IBM Software

Innovate2012

The Premier Event for Software and System Innovation

Next NOW!

Current Trends and Future Directions in Technology for DevOps

Saurabh Agarwal

Product Engineering Team , DevOps Solutions

agarwasa@us.ibm.com

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.

Outline

- Business promise of tools
- Business and Development
- Development and Operation

The Business Promise of Tools Is Widely Anticipated

Companies acquire tools with the best of business-centric aspirations

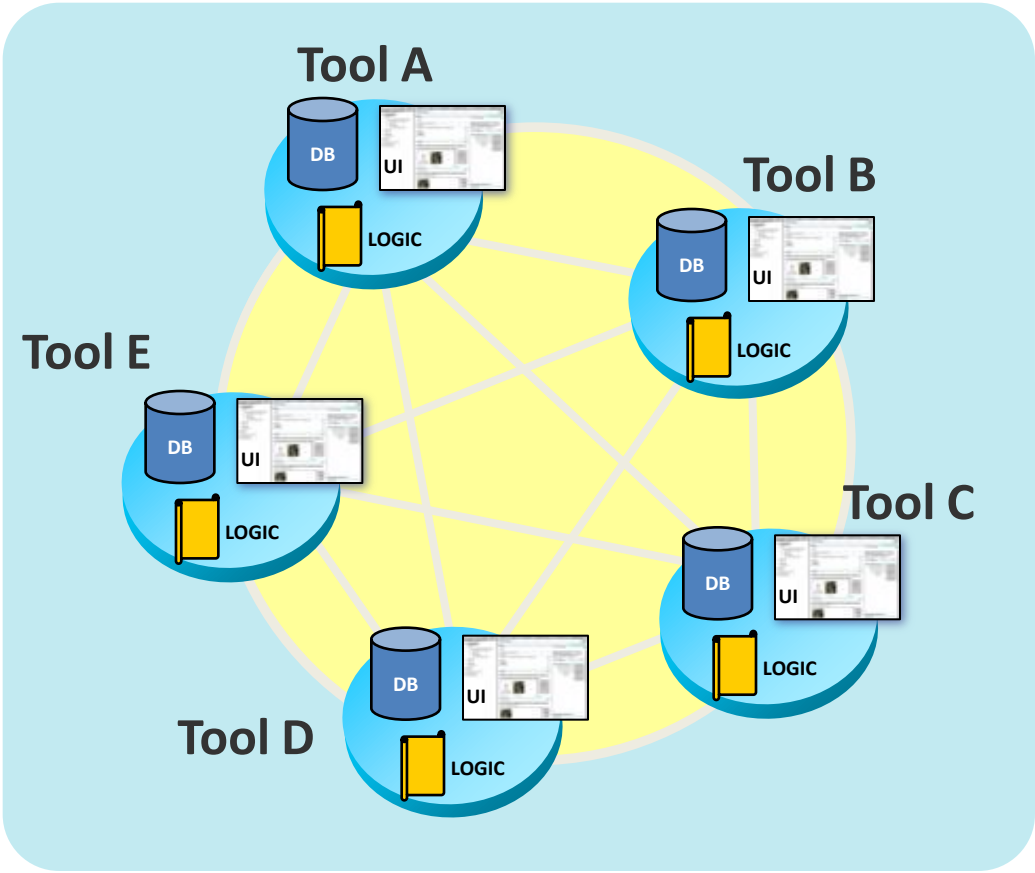
- Higher quality
- More customer satisfaction
- Aligning business and IT
- Faster time to market
- Lower costs/higher productivity
- More predictable delivery

Reality adds significant complexity

- Many tools from many vendors
  - **Heterogeneous environments** that are flexible for partners and suppliers
- Many teams in many places
  - **Distributed** development, cross site product development
  - Many levels of teams

    PMO, Bus, dev teams, ops teams, etc
- Coherent process
  - **Flexible** and **robust** process

    supporting Lifecycle / Agile Methods
  - **Measure** and **improve** effectiveness



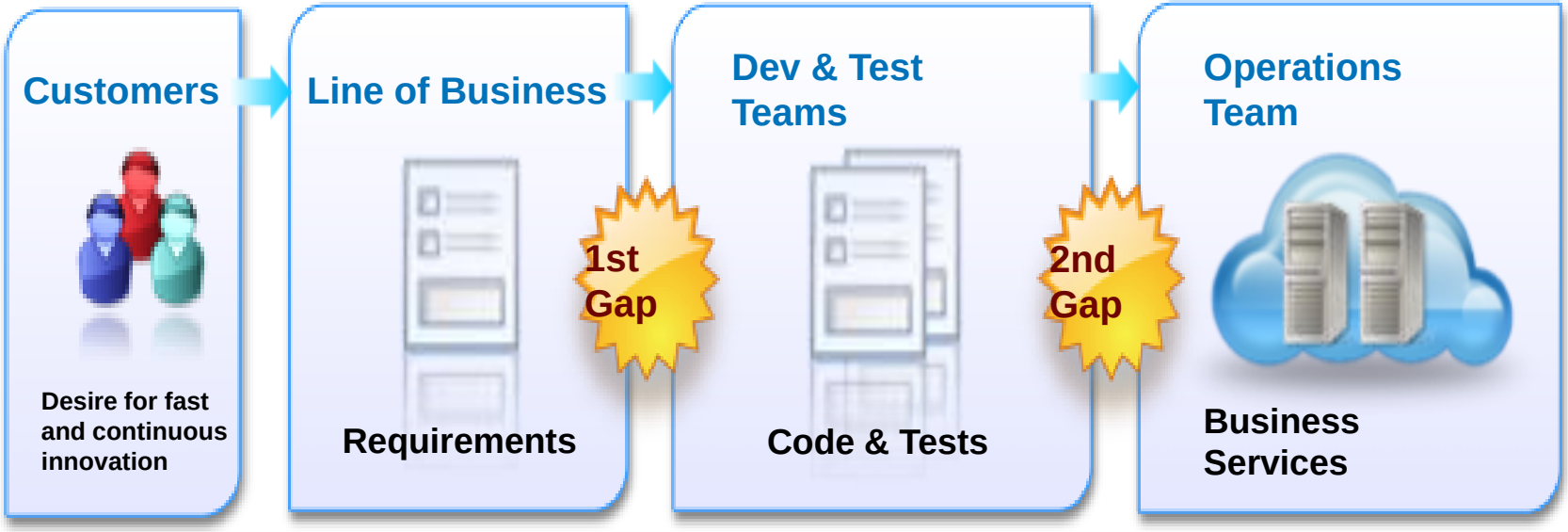
Delivery Challenges

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

People

Process

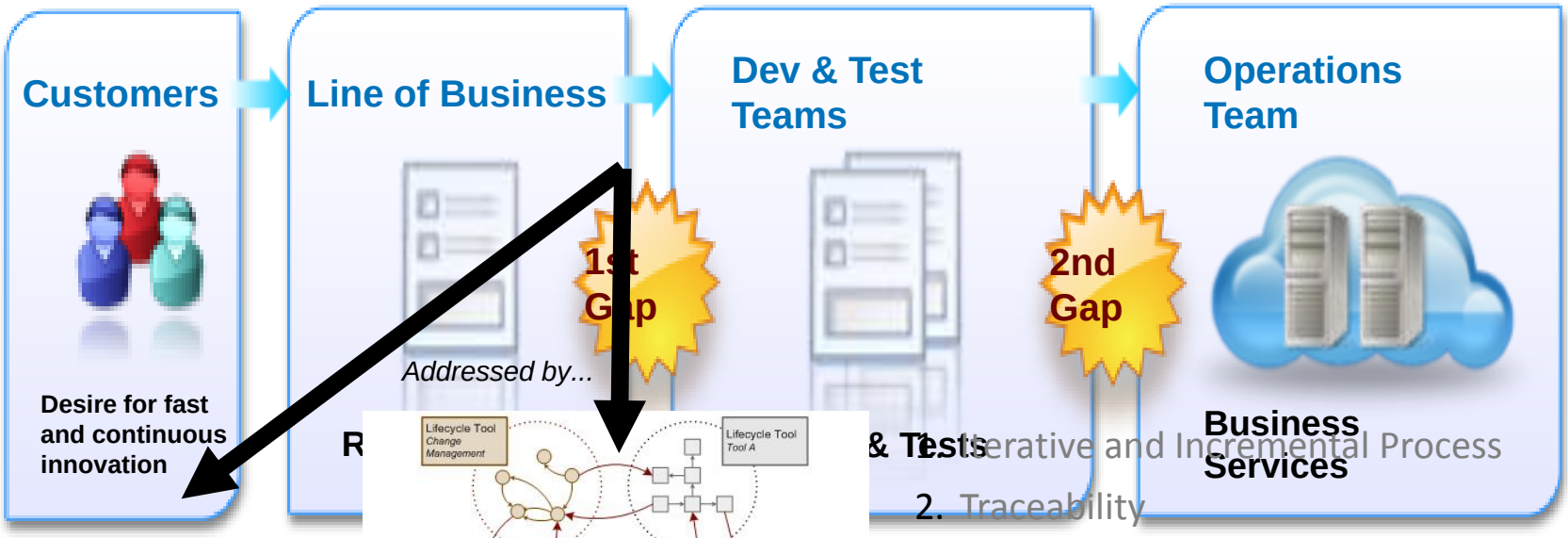
Information



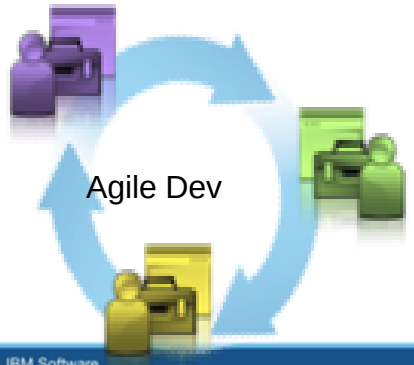
"At some point, you take a step back, and you realize you have an awful lot of **siloed systems** that are **limiting transparency** across strategic projects."

- Development Director
Temenos, Inc.

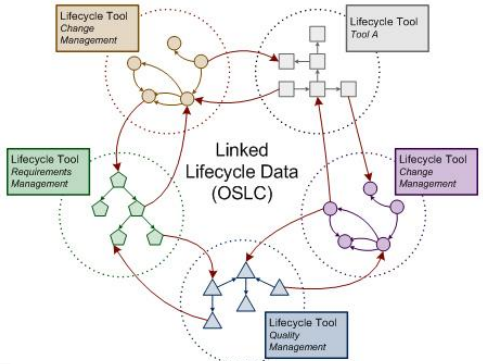
Addressing *BusDev* gaps

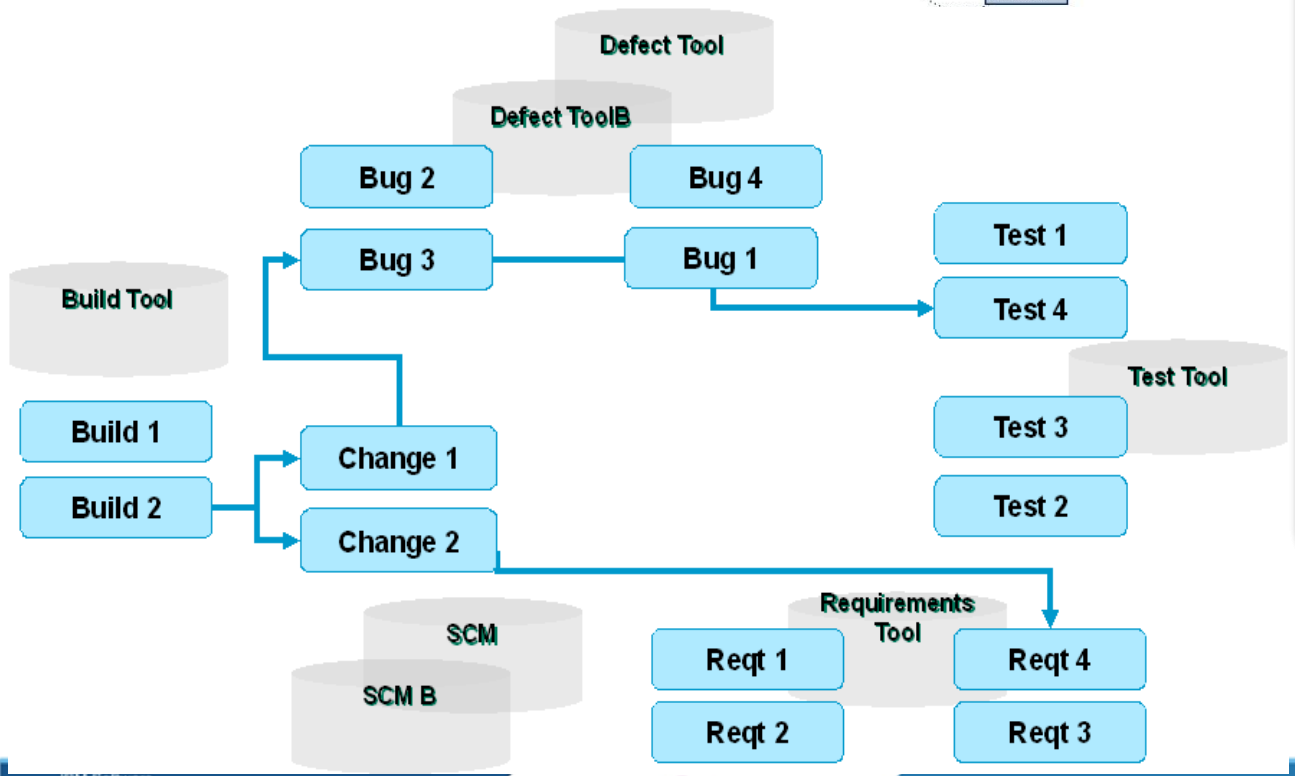


1. Iterative and Incremental Process
2. Traceability



Linked Lifecycle Data





- The data is the thing
  - Resources and relationships
  - Tools operate on the data
  - Tools execute the process
  - Tools expose their data in a common way (REST)
- Lifecycle integration:

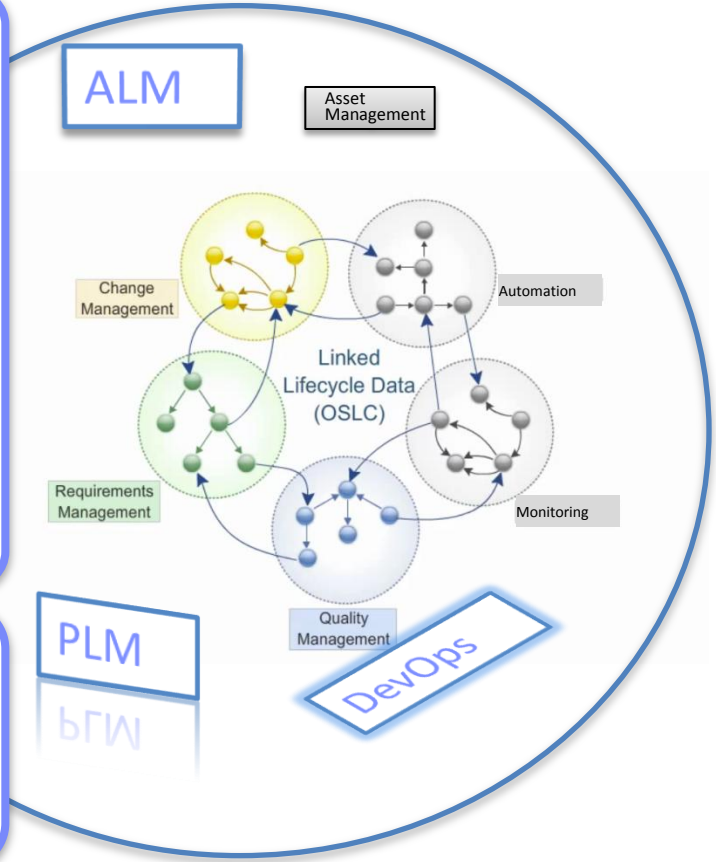
  Tracing, indexing, analyzing the web of lifecycle data where it lives
- Utilizes architecture of the internet
  - All data are resources with URLs
  - Open standards
  - Loosely coupled
  - Technology neutral
  - Scalable, extensible

