

Living Architectures - from eclipse to jazz

Erich Gamma
IBM Distinguished Engineer
IBM Rational Zurich Research Lab

Outline

- First assignment
 - ▶ A tools platform – Eclipse
- Second assignment
 - ▶ A tools integration platform – Jazz
- Comparison and Conclusion

First Assignment: Eclipse

- A tools platform
 - Seamless integration
 - Easy to extend
 - Scalable to many extensions
 - Java APIs

Eclipse Architecture Layers: how buildings last

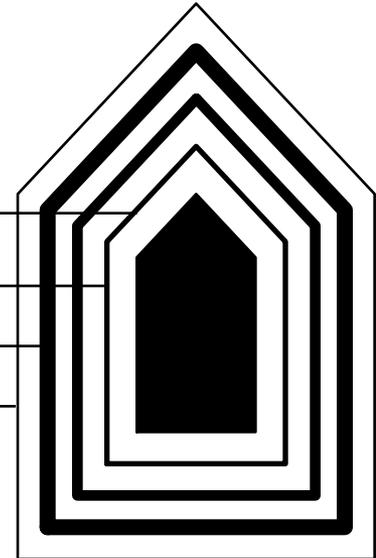
- **Stewart Brand: how buildings learn**
– what happens after they're built

stuff: furniture

services: electrical, plumbing (7-15y)

structure: foundation, load bearing walls (30-300y)

site: geographical setting (forever)

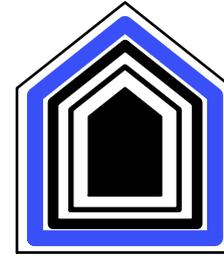


Site

- **layers:**

- evolve at different rates during the life of a building
- shear against each other as they change at different rates
- an adaptive building must allow [slippage](#)
- [a building that lasts is adaptive and can change over time](#)

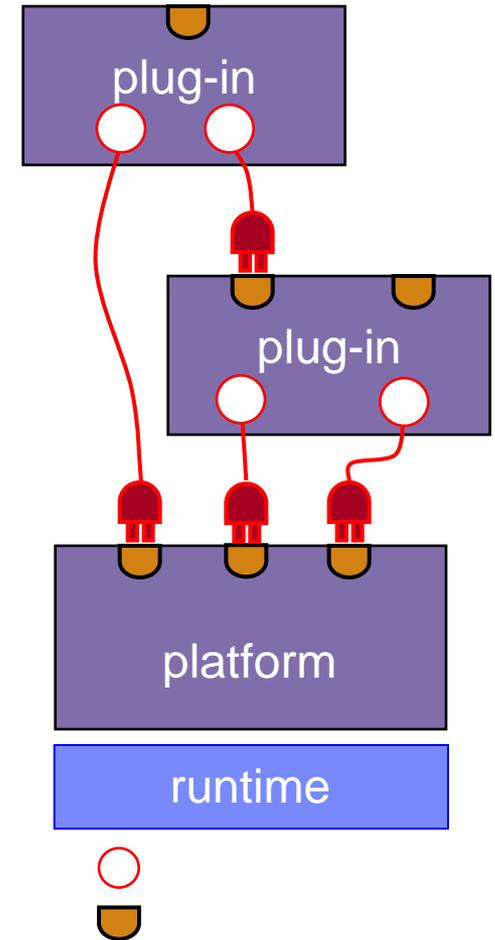
structure foundation



- the eclipse plug-in architecture
- everything is a plug-in
 - simple and consistent

eclipse plug-in architecture

- **plug-in == component**
 - set of contributions
 - smallest unit of Eclipse function
 - details spelled out in plug-in manifest
- **extension point** – named entity for collecting contributions
- **extension** – a contribution
 - Example: a specific spam filter tool
- **runtime** – controls and manages contributions

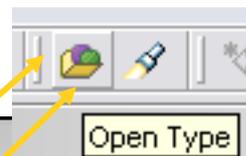


scalability

user visible appearance

`<action`

```
toolbarPath="search"  
icon="icons/opentype.gif"  
toolTip="Open Type"  
class="org.eclipse.jdt.OpenTypeAction"/>
```



Declarative Definition (manifest)

lazily instantiated using reflection

Procedural Implementation (Java JAR)

