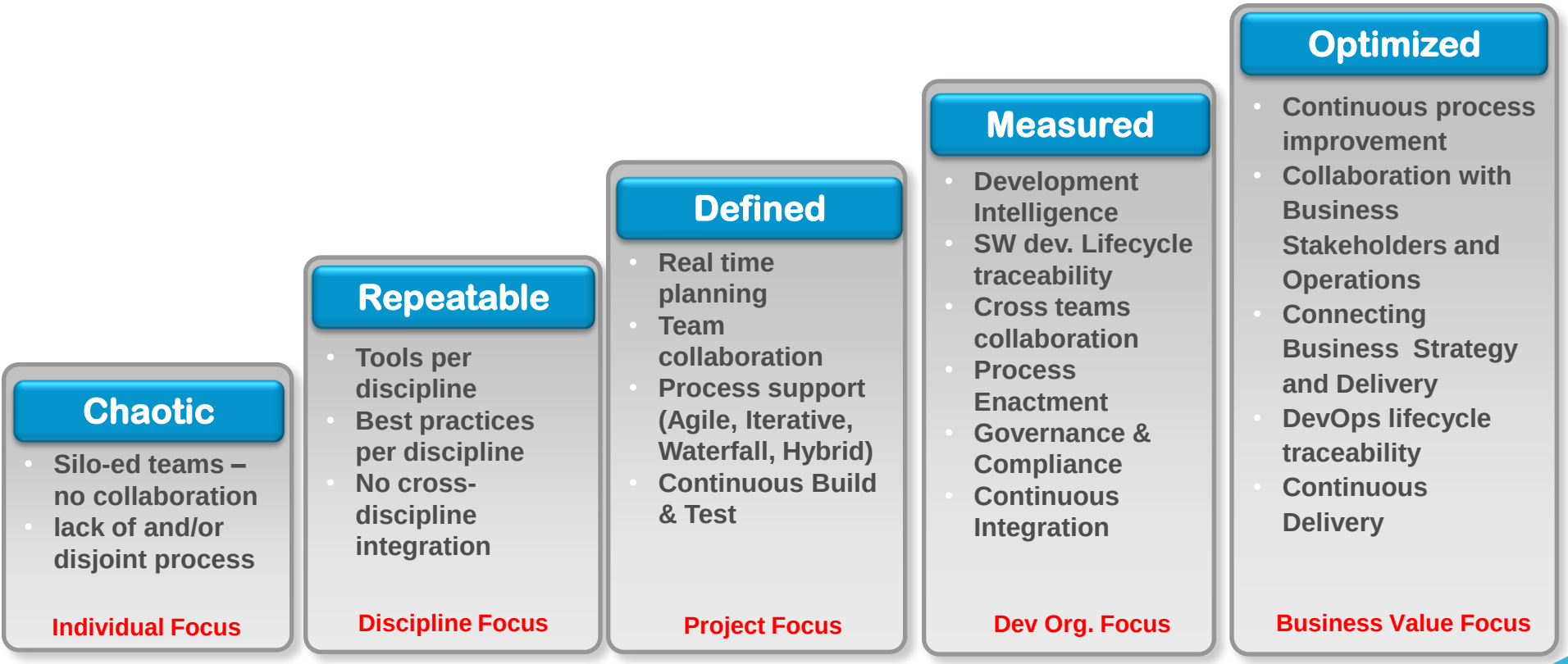
Application Lifecycle Management offerings

Reduce the costs of inefficient, multiplatform software development with integrated ALM solutions!

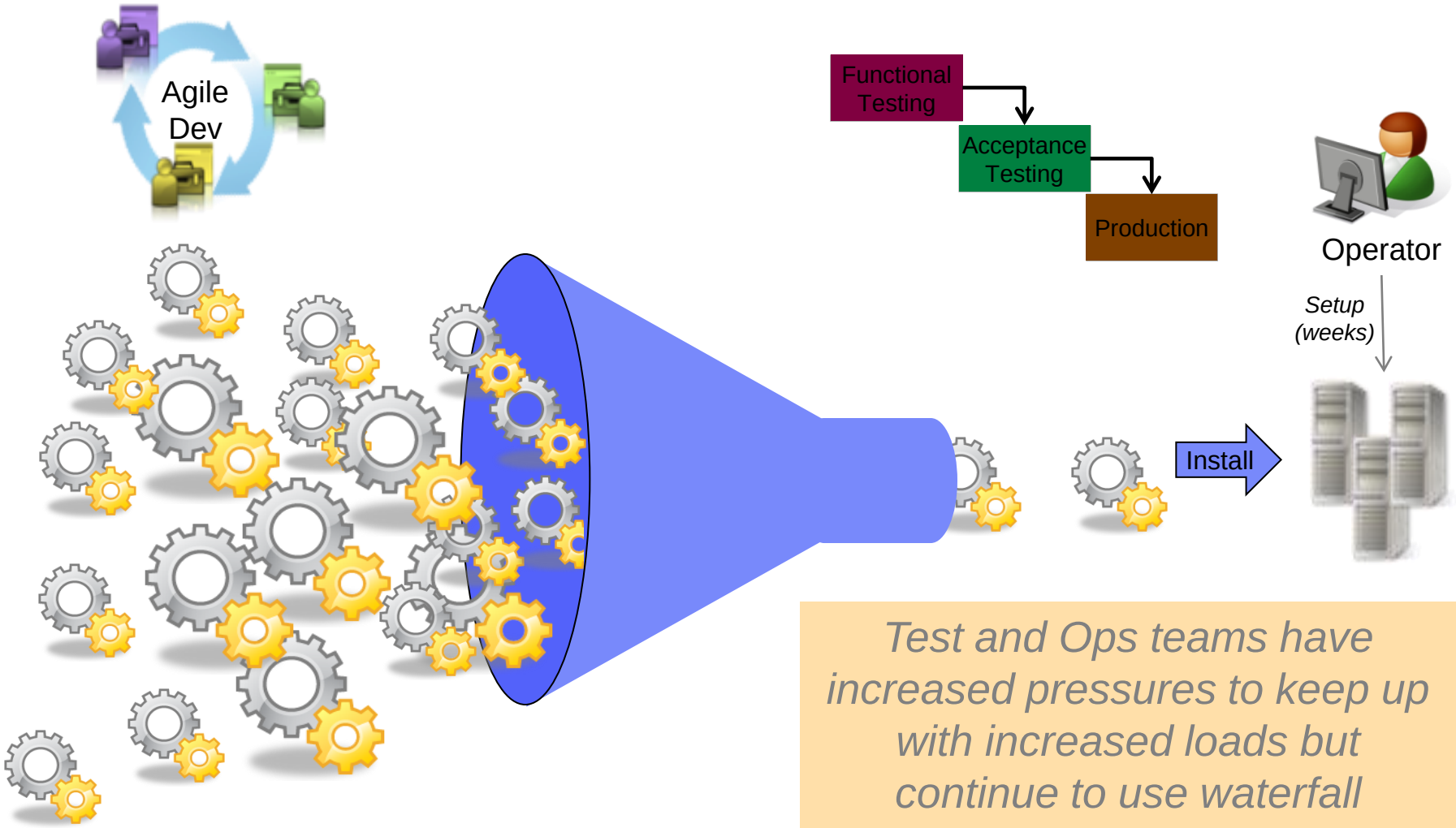
- Optimize your team's productivity through the 5 ALM Imperatives
- Maximize product value with **In-Context Collaboration**
- Accelerate time to delivery with **Real-Time Planning**
- Improve quality with **Lifecycle Traceability**
- Achieve predictability with **Development Intelligence**
- Reduce costs with **Continuous Improvement**
- Collaborate across teams and create deep integrations across the lifecycle
- Extend as your needs evolve with role-based licensing
- Unify your infrastructure and protect your current investments with a single, open, extensible platform