Open Services for Lifecycle Collaboration (OSLC)

Working to improve the way software lifecycle tools share data

Open Services for Lifecycle Collaboration
Lifecycle integration inspired by the web

- Community driven and governed
  - 400+ registered community members
  - Workgroup members from 34+ organizations
- Wide range of interests, expertise, & participation
- Open specifications for numerous disciplines
- Defined by scenarios – solution oriented
- Implementations from IBM, BPs, and Others
- Based on W3C Linked Data



Inspired by the web **Proven**

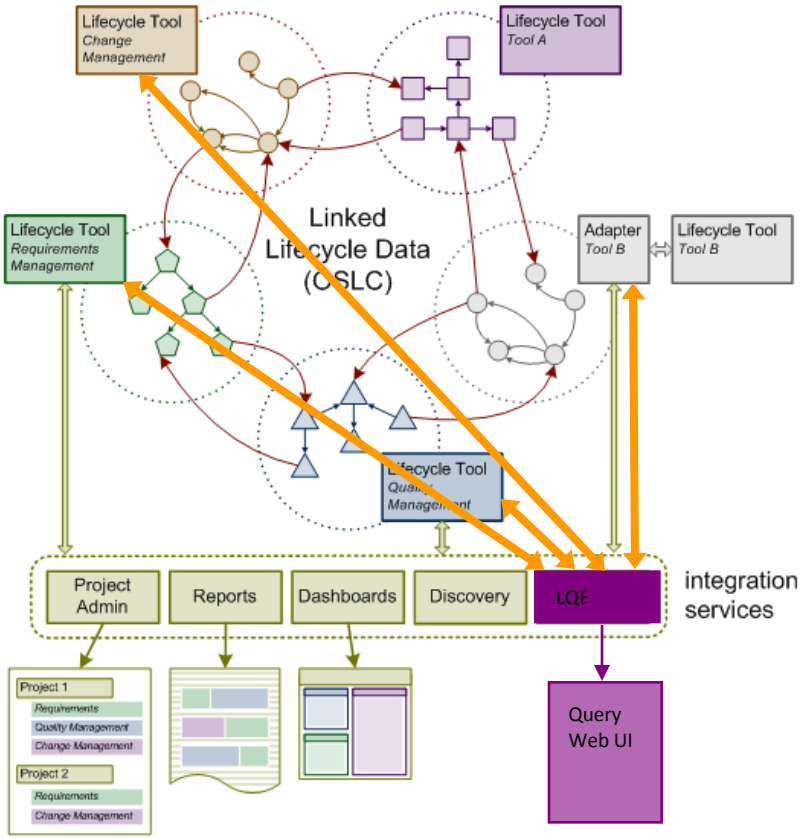
Free to use and share **Open**

Changing the industry **Innovative**

open-services.net

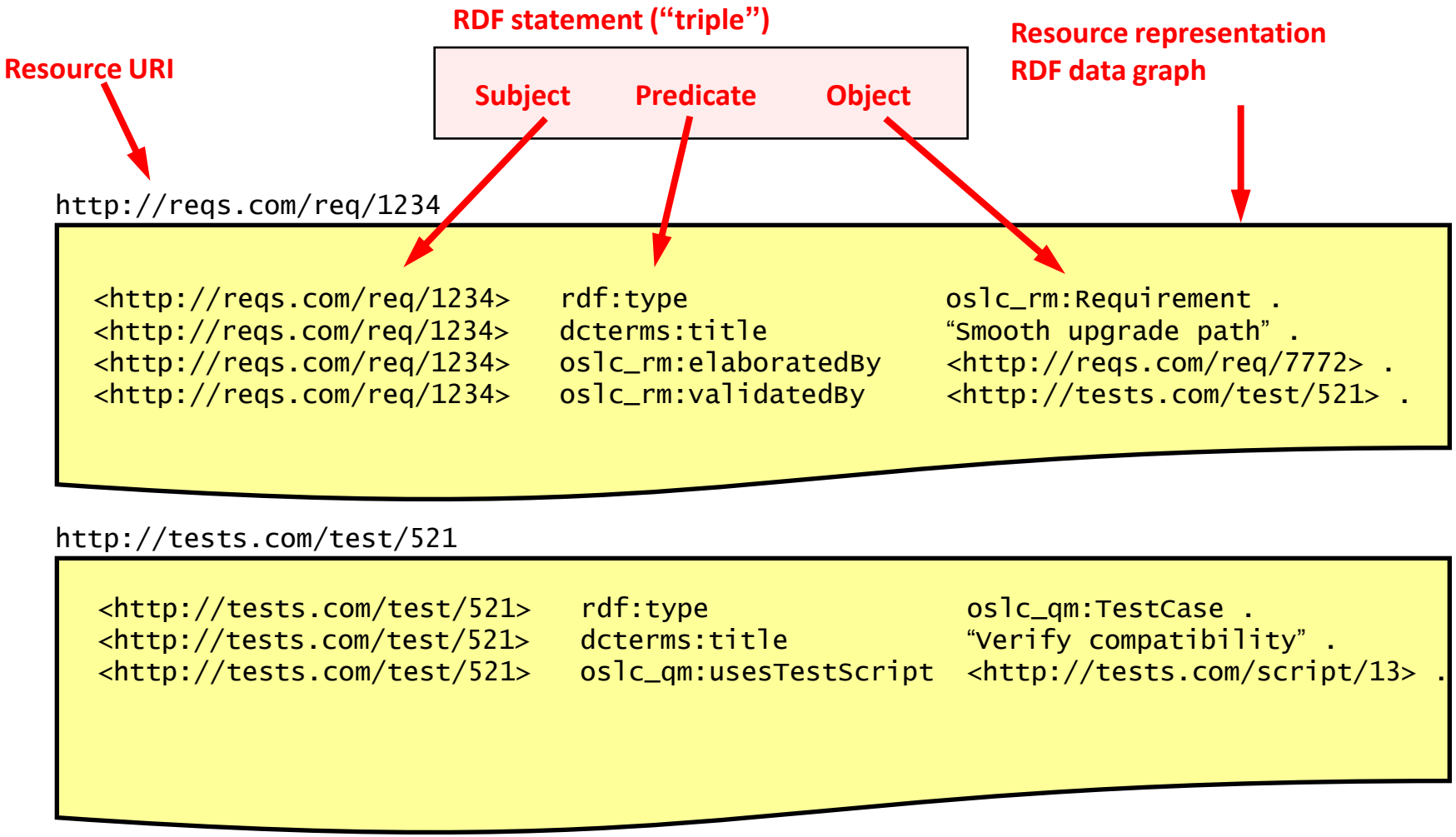
GET INVOLVED AND CONTRIBUTE!

New Integration Service - Lifecycle Query



Provides ability to run queries over linked lifecycle data aggregated from multiple lifecycle tools

Linked Lifecycle Data (LLD)



RDF standard www.w3.org/RDF/

SPARQL Query Language

- SPARQL is standard query language for RDF datasets

SPARQL query

```
SELECT ?x ?title1 ?y ?title2 WHERE {
  ?x   rdf:type             oslc_rm:Requirement .
  ?x   dcterms:title        ?title1 .
  ?x   oslc_rm:validatedBy  ?y .
  ?y   rdf:type             oslc_qm:TestCase .
  ?y   dcterms:title        ?title2 . }
```

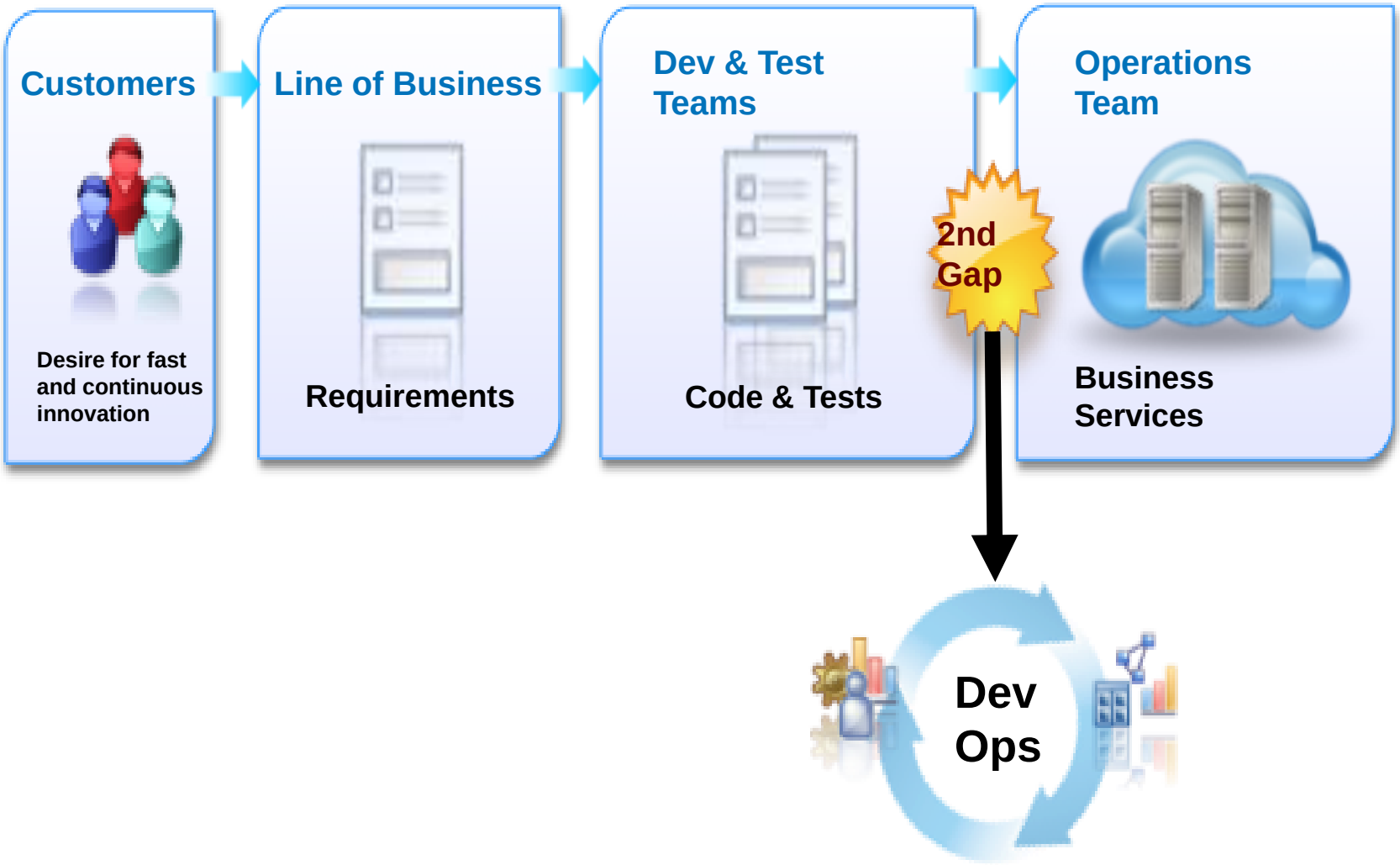
Query results

x	title1	y	title2
<http://reqs.com/req/1234>	"Smooth upgrade path"	<http://tests.com/test/521>	"Verify compatibility"

- Queries can mine linked lifecycle data aggregated from multiple lifecycle tools

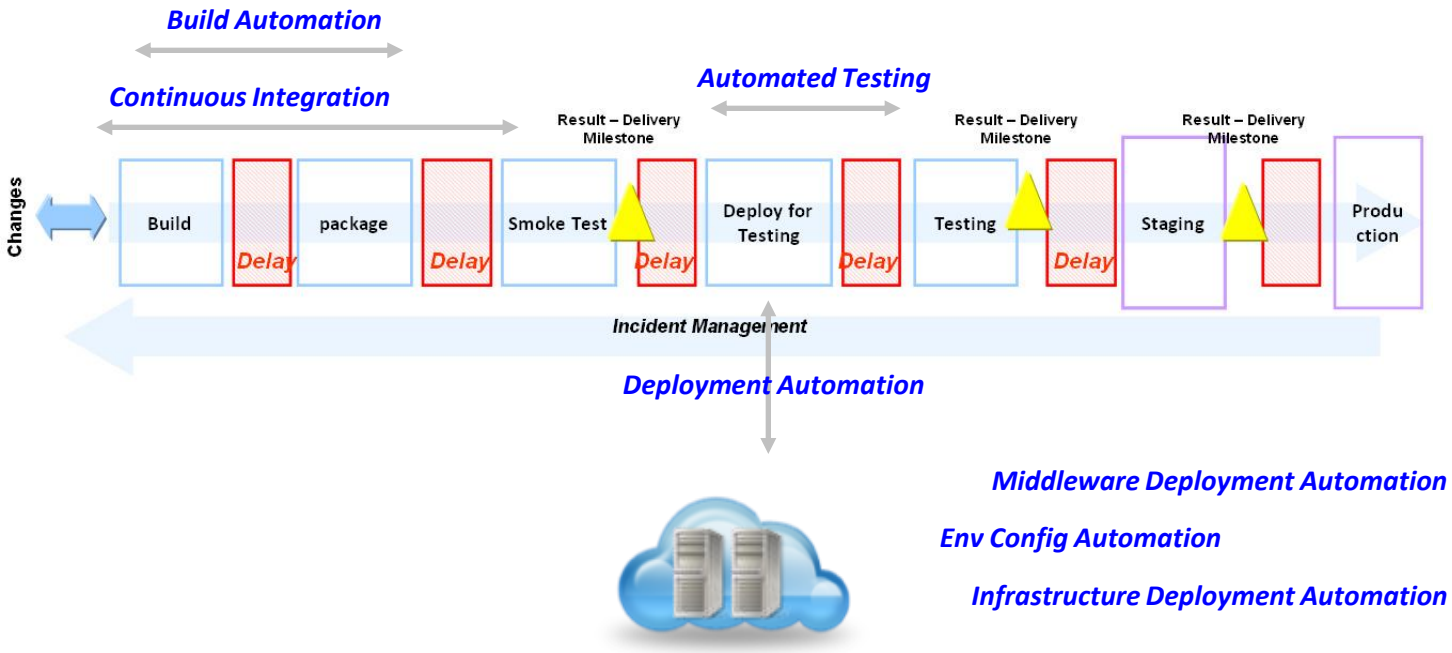
SPARQL standard www.w3.org/TR/rdf-sparql-query/

Addressing Application Lifecycle Management gaps

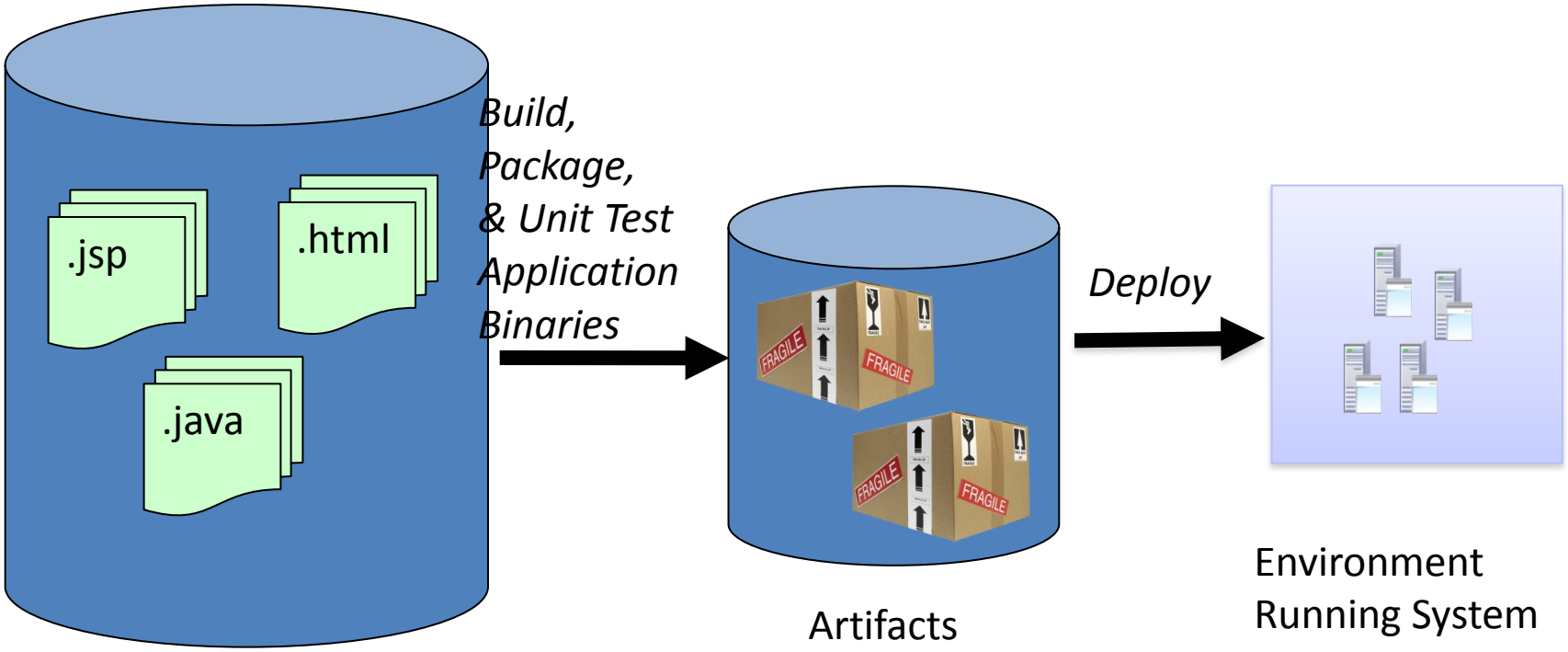


Current Customer approaches addressing these challenges..

- Selective & Siloed automation of the delivery process with limited benefits
- Poor visibility and control impacting cycle time

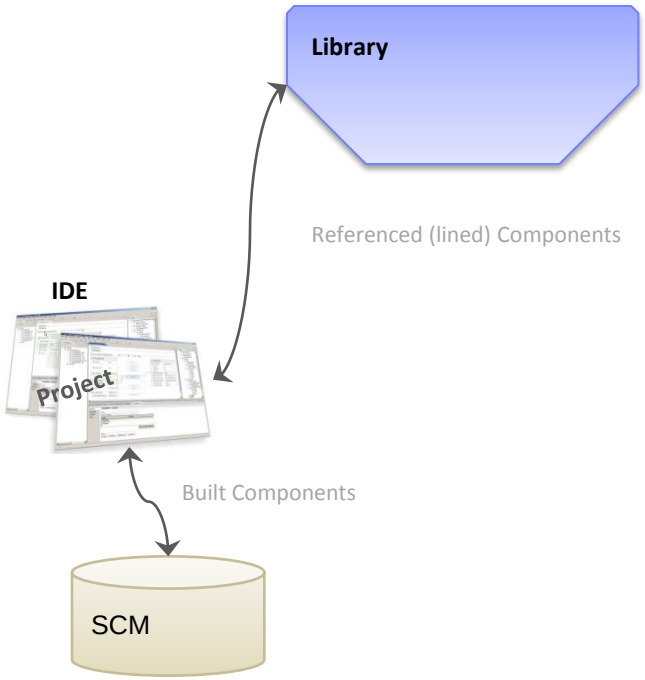


Automating development hand off today



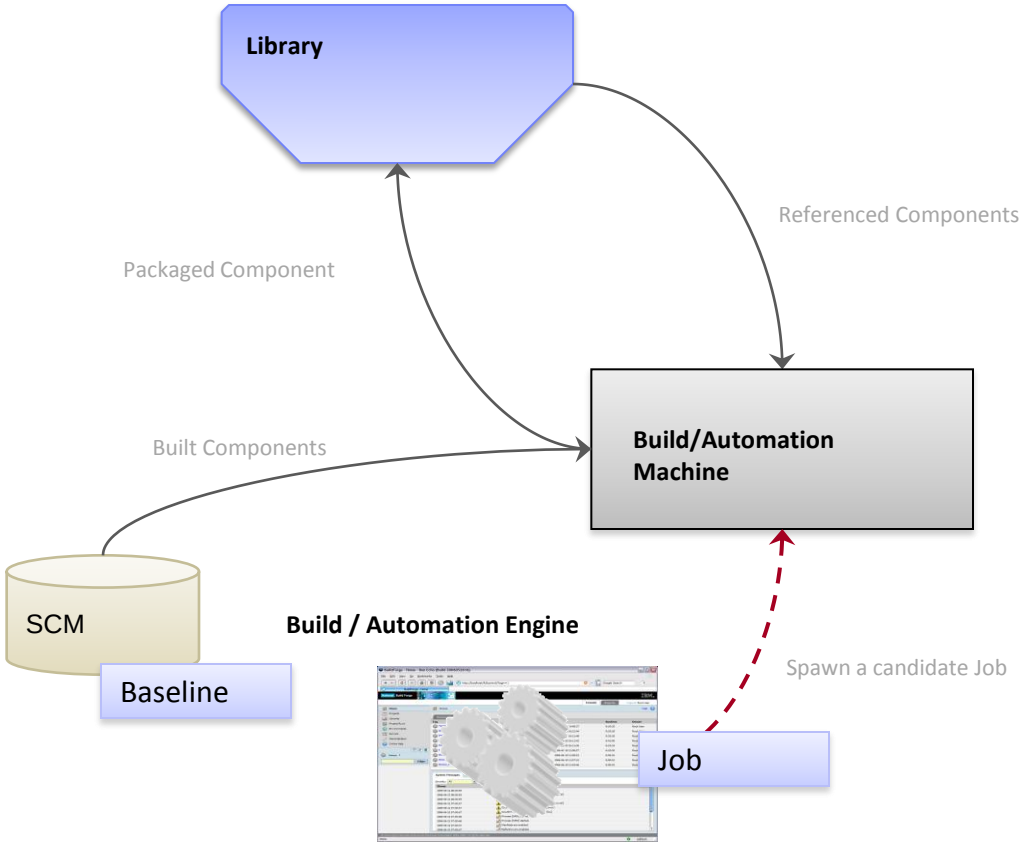
Development phase

1. Use SCM / CM to manage the things you build.
2. Use a Library to link with the thing other have built.



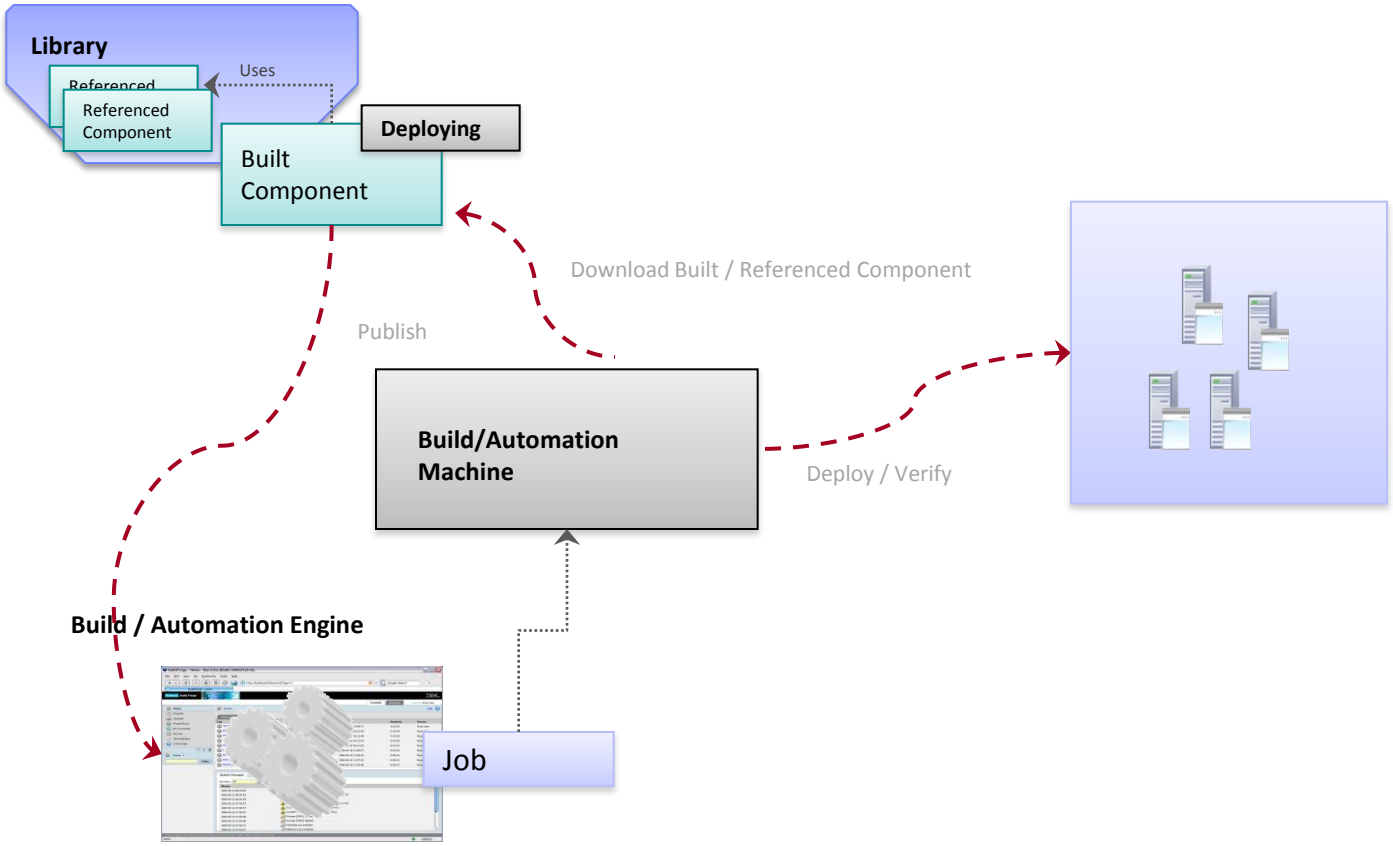
Build / Automation Phase

1. Track the Bill of Materials used in a build
2. Manage which build move onto the next stage

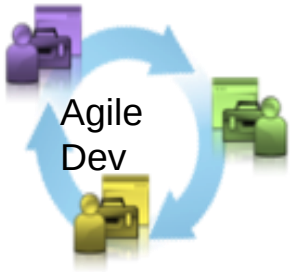


Deploy Automation Phase

1. Control what is deployed
2. Track the Bill of Materials used to deploy



With only Agile Development improvements…



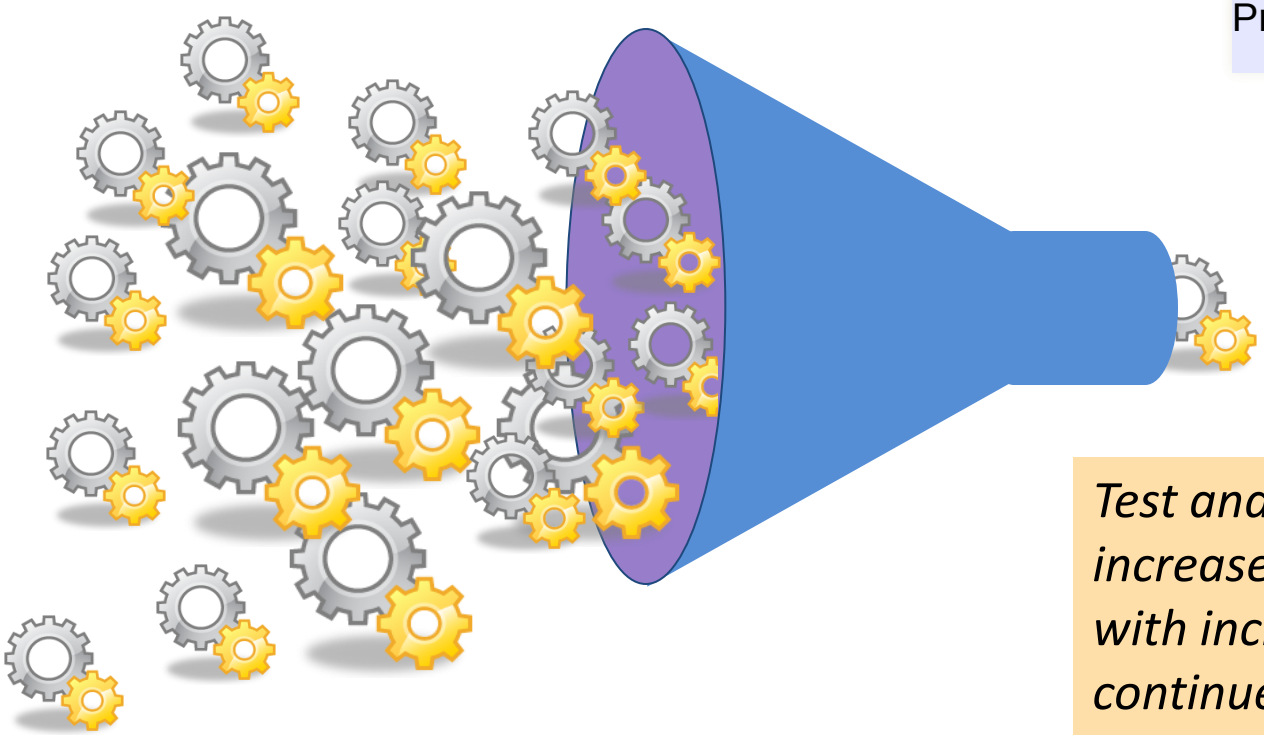
Functional Testing

Acceptance Testing

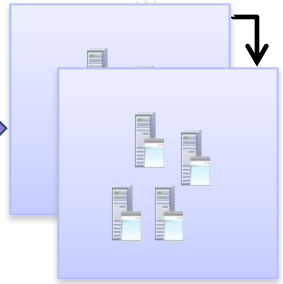
Production

Operator

Setup (weeks)



Install



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

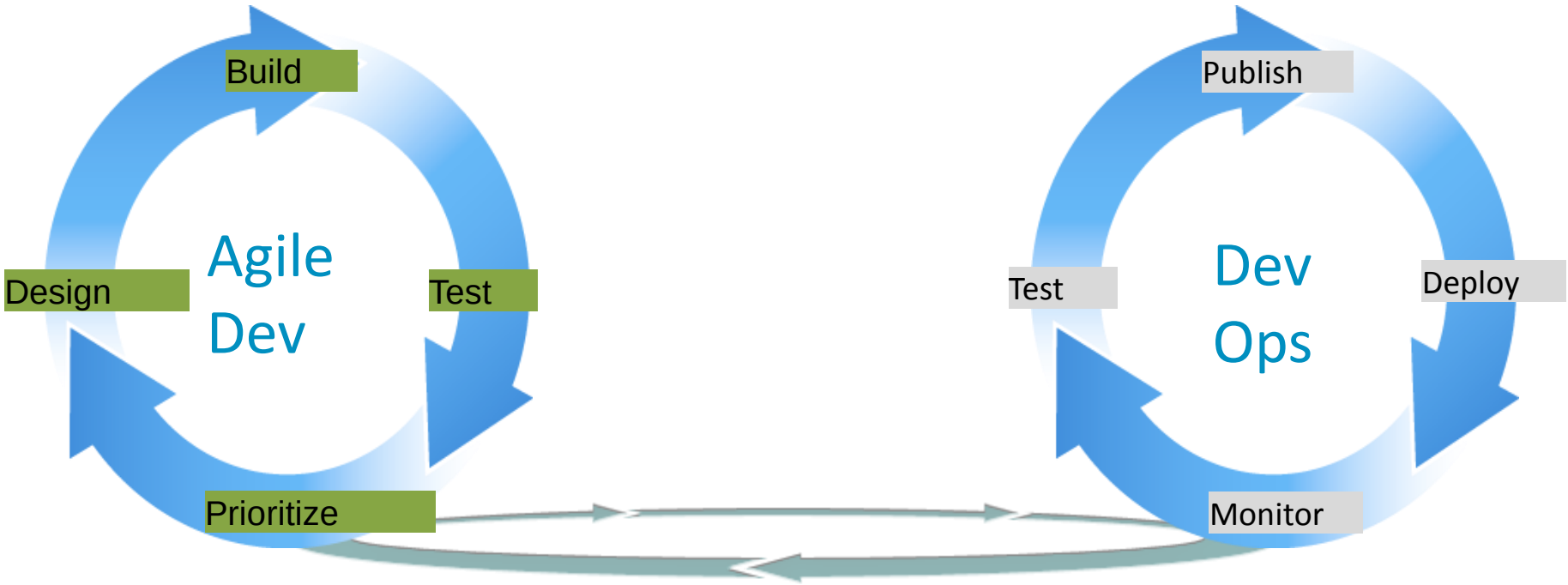
CI builds are piling up

Adjusting Delivery Mindset

- Infrastructure Developer vs. Operator/Administrator
- Need to bring a software development mindset to the operational areas
- Replicate, where appropriate, standard architecture/development tools and methodologies
- Use an Agile approach to delivery of routines
- Continuous, incremental improvements and delivery of new functionality
- Automated unit and integration testing improves operational runtimes
- Source Control Management
- Automation routines and scripts are fundamental to Operations
- Managing Operations routines like source code offers several benefits:
  - Central point of truth as routines and environments change
  - Backup in case of loss
  - Identify possible regressions by comparing with prior versions
- Example Managed Assets:
- Perl, Jython, WSADMIN, ANT scripts, Service orchestration routines (opsware, b
  Infrastructure Gold copies components