Application lifecycle Management adoption steps



With only Agile Development improvements…

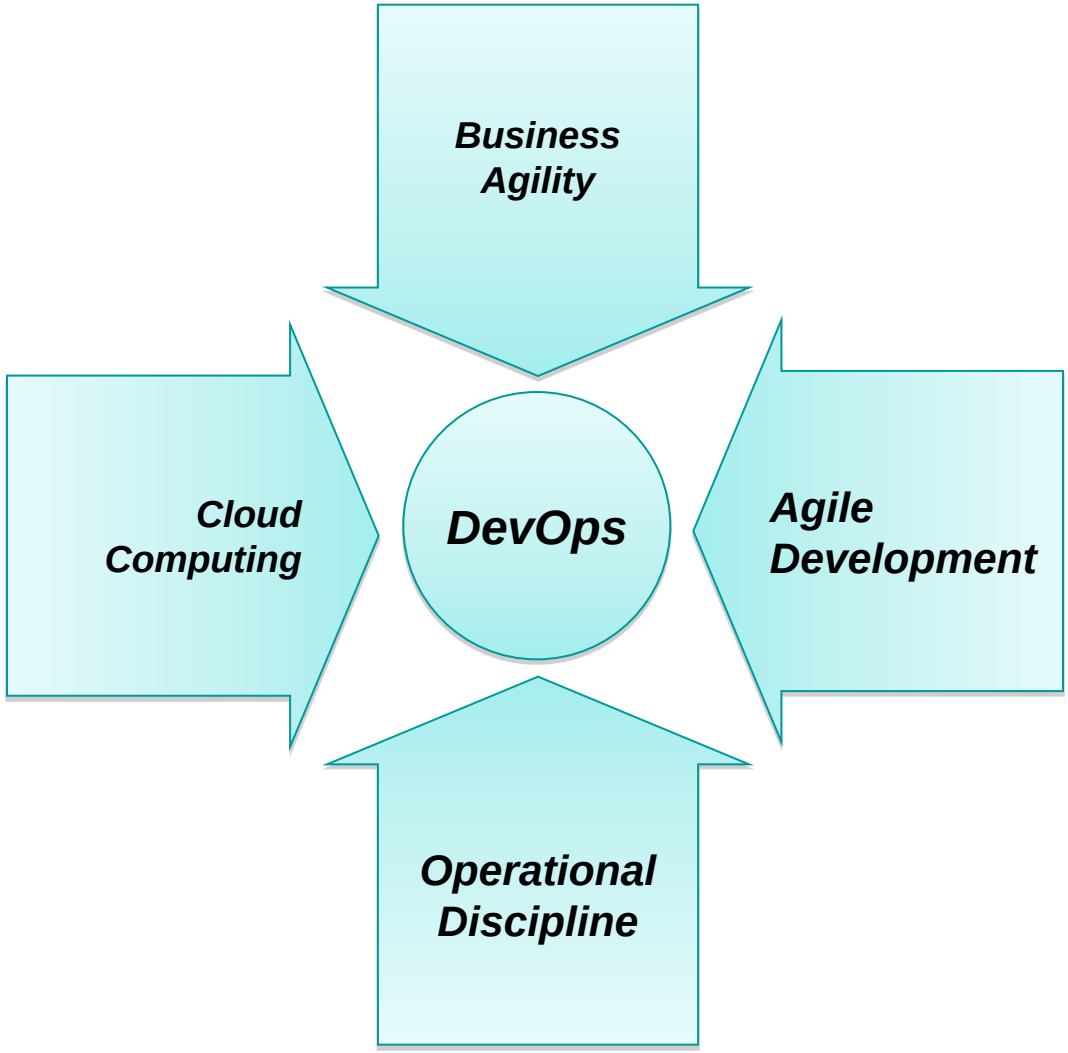


CI builds are piling up

Test and Ops teams have increased pressures to keep up with increased loads but continue to use waterfall approaches and traditional tools.