Agile Development and Delivery

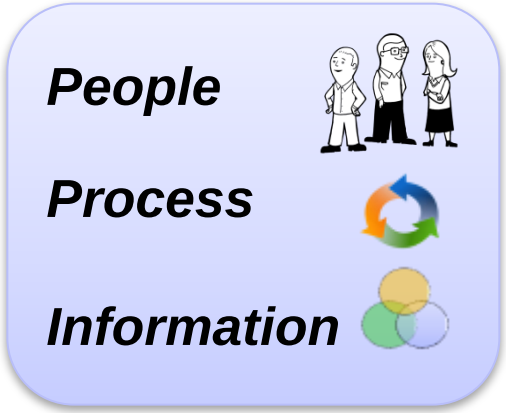
Continuous Integration extends to Continuous Delivery



Continuous Feedback

DevOps: Tighter alignment between Development & Operations to increase application velocity with managed risk

DevOps Principles & Values



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

12 Principles for Better DevOps*

Collaborate

1. Do your developers and operators communicate the production realities and the application's requirements?
2. Do you have patterns for platforms and applications, designed jointly by development and operations?
3. Do you have a delivery pipeline exposed through a summary dashboard to assess delivery velocity?

Automate

4. Are your patterns based on reusable deployment configuration scripts?
5. Can you deploy an environment (platform and application) in one step?
6. Can your developers launch and destroy production-like environments from those patterns?

*Based on "The Joel Test: 12 Steps to Better Code" http://www.joelonsoftware.com/articles/fog0000000043.html

12 Principles for Better DevOps

Validate

7.Do you have automated tests to validate your application and platform function and characteristics?

8.Do you monitor software against expectations after deploying your application?

9.Do you deploy your applications daily into production-like environments and verify them?

Manage and Control

10.Do you version deployment configuration and scripts along with your source code?

11.Do you link bugs and work items to changes in the application and configuration?

12.Do you associate tickets for production issues with relevant bugs opened for development to fix?

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.

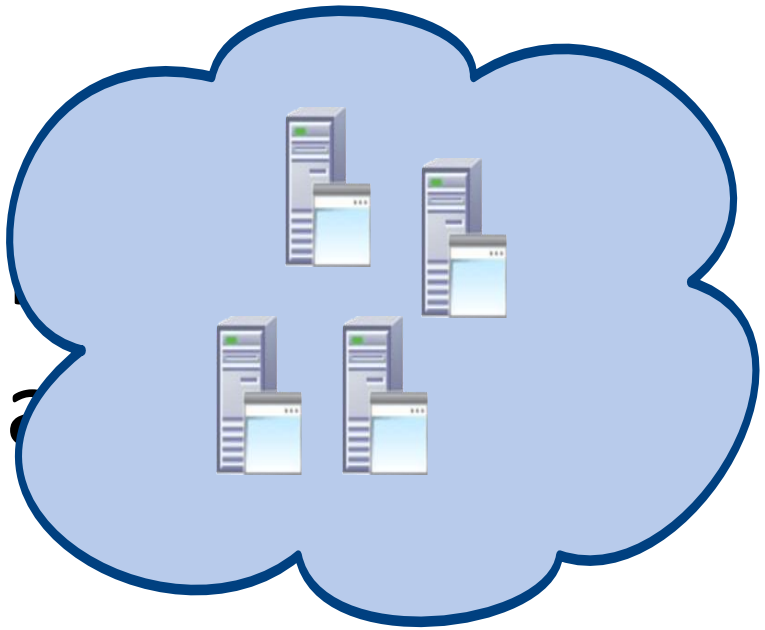
Python

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,

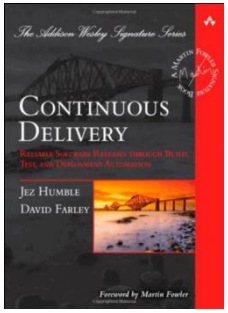
3. vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

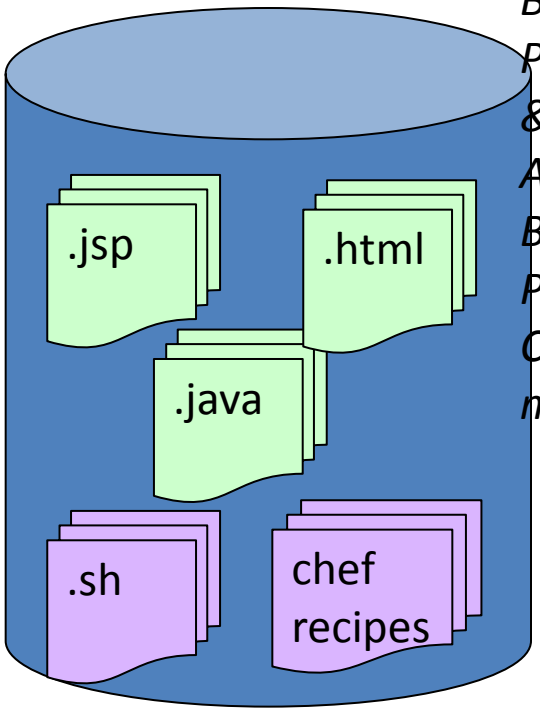
```
#!/usr/bin/env ruby

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

# ...
```



De

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

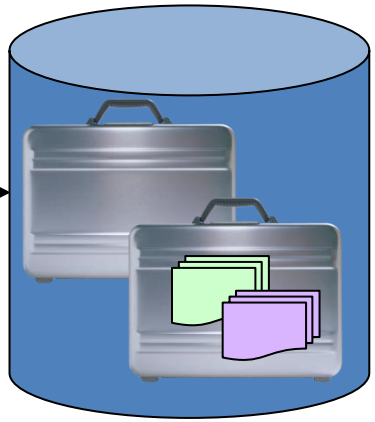




Source Artifacts

Source Control Management

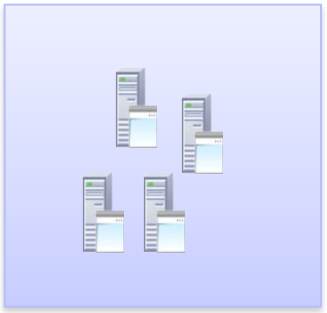
Build, Package, & Unit Test Application Binaries & Platform Configuration



Deployable Artifacts

Library

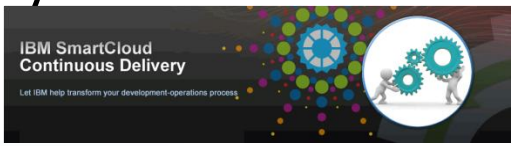
Deploy

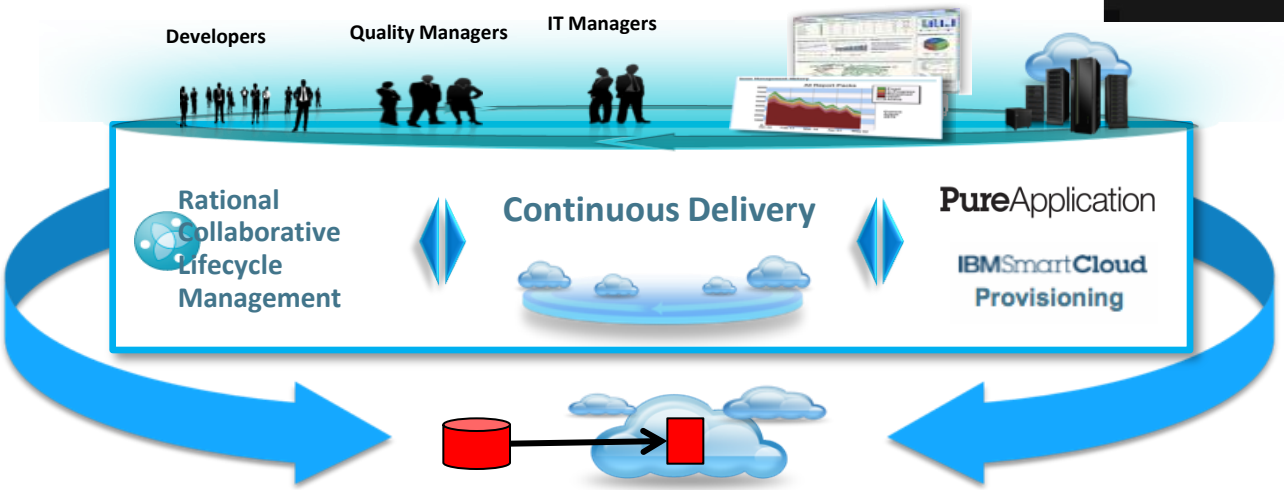


Environment Running System

IBM SmartCloud Continuous Delivery

Extending Agile disciplines through delivery





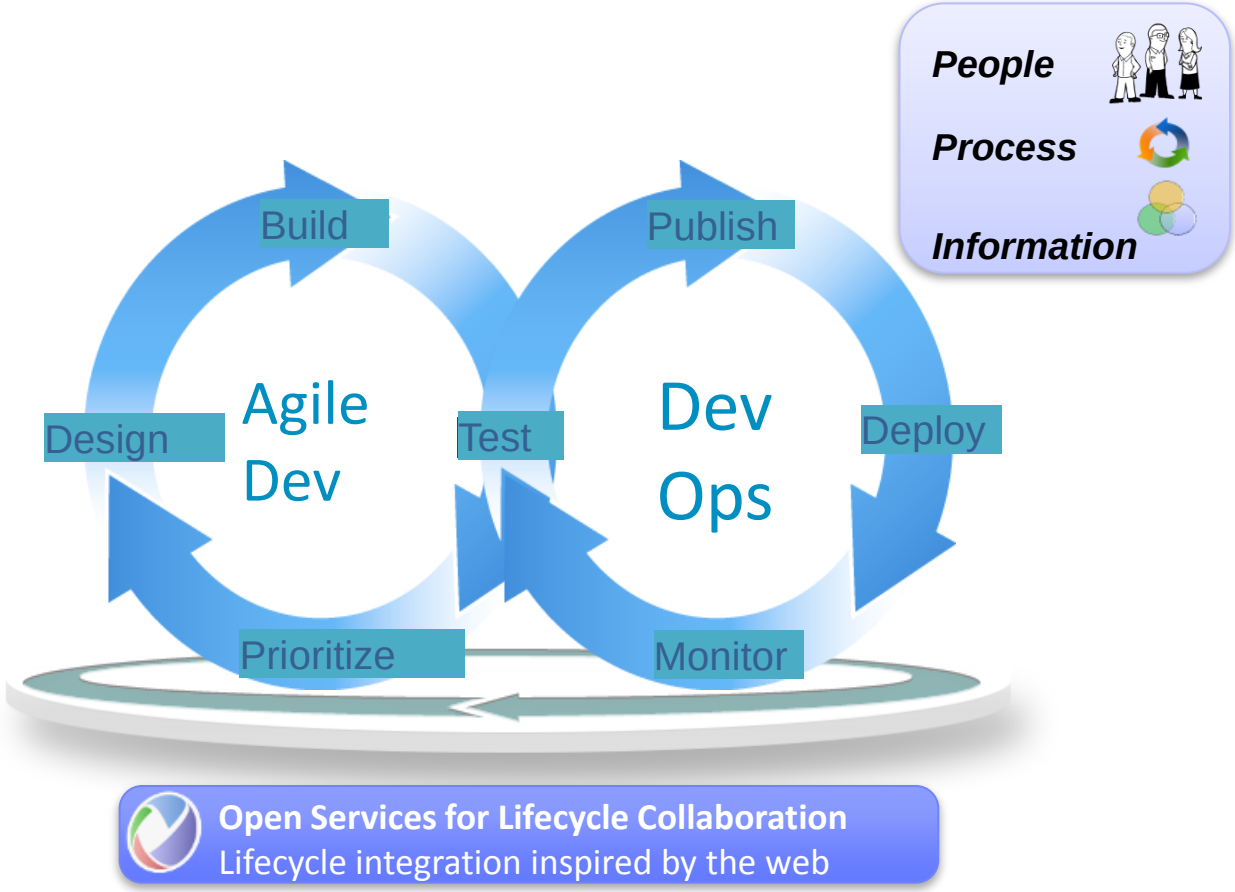
Client Value

- 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

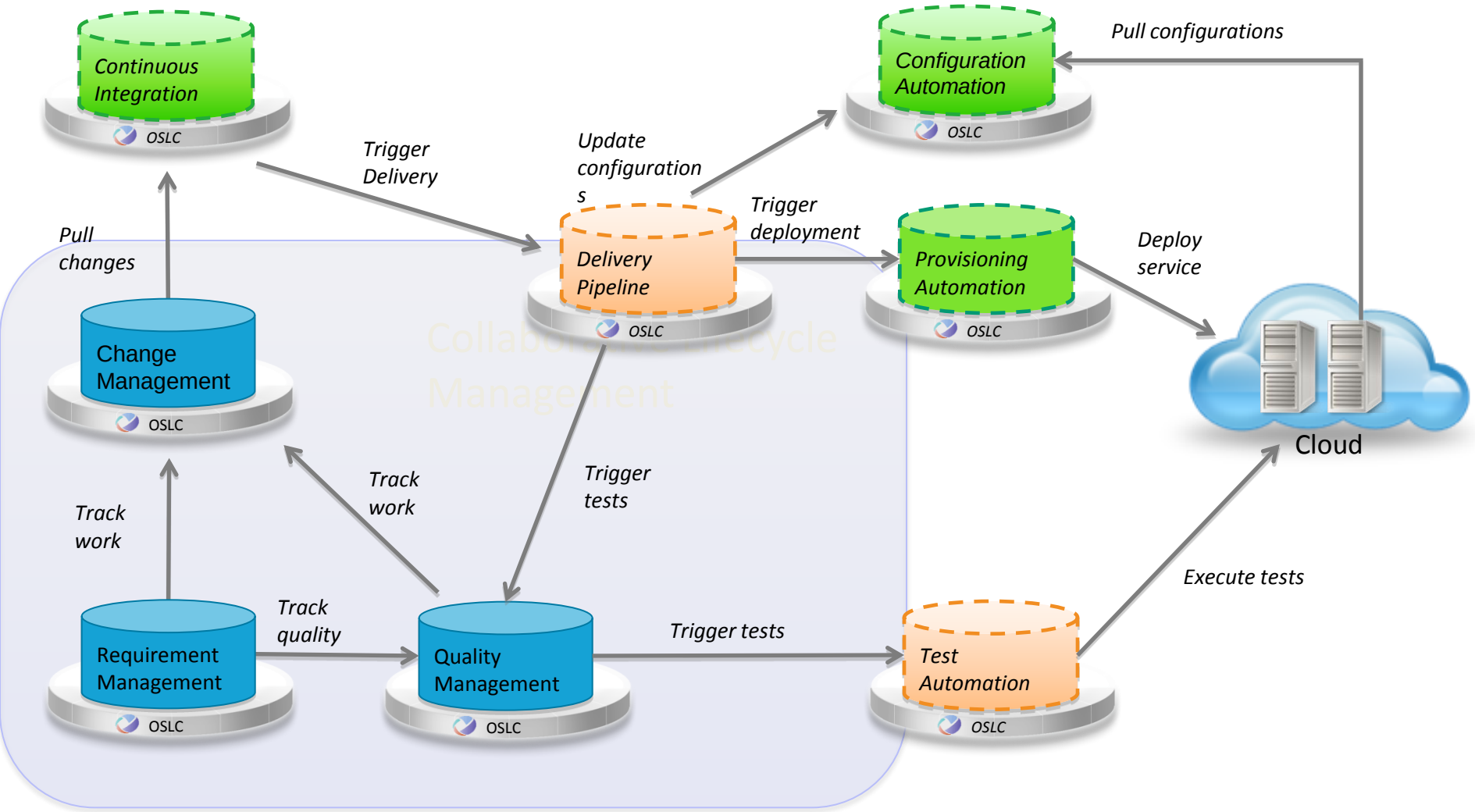
- Development team extending Agile into rapid workload deployment in the cloud
- Operation teams delivering scalable, continuous delivery services to the development organization

End-to-End Lifecycle Optimization



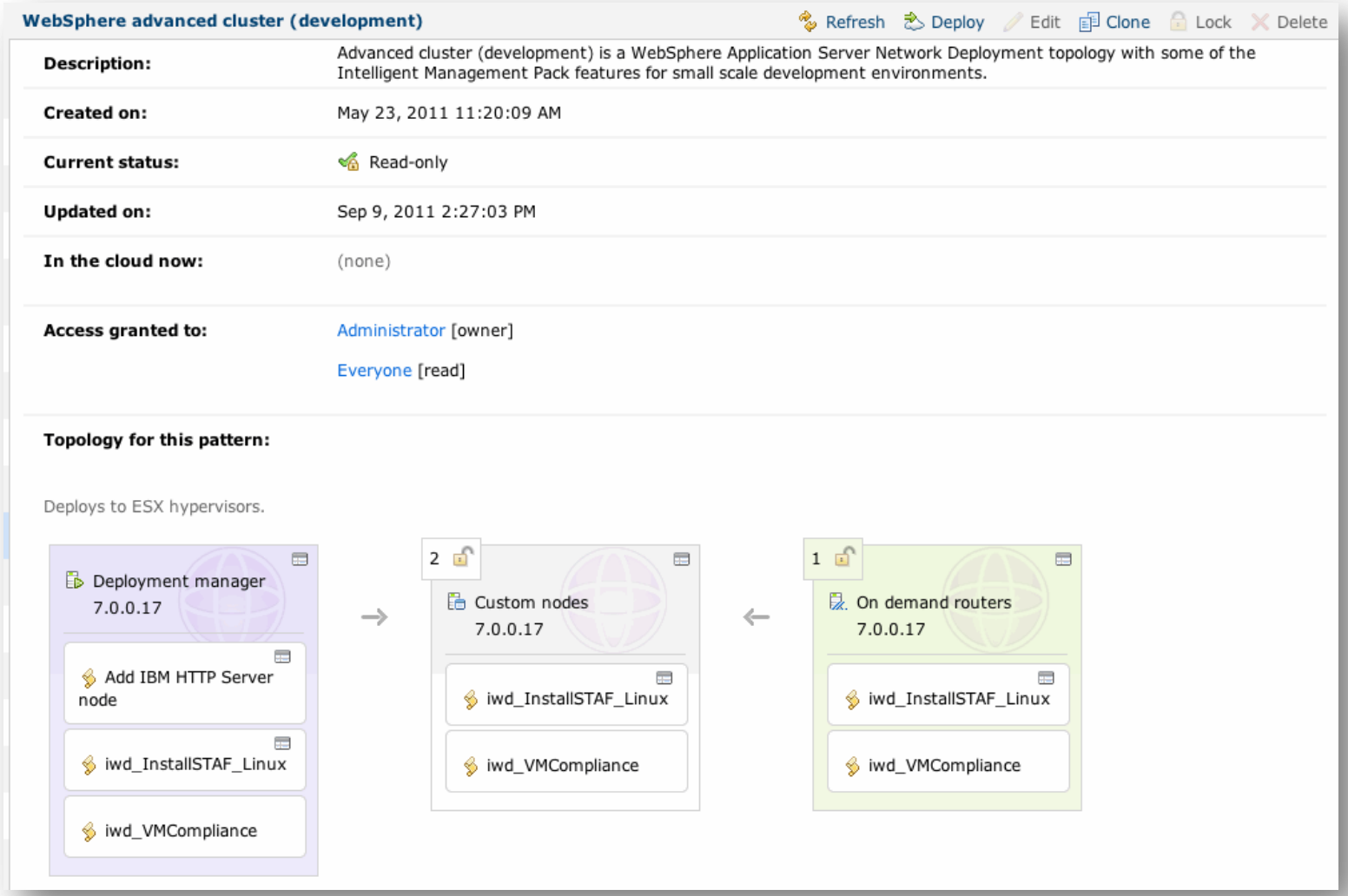
Follow us on the Enterprise DevOps and Jazz Team Blogs