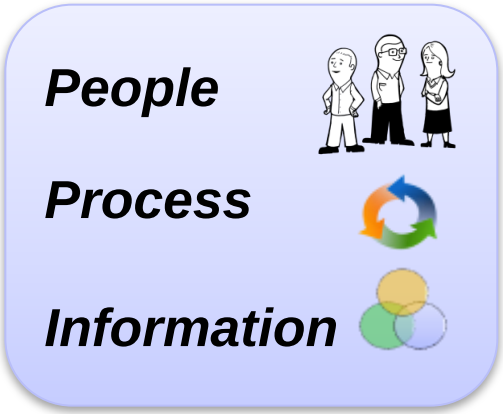
DevOps: The time is now

Four key drivers are making DevOps a 2012 imperative for all organizations.



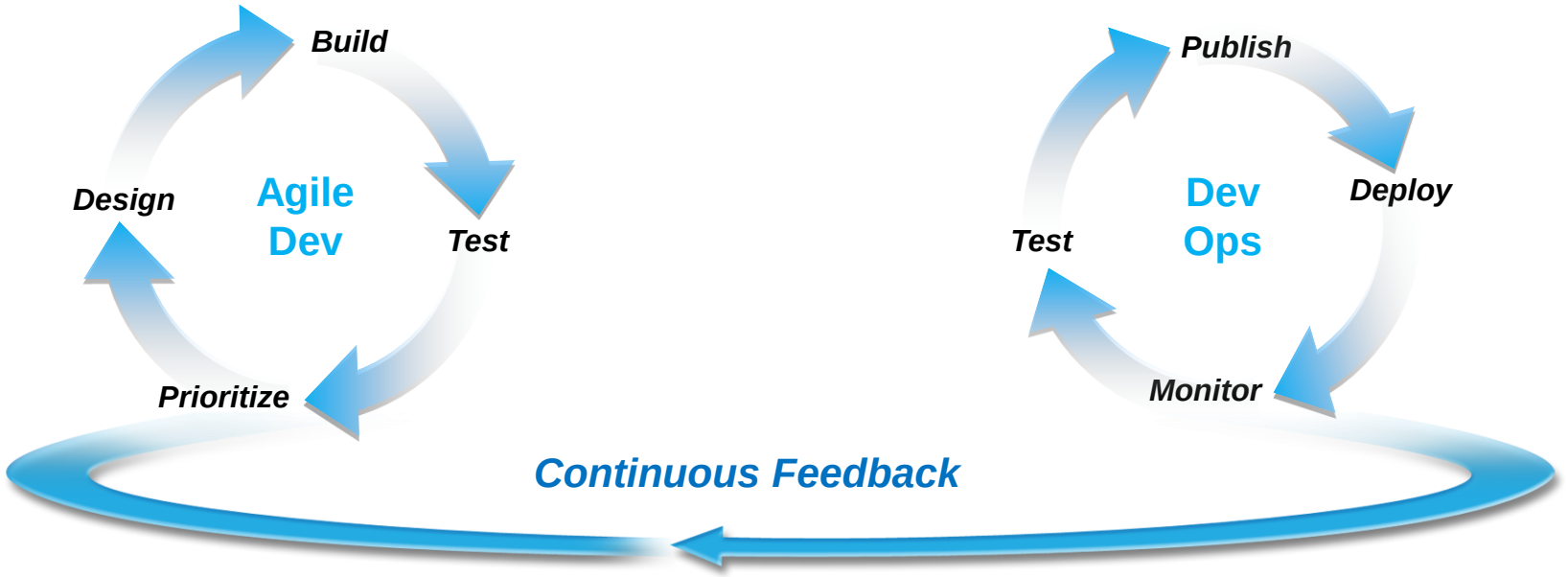
DevOps Principles & Values

- Collaborate across disciplines
- Develop and test against a production-like system
- Deploy frequently
- Continuously validate operational quality characteristics