Lifecycle Management Reference Architecture

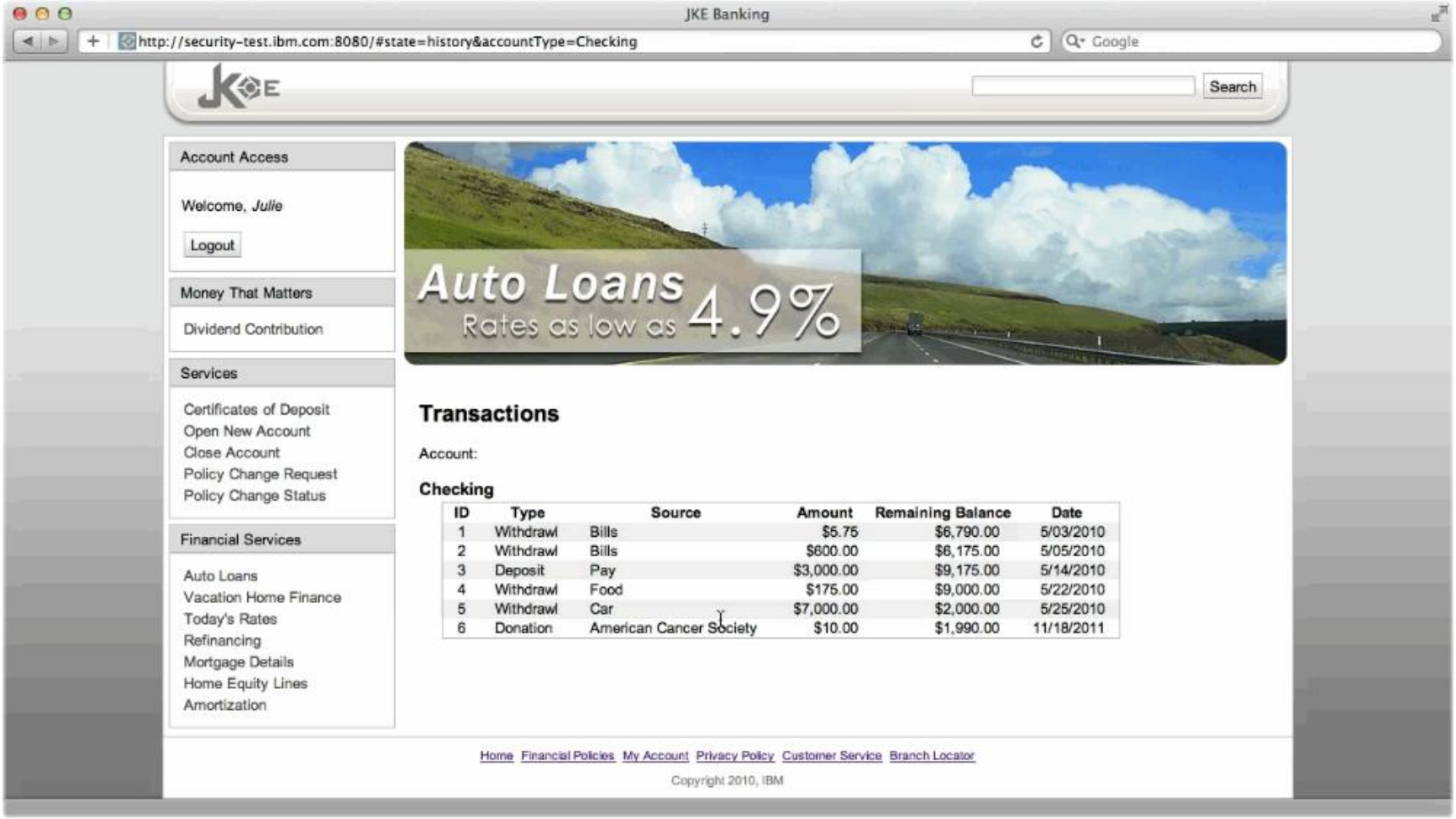


Adopt in any order, at any time

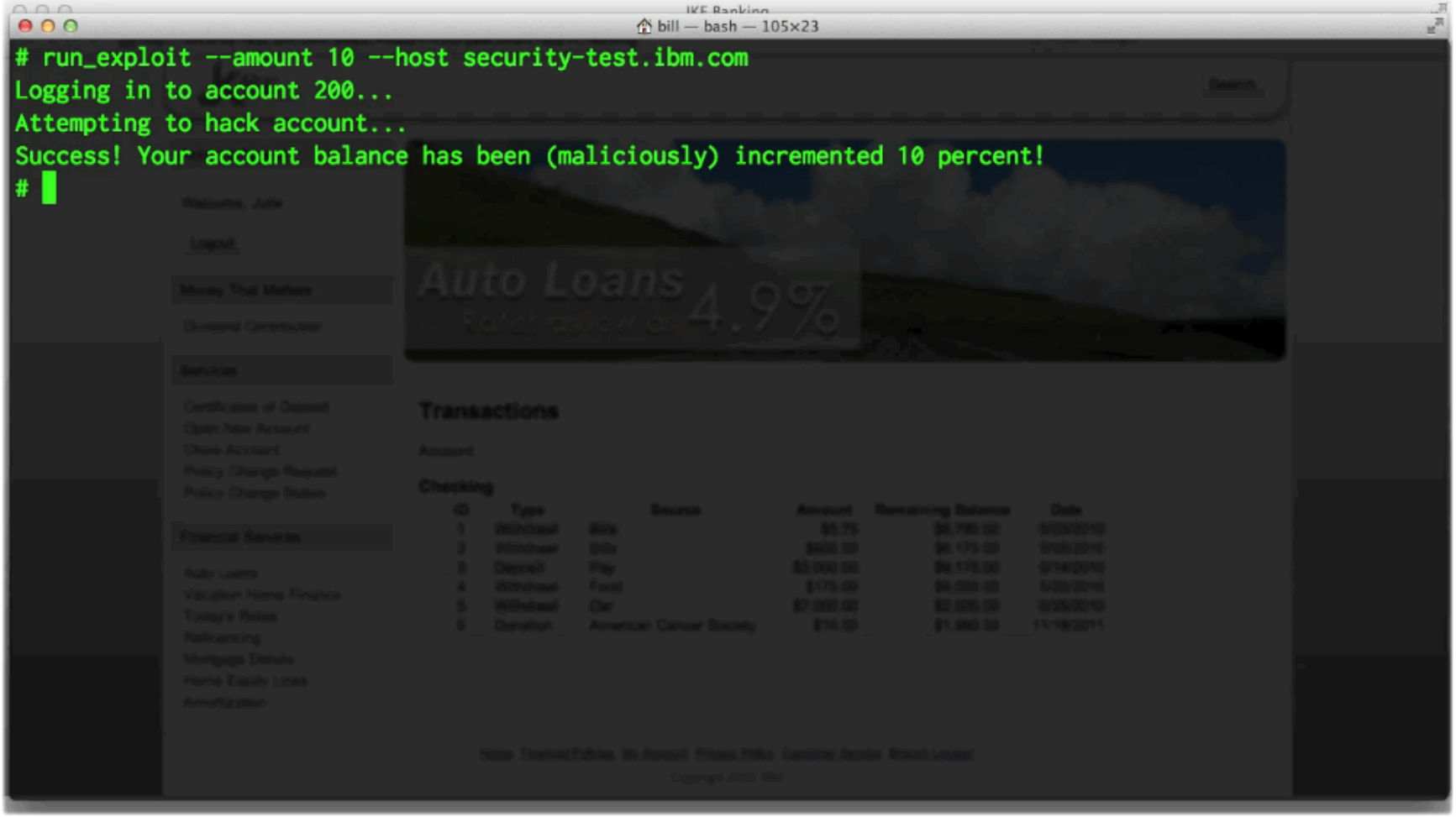
Creating standard workload patterns



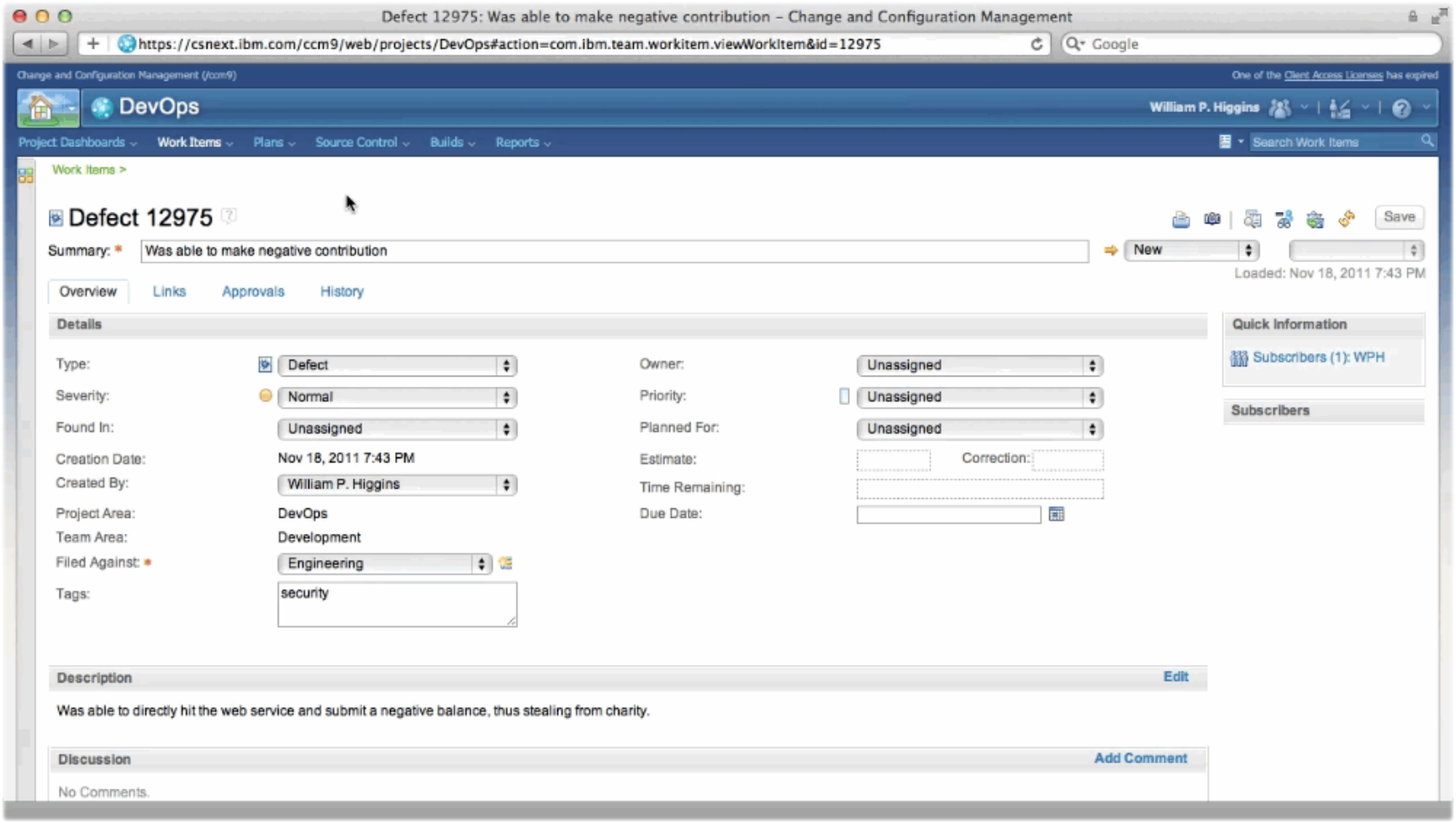
Demo: Application



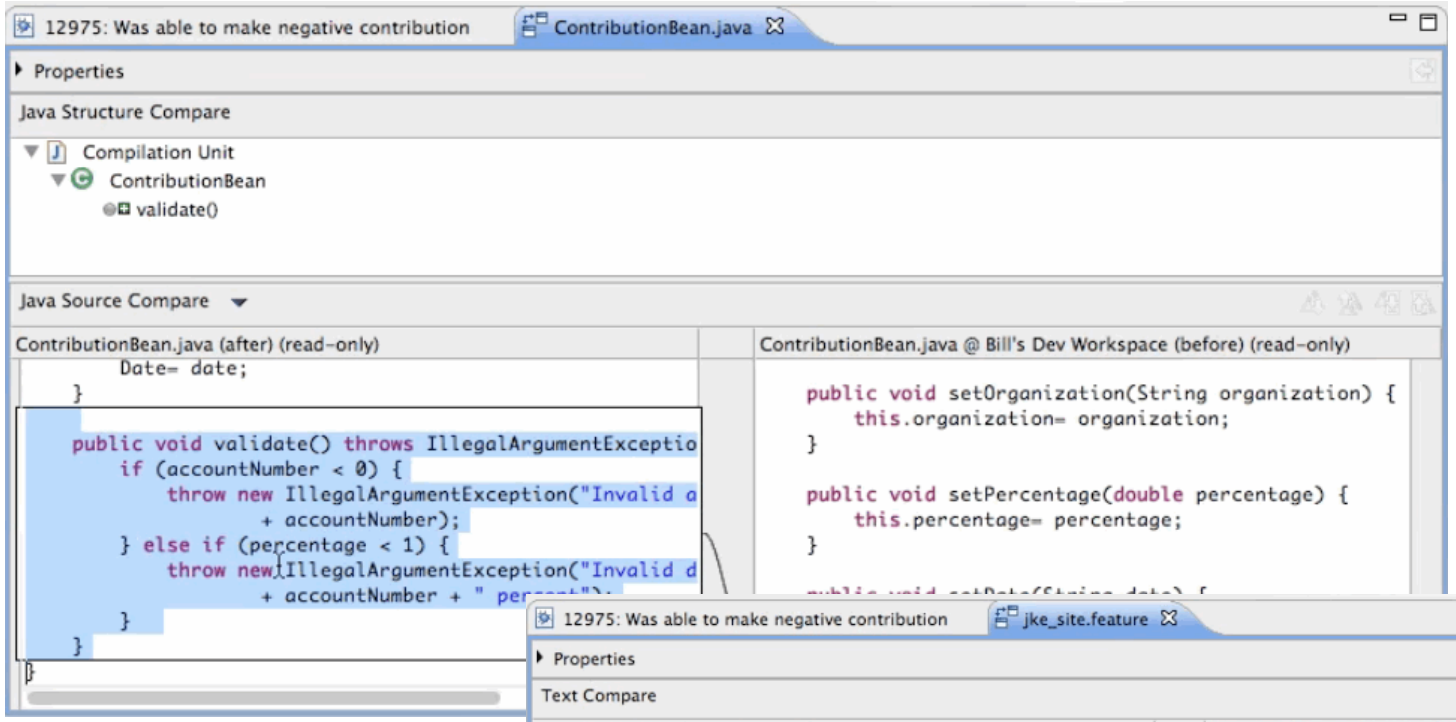
Demo: Security Test



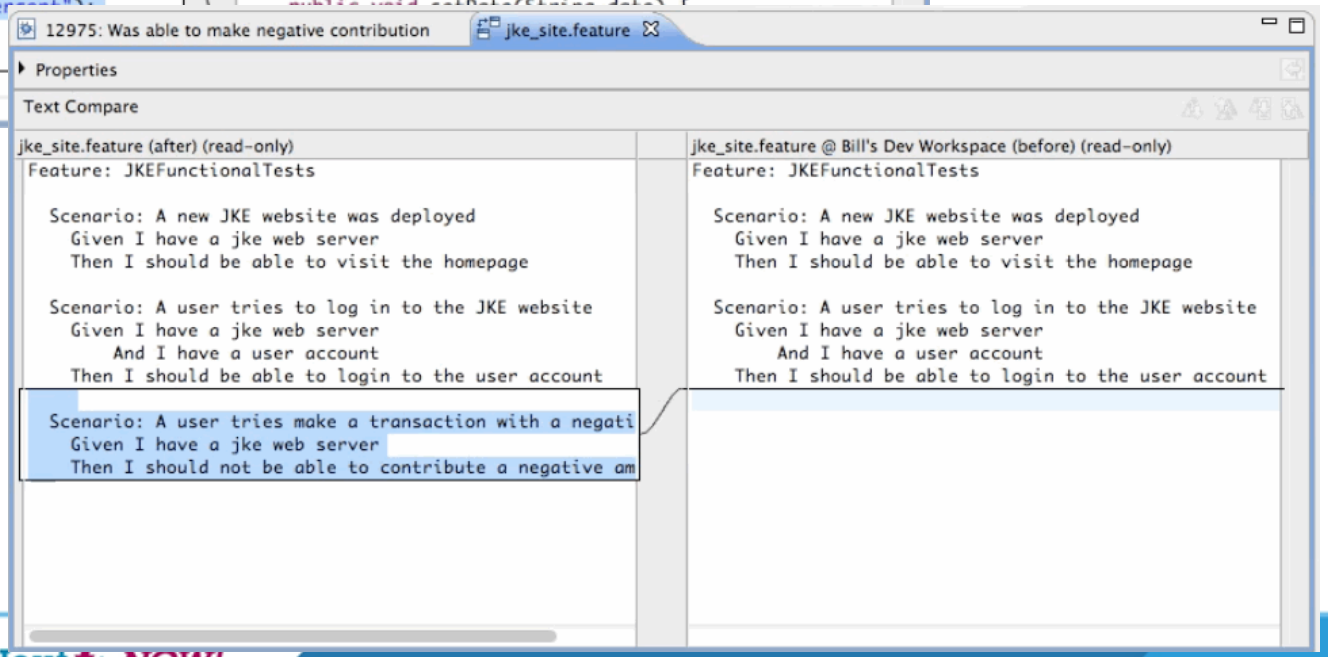
Demo: Defect created



Demo: Fix the defect

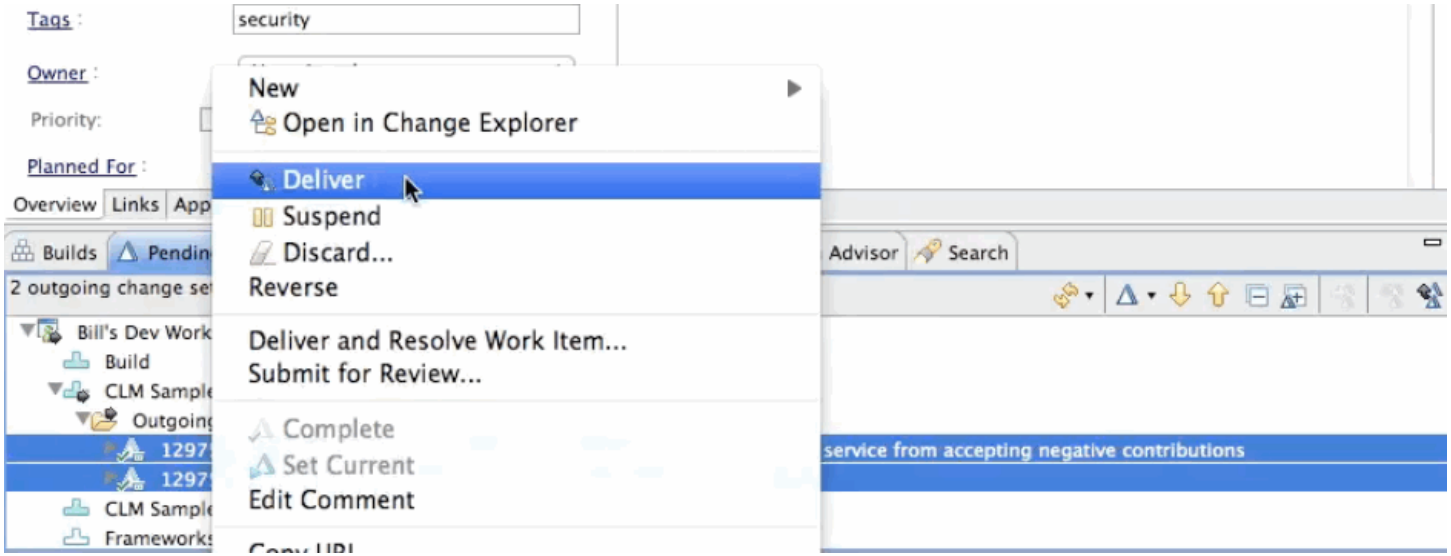


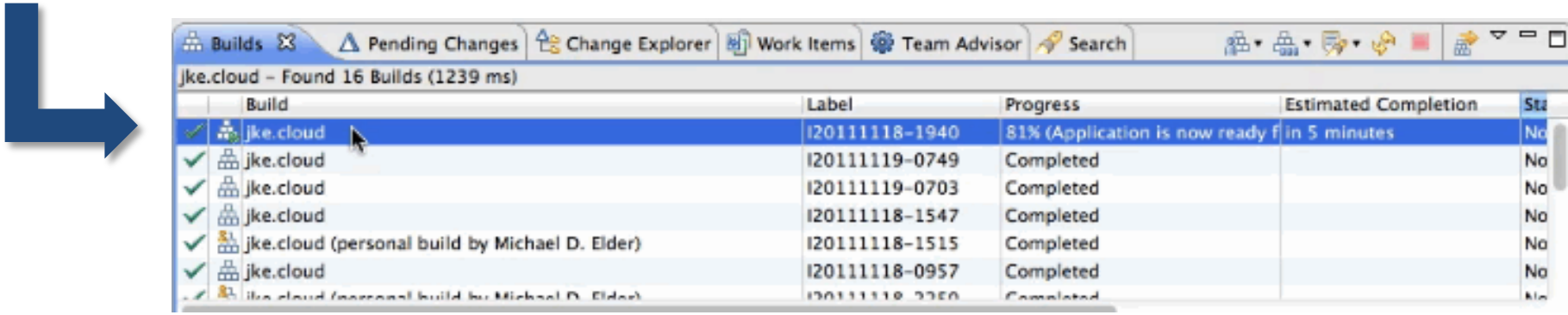
Fix



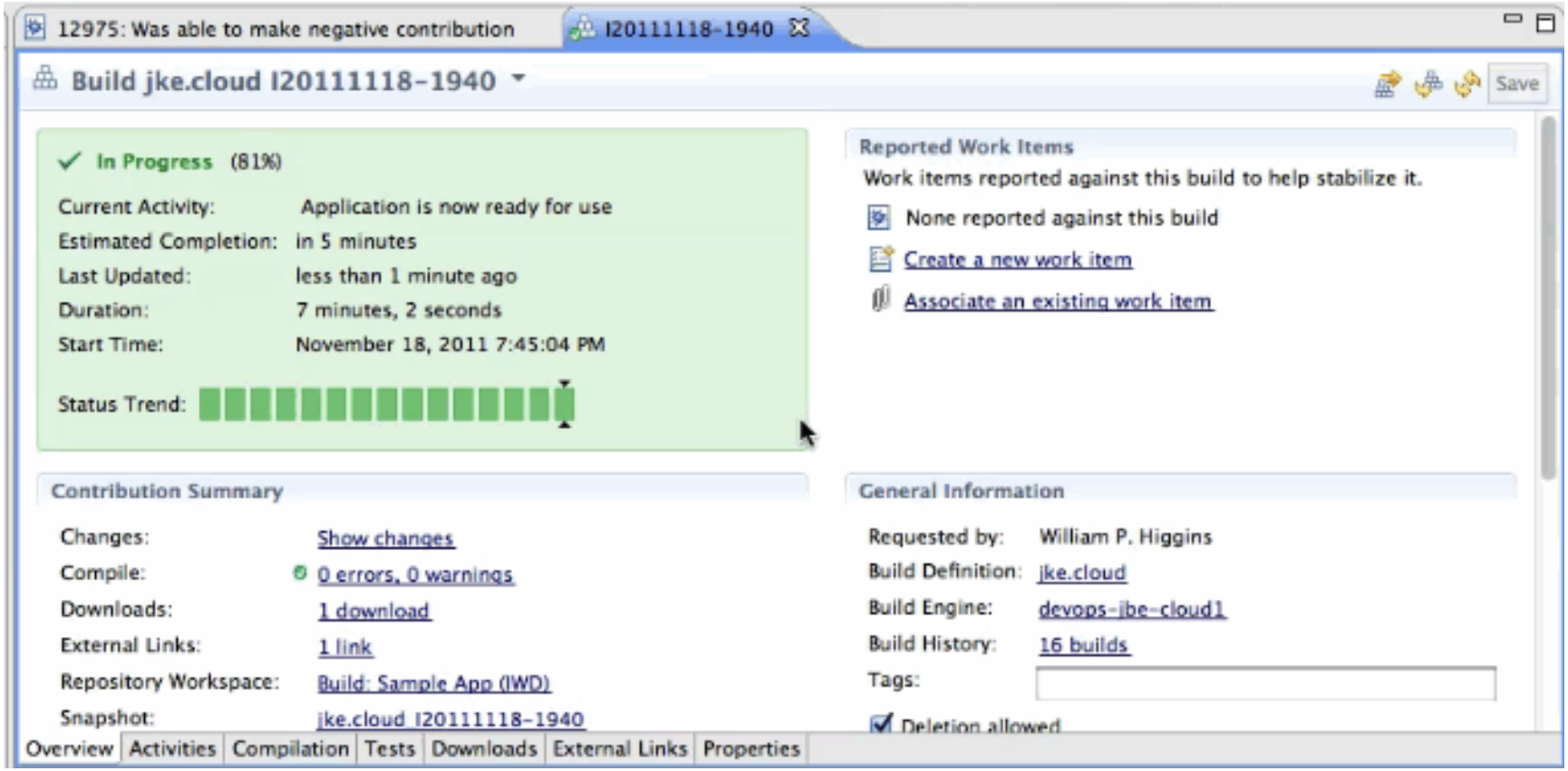
New Test

Demo: Deliver the fix

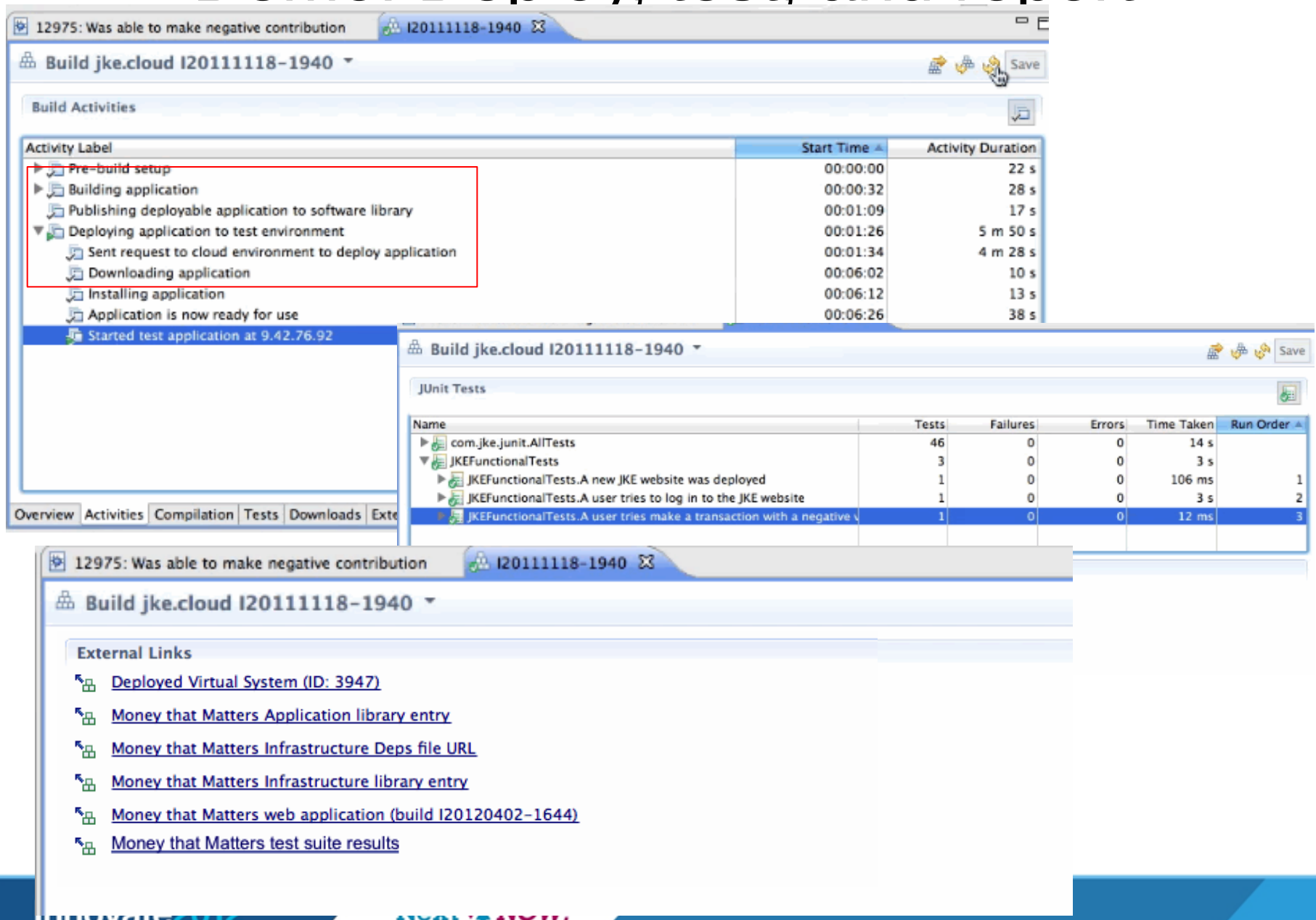




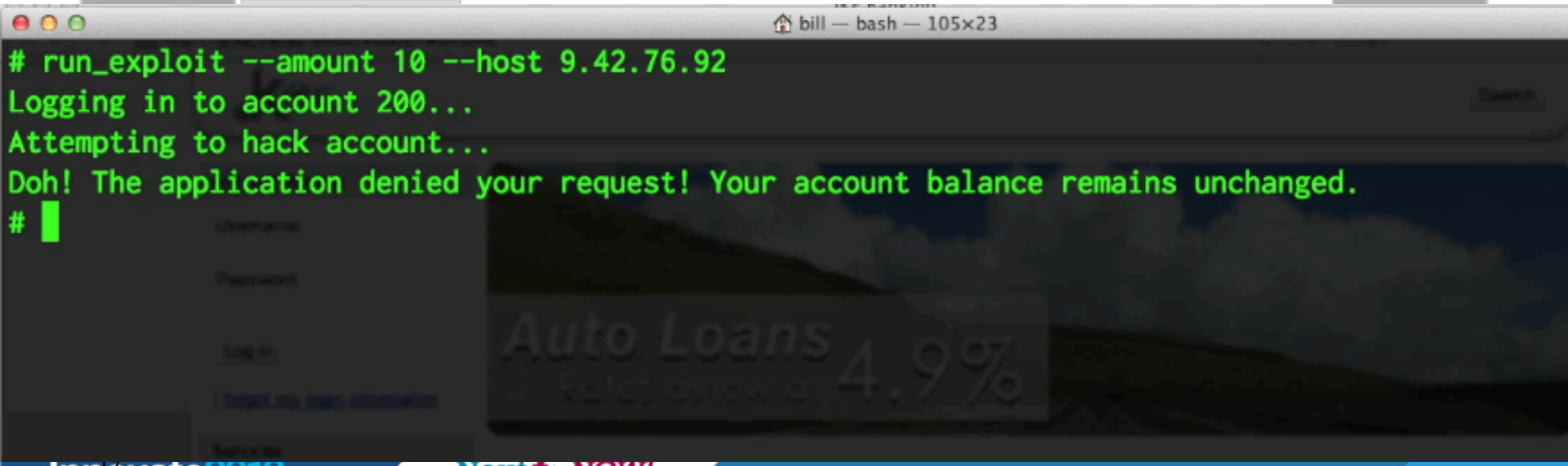
Demo: Build starts



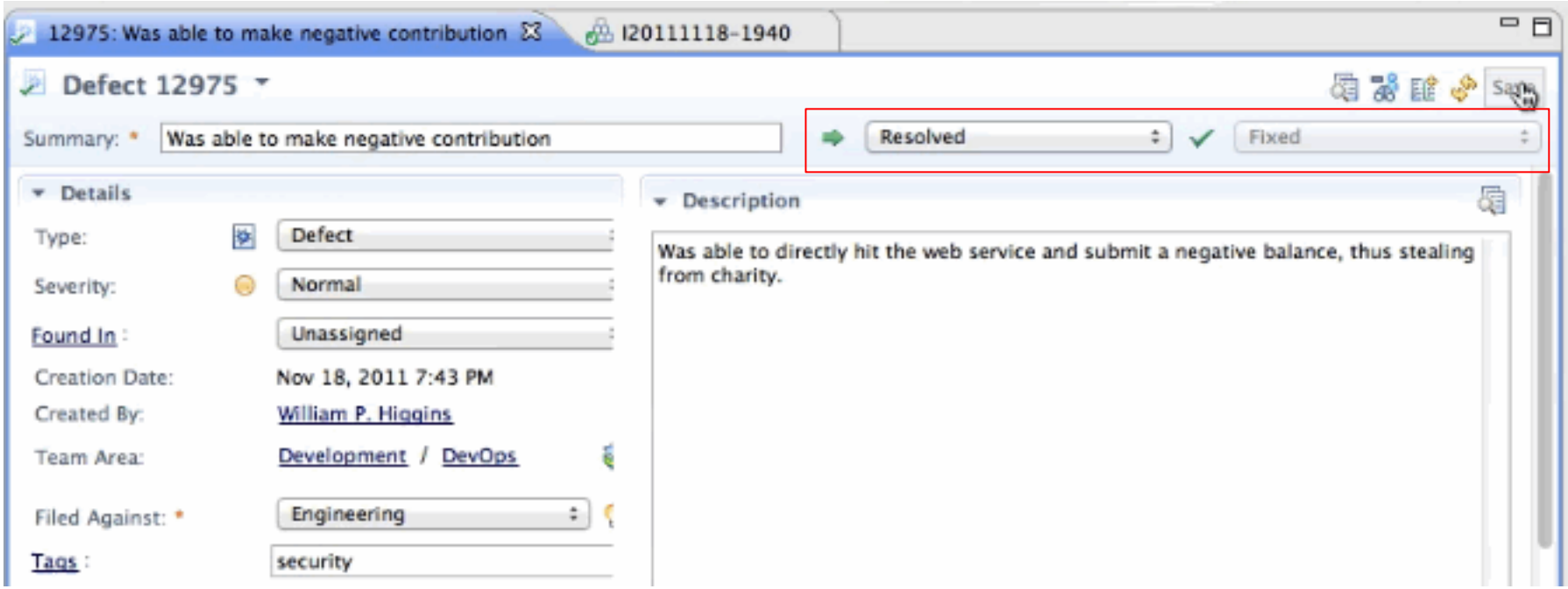
Demo: Deploy, test, and report



Demo: Test with fix



Demo: Resolve the defect



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.

© Copyright IBM Corporation 2012. All rights reserved.

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

IBM, the IBM logo, ibm.com, Rational, the Rational logo, Telelogic, the Telelogic logo, Green Hat, the Green Hat logo, and other IBM products and services are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml