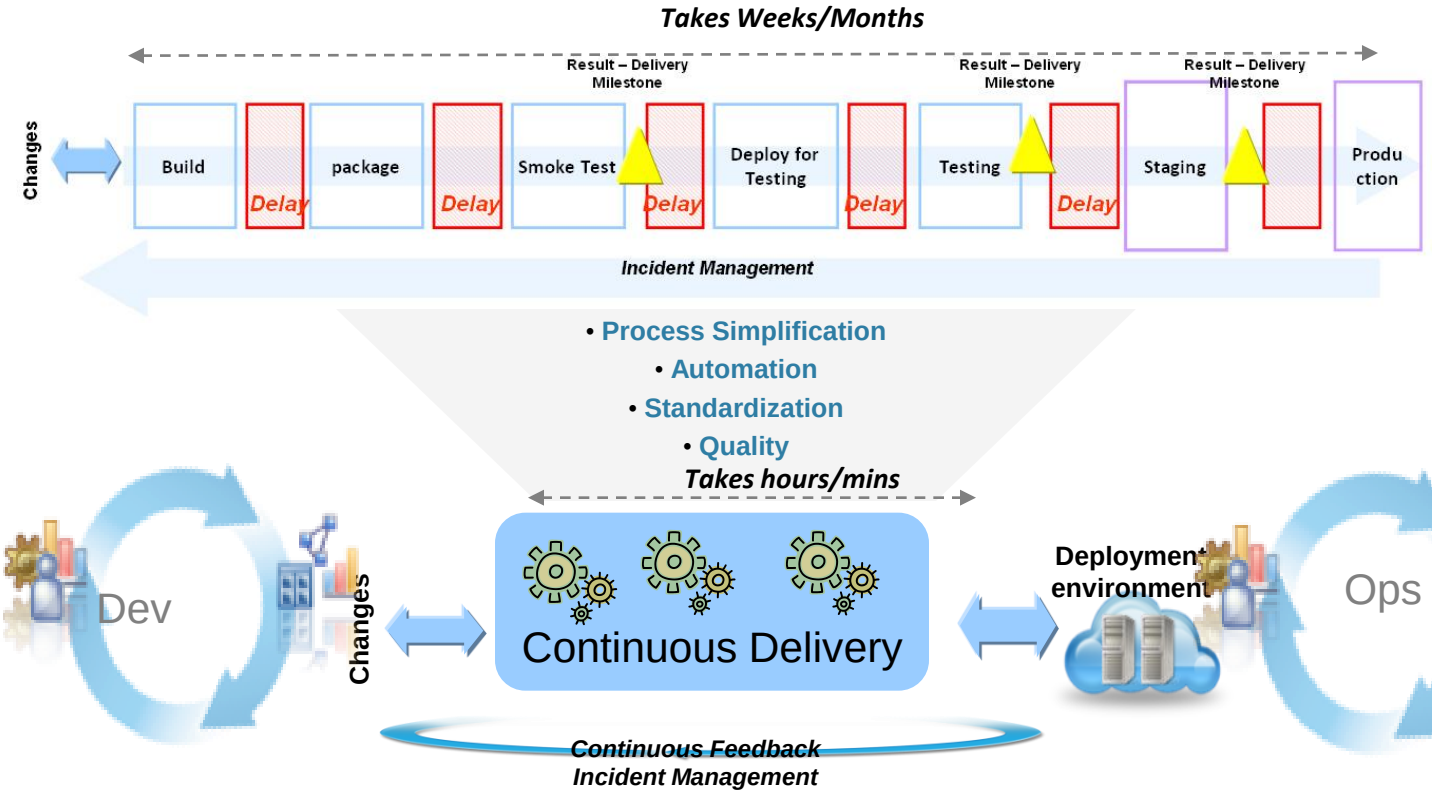
Agile development and delivery

Continuous Integration extends to Continuous Delivery



Need for a Simple approach to bringing agility across the lifecycle

Continuous and automated delivery of changes leveraging Cloud



- Long Cycle Time
- Poor Visibility and Control

- Reduced cycle time
- Control, Agility and scalability

Installation Instructions

RedHat Linux

1. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

2. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Apache Web Server

1. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.

2. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur,

3. adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem.

Python

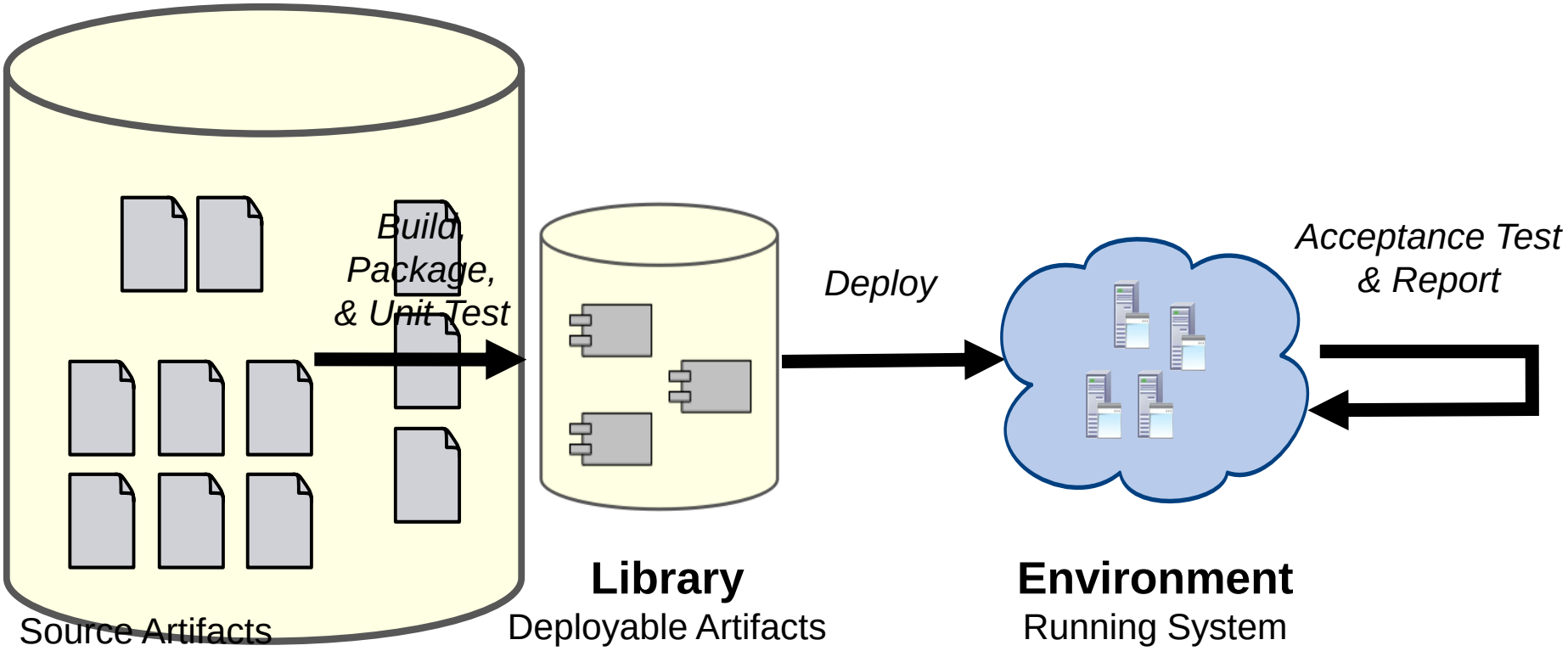
1. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur?

2. Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur,

3. vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

```
#!/usr/bin/env ruby

class DevopsDeployer
  def initialize(build_url, build_id)
    @log = Logger.new(LOG_FILE)
    @log.level = LOG_LEVEL

    @iaas_gateway = IaasGateway.new(HsltProvider.new(),
LOG_FILE, LOG_LEVEL)
    @server_instance = nil

    rtc_build_system_provider = RtcBuildSystemProvider.new(
RTC_REPOSITORY_URL, RTC_USER_ID, RTC_PASSWORD_FILE)
    @build = rtc_build_system_provider.resolve_build(
build_url, ENV['buildResultUUID'], build_id)
    @build_system_gateway = BuildSystemGateway.new(
rtc_build_system_provider, LOG_FILE, LOG_LEVEL)
  end

  def add_build_stamp
    template_file = WEB_APP_ROOT +
"/app/templates/pages/page.html"
    @log.info "Adding build ID stamp #{@build.id} to \
#{template_file}"

    # Read in the file's contents as a string, replace
    # the build_id, then overwrite the original contents
    # of the file
    text = File.read(template_file)
    new_text = text.gsub(/\{\{ build_id \}\}/,
"<a href=\"#{@build.uri}\">#{@build.id}</a>")
    File.open(template_file, "w") { |file|
      file.puts new_text
    }
  end

# ...
```

Delivery Pipeline



SCM

Using the same tools and methodologies to manage and deliver software and deployment configuration changes.

Introducing IBM SmartCloud Continuous Delivery

Beta

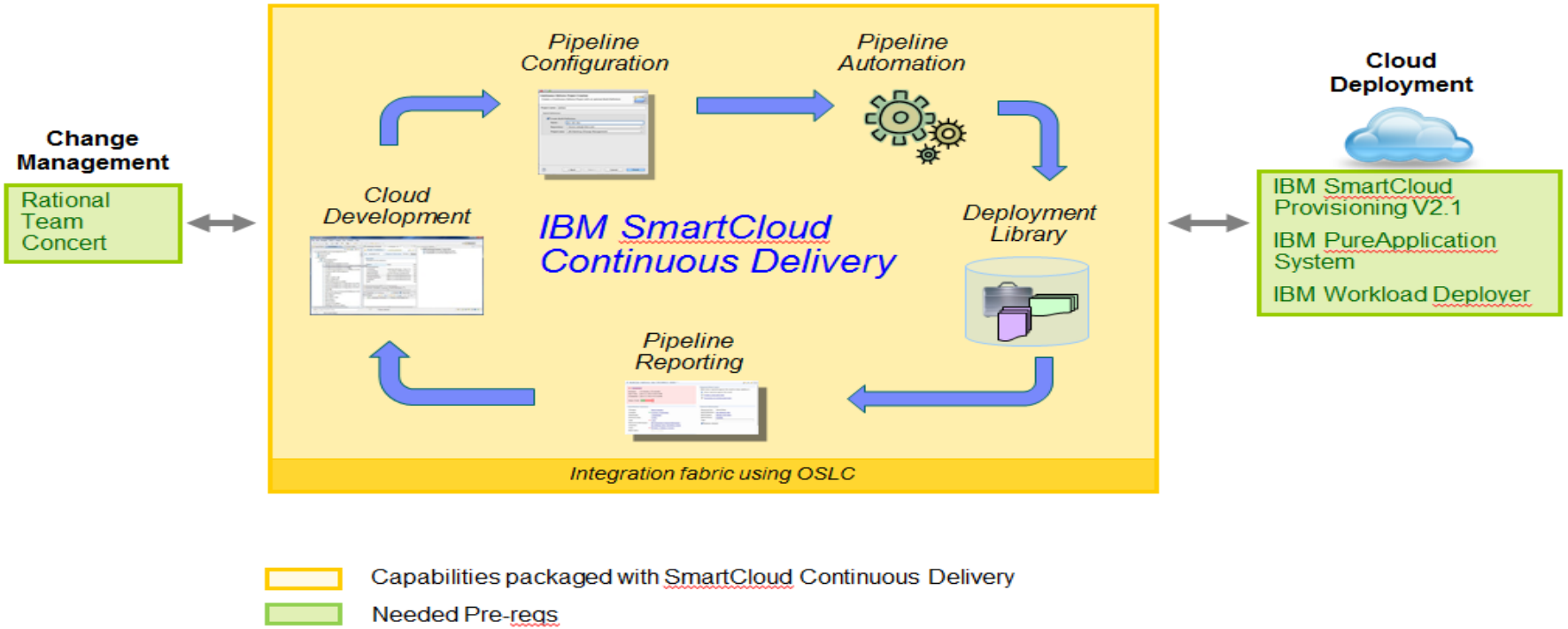
Collaborate: Dev and Ops co-develop app environment definitions and patterns

Dev and Ops use the same metrics in Dev/Test/Prod

Integrate: Continuously integrate , test & automate build changes onto standard cloud environments

Continuously deploy app changes into multiple (distributed and mainframe) environments

Optimize: Measure velocity of change based on agile processes supported by DevOps

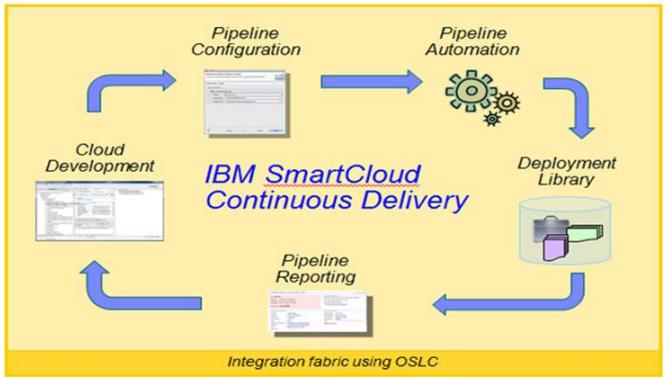


Scenarios

2 - Reliable Application Delivery - Deploy app changes into multiple environments with consistency and quality using patterns