If you have mentioned trademarks that are not from IBM, please update and add the following lines:

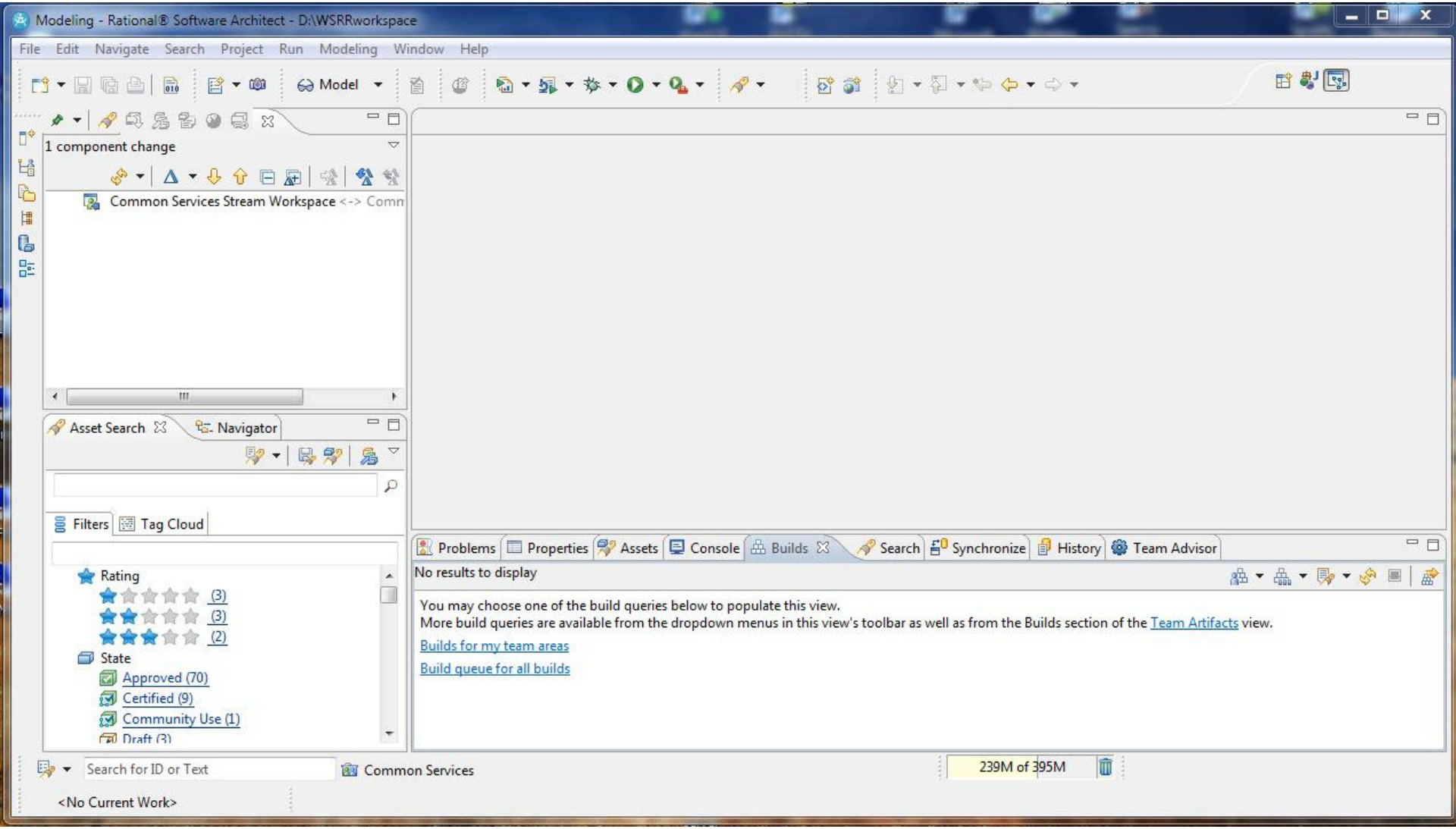
[Insert any special third-party trademark names/attributions here]

Other company, product, or service names may be trademarks or service marks of others.

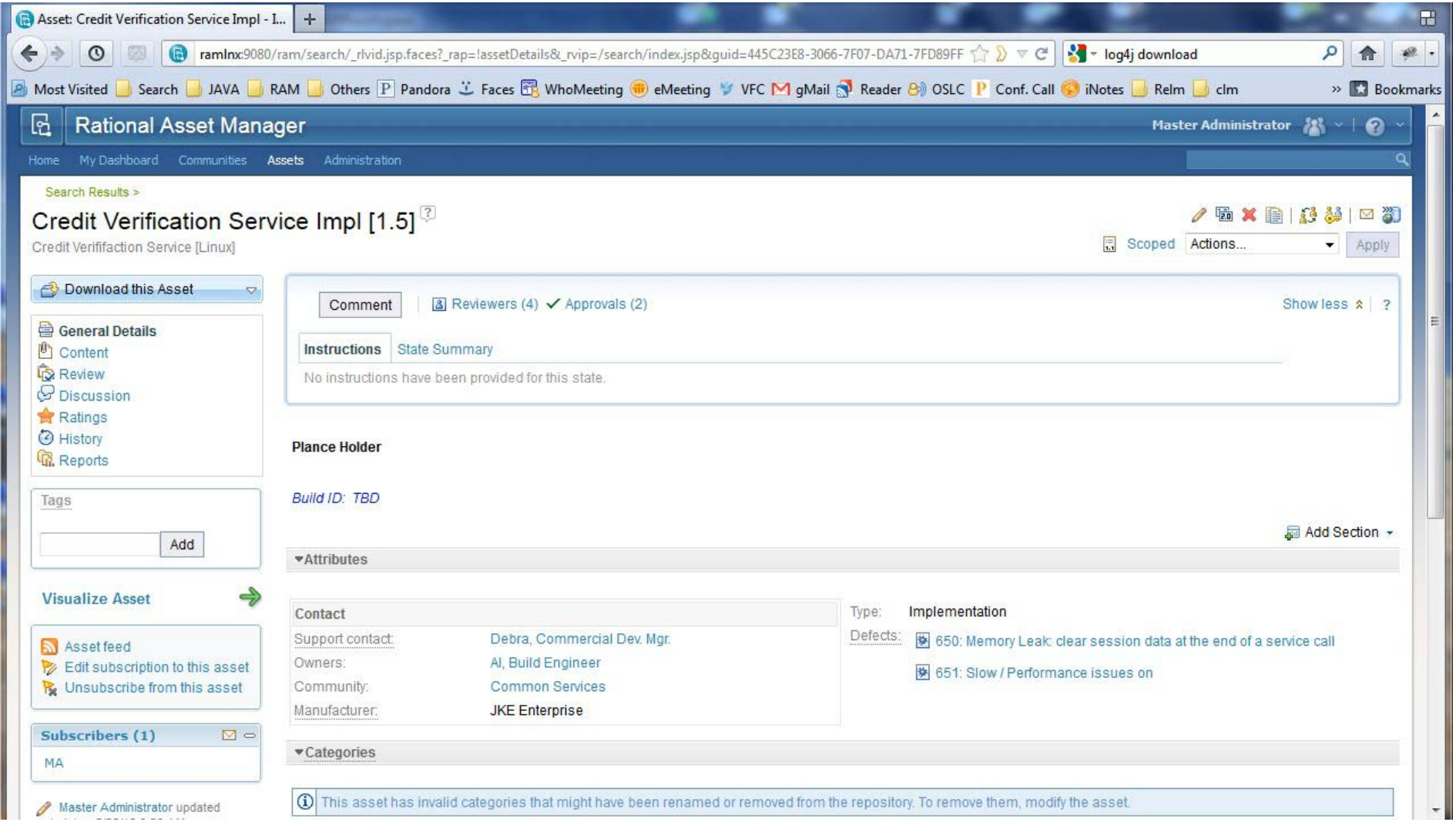
www.ibm.com/software/rational

© Copyright IBM Corporation 2012. All rights reserved. The information contained in these materials is provided for informational purposes only, and 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, these materials. Nothing contained in these materials 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. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

Development phase



Build Deploy automation



BACKUP

IMPORTANT: Directions for presentation templates

- IBM speakers, if you make any reference to product/services content that won't be announced until after Innovate 2012 concludes, you must include the disclaimer language
(on slide 3 here) verbatim, the disclaimer language must be on the second slide in your presentation and is in addition to the Acknowledgements and Disclaimers slide which is to be included in all presentations (slide 9 here).