3 - Automated Pipeline Delivery **–** Continuous and automated build and testing of app changes onto std. cloud environments

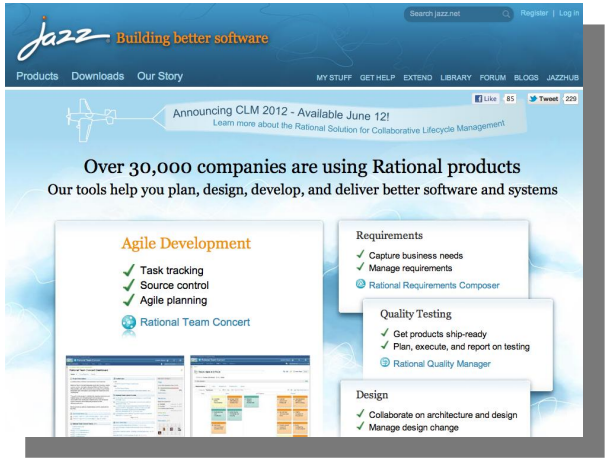
1 - Environment Version Mgmt **–** Dev and Ops co-develop app environment definitions and patterns used across Dev/Test/Prod



4 - Integrated Performance Management - Dev and Ops use the same instr. and metrics in Dev/Test/Prod (APM designed with app)

5 - Visibility , Control and Metrics - Measuring velocity of change based on agile processes supported by DevOps

Self-hosting CLM using Continuous Delivery



Increased Frequency

Deliver

Collaborative Lifecycle Management

Rational Requirements Composer | Rational Team Concert | Rational Quality Manager

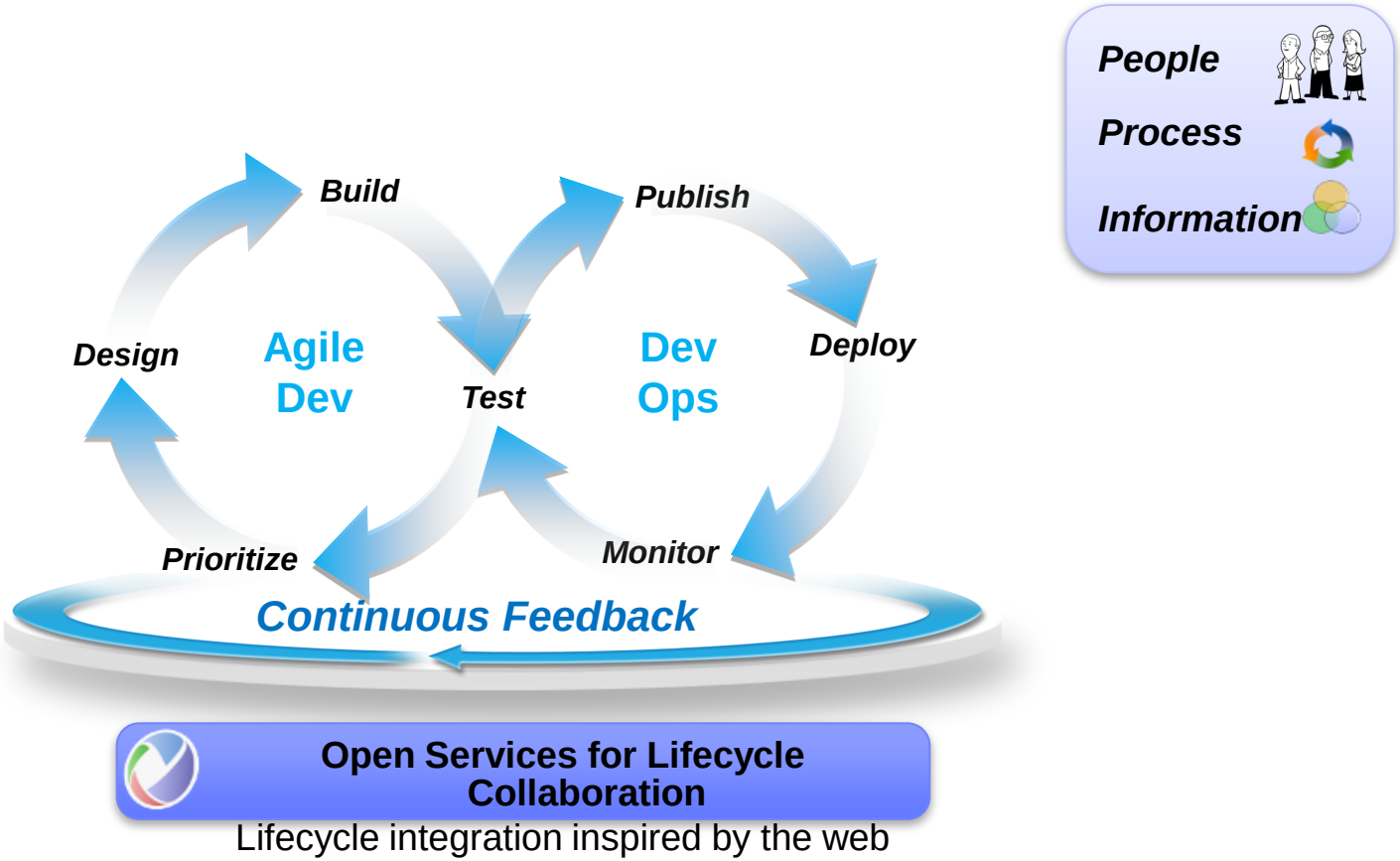
Scaling
Failover
Load Balancing

Monitoring
Security

Lifecycle Management

Cloud

End-to-End Lifecycle Optimization



Follow us on the Enterprise DevOps and Jazz Team Blogs

www.ibm.com/software/rational

Acknowledgements and disclaimers

Availability: References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

www.ibm.com/software/rational

Companies Considering Agile Need Help

Low Success Rates

42%

The percentage of agile project that are successful

10%

The number of projects that can actually prove why they were successful

Increased Inhibitors

77%

Of all companies that report they need to monitor and measure mixed environments

75%

The total percentage of companies citing geographic distribution, regulatory compliance or management support as a key inhibitor