- IBM speakers, you must include the Acknowledgements and Disclaimers slide at the end of your presentation (slide 9 here). Please see the notes in **red** text and update appropriately for your presentation. If you are speaking with a customer or business partner, you MUST use the disclaimer slide (on slide 3 here).

www.ibm.com/software/rational

www.ibm.com/software/rational

How to use this template

- To allow all masters of your presentation to be updated correctly, download this template to your hard drive and copy your existing slides into the new template
- See slide notes for further formatting instructions

What companies want to achieve

1. Communication of Knowledge and Integration of Pec
2. Better Process
3. Reality-based Measurements

What companies encounter instead

1. Distracted by day-to-day delivery pressures – 78%
2. Tools don't integrate properly – 62%
3. Lack the necessary internal expertise – 56%

Source: Forrester study commissioned by Wipro, 2008

Demo

- Download and configure DevOps CLI
- Generate continuous delivery project with content
- Develop infrastructure as code
- Update continuous delivery properties
- Update and customize delivery process file
- Configure RTC build definition for continuous delivery
- Request build and delivery

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

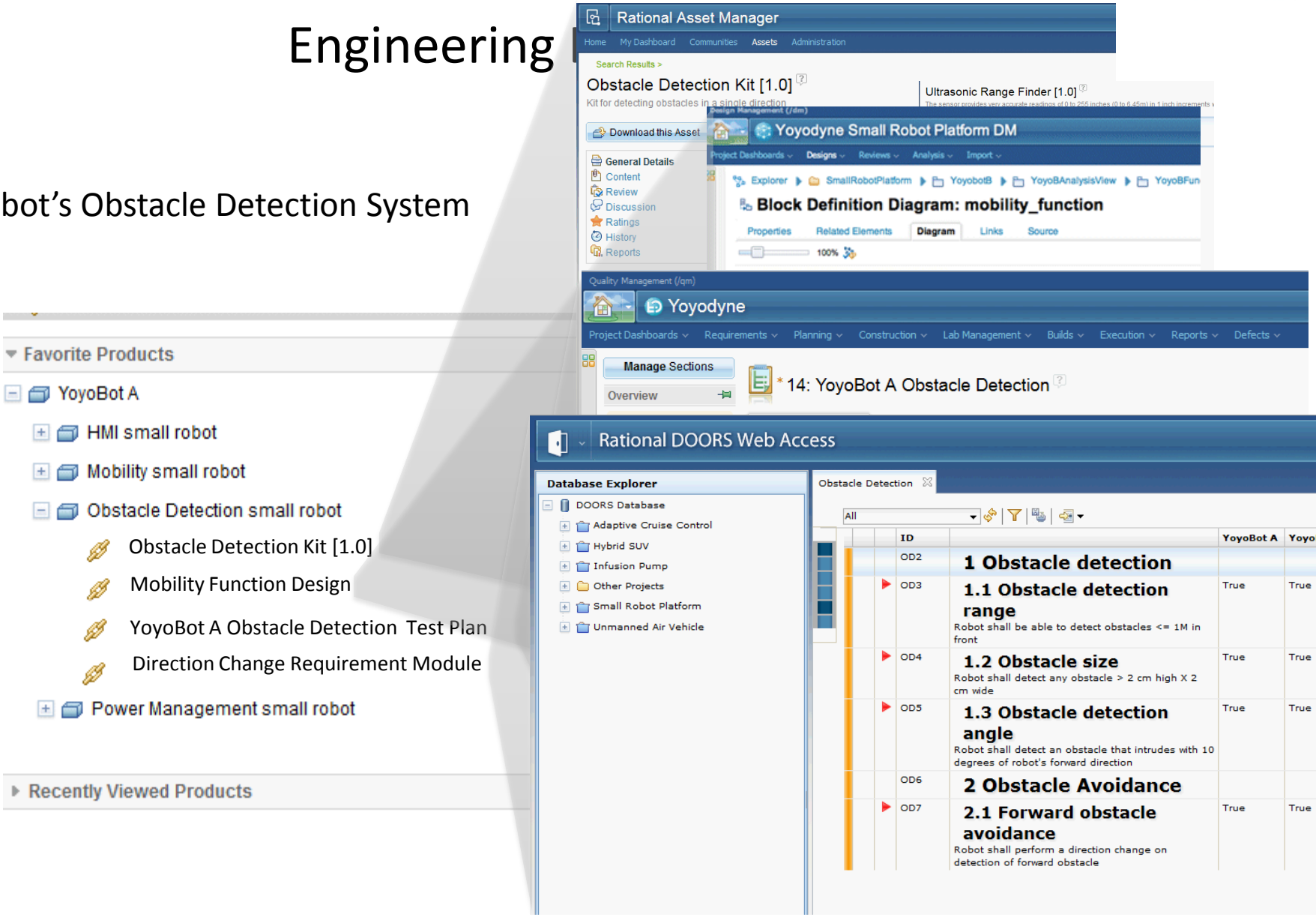
1. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur?

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

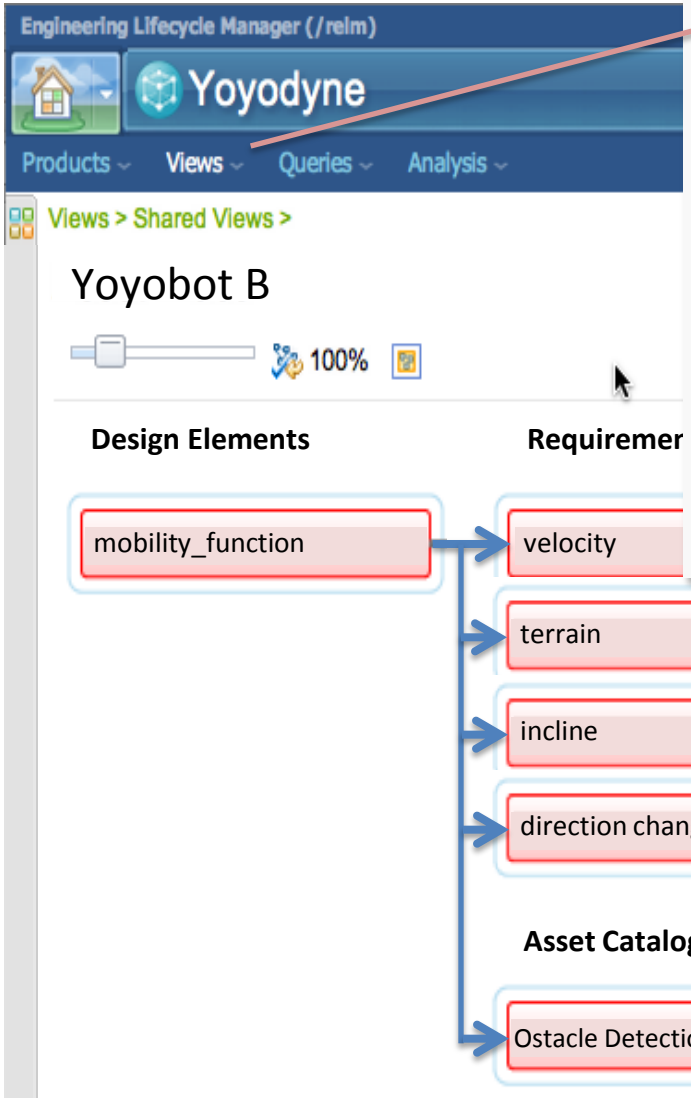
3. vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

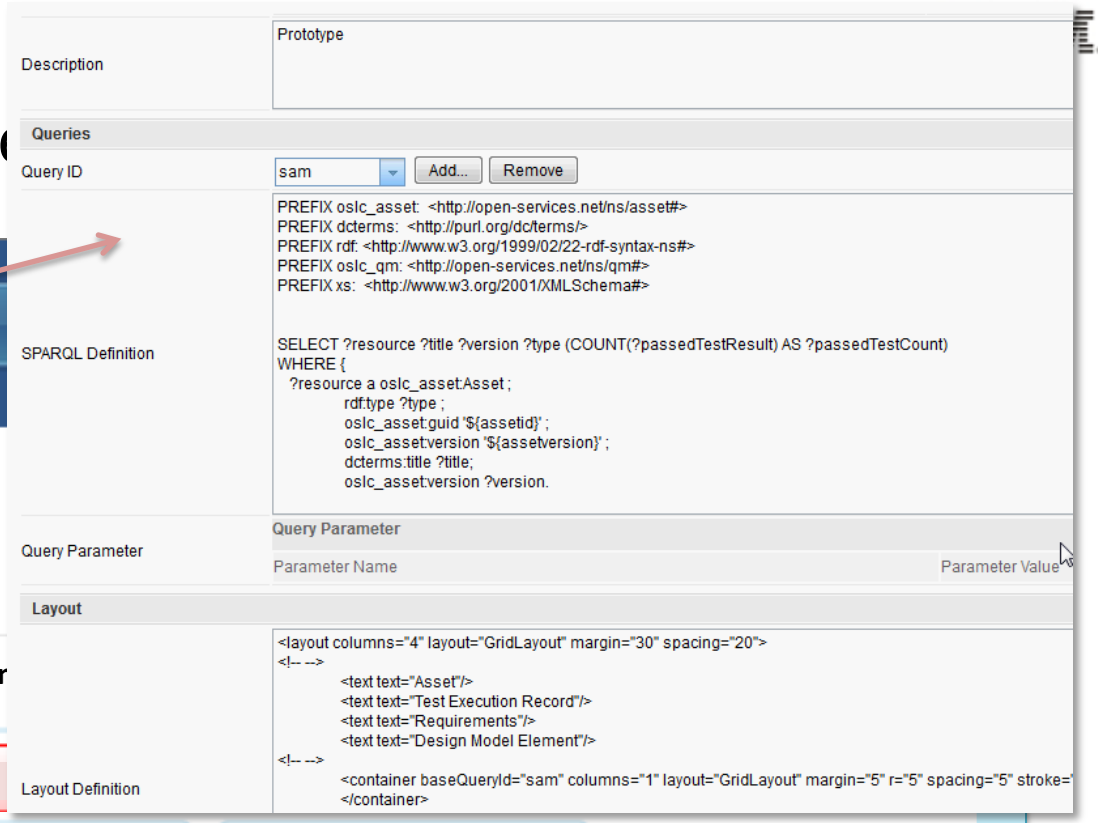
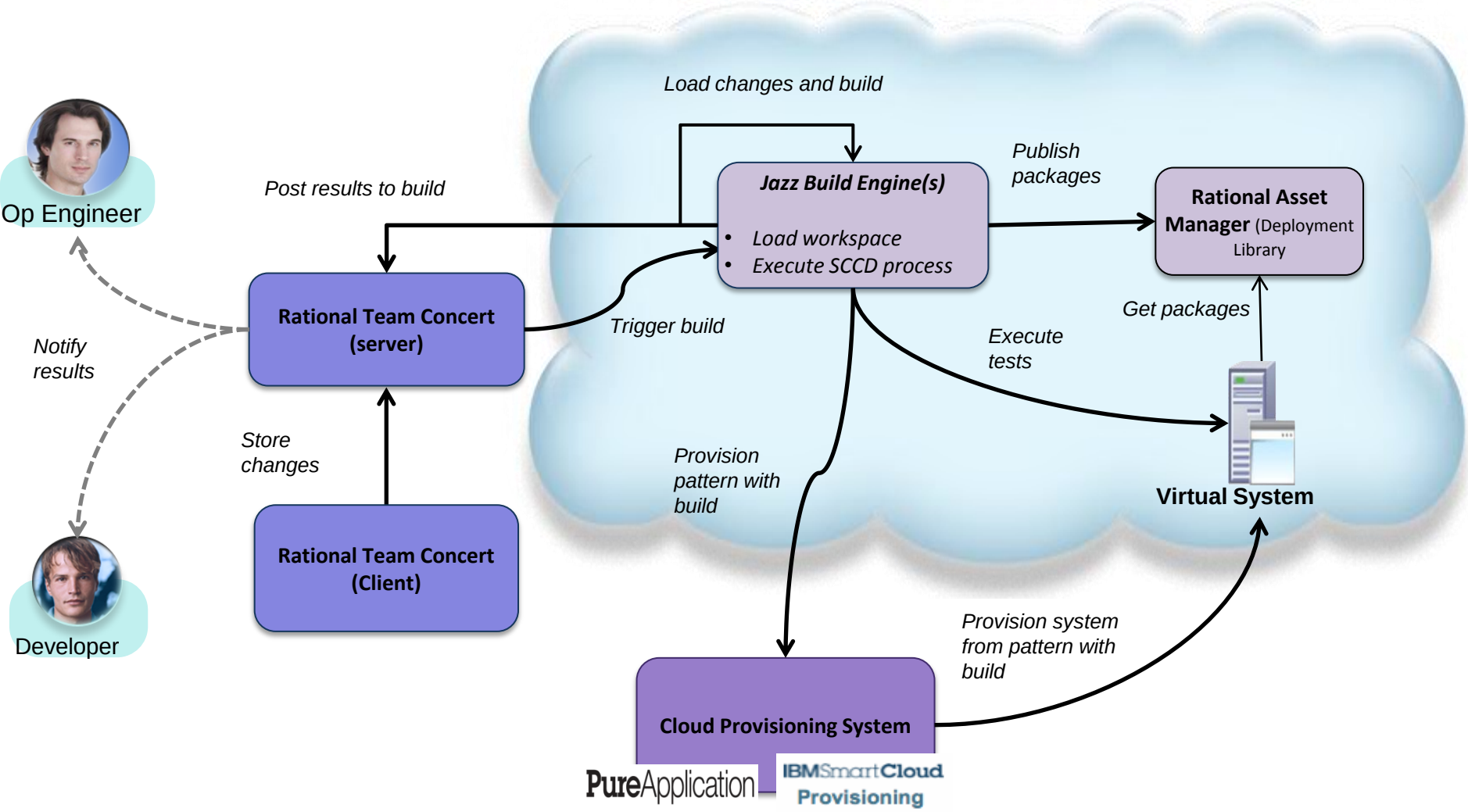
Engineering

Robot's Obstacle Detection System



Link


Engineering Lifecycle Manager (/relm)
Yoyodyne
Products
Views
Queries
Analysis
Views > Shared Views >
Yoyobot B
100%
Design Elements
Requiremer
mobility_function
velocity
terrain
MR9: Terrain
incline
MR10: Incline
direction change
MR3: Direction Change
Asset Catalog
Ostacle Detection Kit [1.0]


Description
Prototype
Queries
Query ID
sam
Add...
Remove
SPARQL Definition
PREFIX oslc_asset: <http://open-services.net/ns/asset#>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX oslc_qm: <http://open-services.net/ns/qm#>
PREFIX xs: <http://www.w3.org/2001/XMLSchema#>
SELECT ?resource ?title ?version ?type (COUNT(?passedTestResult) AS ?passedTestCount)
WHERE {
?resource a oslc_asset:Asset ;
rdf:type ?type ;
oslc_asset:guid '${assetid}' ;
oslc_asset:version '${assetversion}' ;
dcterms:title ?title;
oslc_asset:version ?version.
Query Parameter
Query Parameter
Parameter Name
Parameter Value
Layout
Layout Definition
<layout columns="4" layout="GridLayout" margin="30" spacing="20">
<!-- -->
<text text="Asset"/>
<text text="Test Execution Record"/>
<text text="Requirements"/>
<text text="Design Model Element"/>
<!-- -->
<container baseQueryId="sam" columns="1" layout="GridLayout" margin="5" r="5" spacing="5" stroke=
</container>

Continuous Delivery architecture flow