Water-Scrum-Fall

78%

Percentage of the organization who feel they can't keep up with an agile development team

26%

Estimated number of organizations who use agile methodologies ONLY in development

Sources: NIST, Planning Report 02-3. The Economic Impacts of Inadequate Infrastructure for Software Testing, May 2002;
[a]The Times of India, IT sector to get 12% average salary hike in 2011, TOI Tech & Agencies, Mar 8, 2011

2

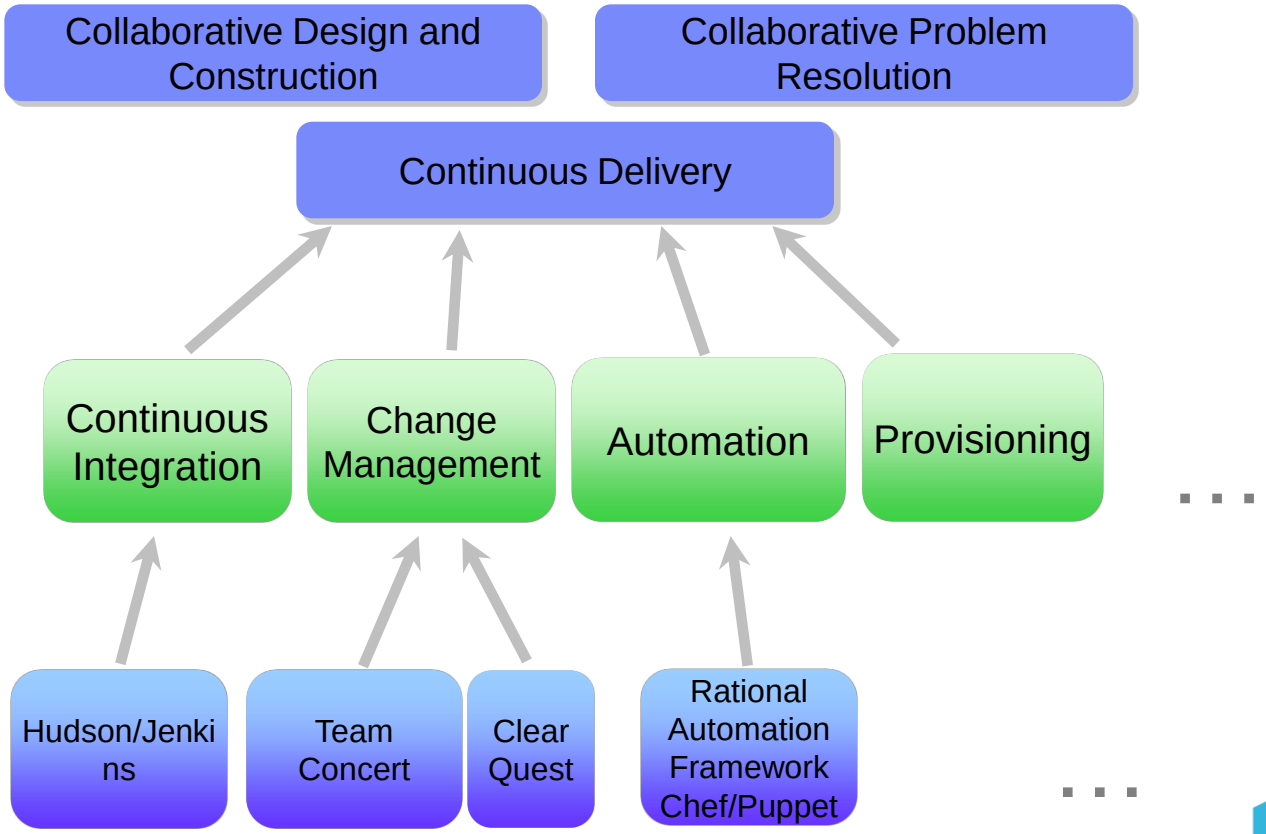
Principles & Values

Scenarios & Solutions

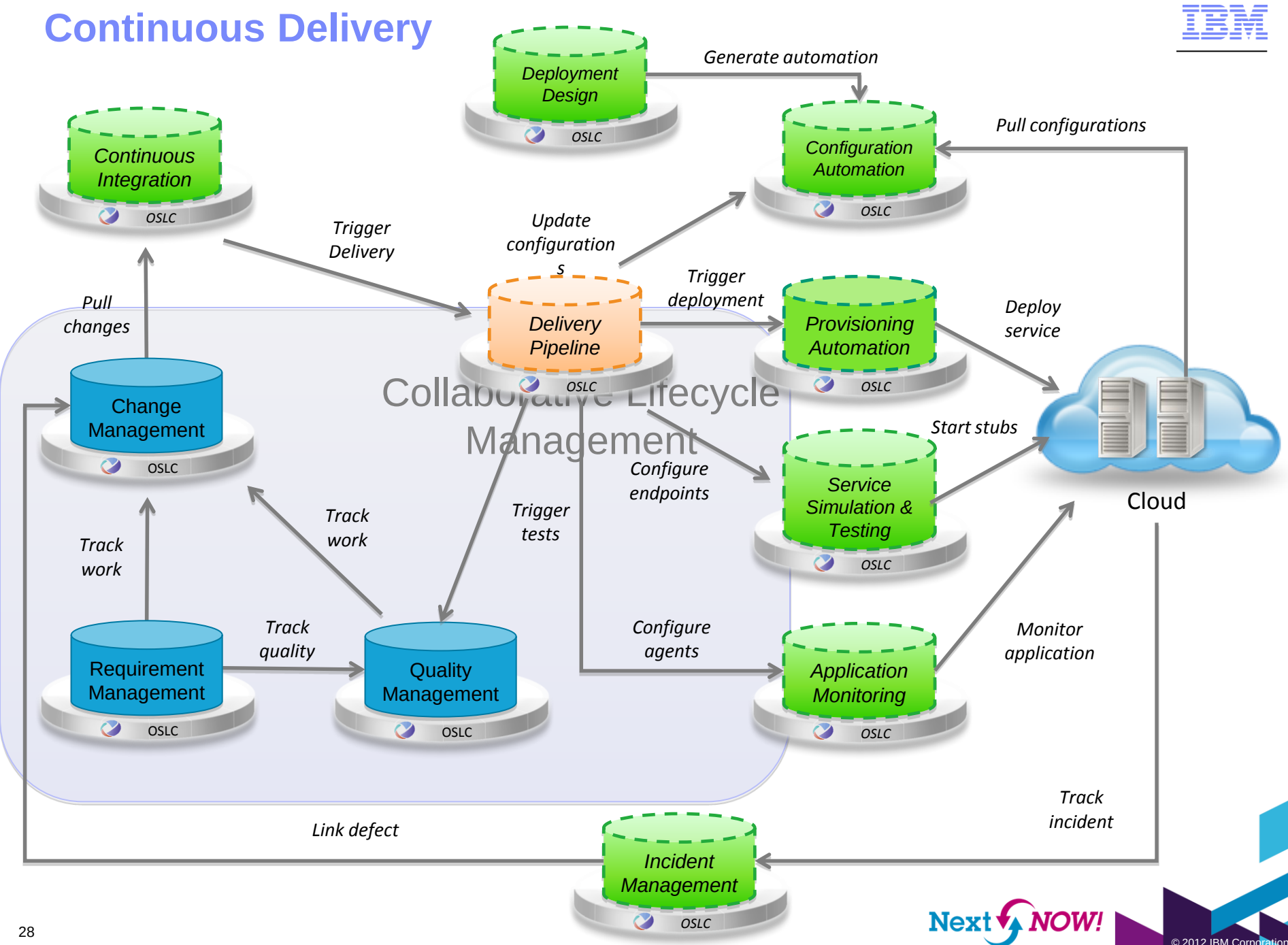
Integration Architecture

Tools

DevOps

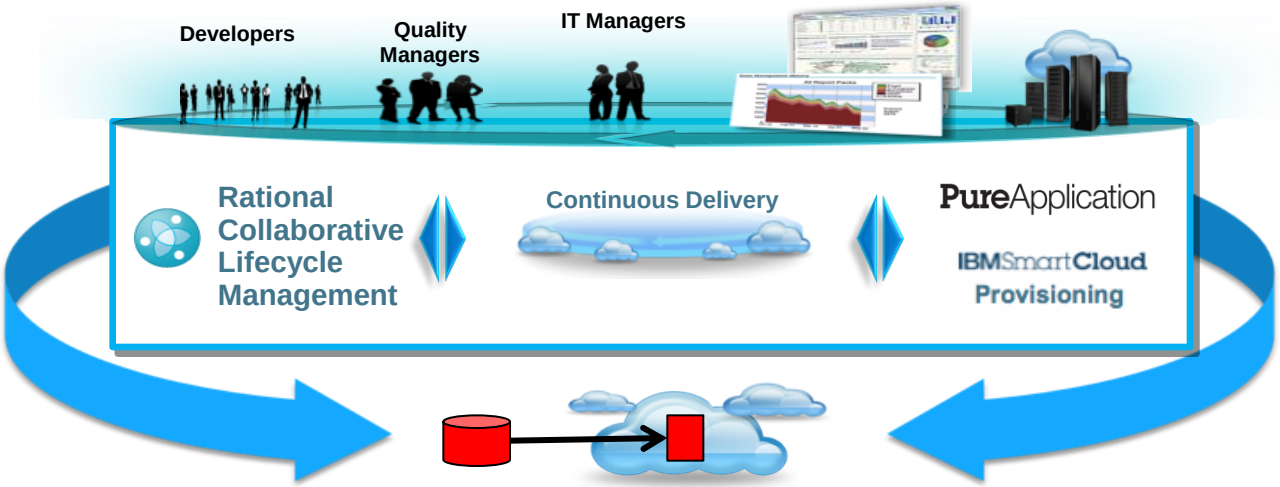


Continuous Delivery



IBM SmartCloud Continuous Delivery

Extending Agile disciplines through delivery



Client Value

- Reduce risk, improve quality; manage change from development to deployment
- Improve efficiency, accelerate delivery; automated handover between processes
- Optimize resources; workload pattern composition delivery

Targeted Entry

- Development team extending Agile into rapid workload deployment in the cloud
- Operation teams delivering scalable, continuous delivery services to the development organization

Continuous Delivery of application changes

Put configuration version

Rational Automation Framework

Apply configurations to system

Load changes and build

Jazz Build Engine(s)

Publish packages

Rational Asset Manager (Deployables Library)

Post results to build

Rational Team Concert (server)

Trigger build

Get packages

Execute tests

Notify results

Store changes

Rational Team Concert (Client)

Provision pattern with build

Virtual System

Provision system from pattern with build

Cloud Provisioning System

PureApplication

IBM SmartCloud Provisioning