`org/eclipse/jdt/OpenTypeAction.class`

contribution implementation



services plumbing: APIs

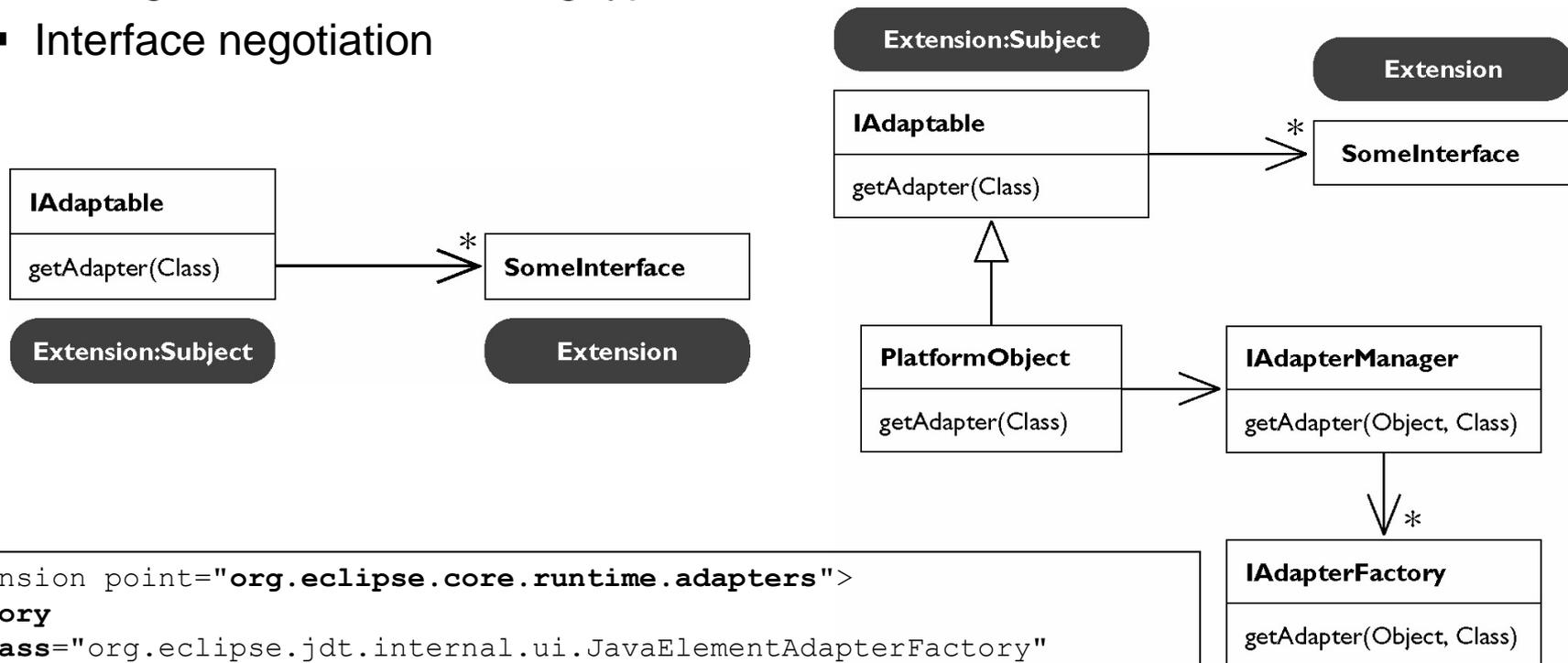
- Plug-in dependencies through APIs
- But... APIs don't just happen; we need to design them
 - specifications with precisely defined behavior
 - what you can assume (and what you cannot)
 - it works \neq API compliant
 - documented classes \neq API
- Must not break existing clients when evolving APIs

API Stability Techniques

- compatibility layer
- eclipse extension interface support: IAdaptable
- I*2 extensions interfaces
- restart in a new name space/package

extension interfaces: IAdaptable

- adding interfaces to existing types
- Interface negotiation



```

<extension point="org.eclipse.core.runtime.adapters">
<factory
  class="org.eclipse.jdt.internal.ui.JavaElementAdapterFactory"
  adaptableType="org.eclipse.jdt.core.IJavaElement">
  <adapter type="org.eclipse.ui.IPersistableElement"/>
  ...
</factory>

```

I*2 extension interfaces

- add new methods in extending API interface with extension interfaces
 - avoids breaking existing implementors of an interface

```
public interface IActionDelegate { ... } // original interface
```

```
public interface IActionDelegate2 extends IActionDelegate {  
    void dispose();  
}
```

```
if (d instanceof IActionDelegate2) {  
    IActionDelegate2 d2 = (IActionDelegate2) d;  
    d2.dispose(); // call new method  
}
```

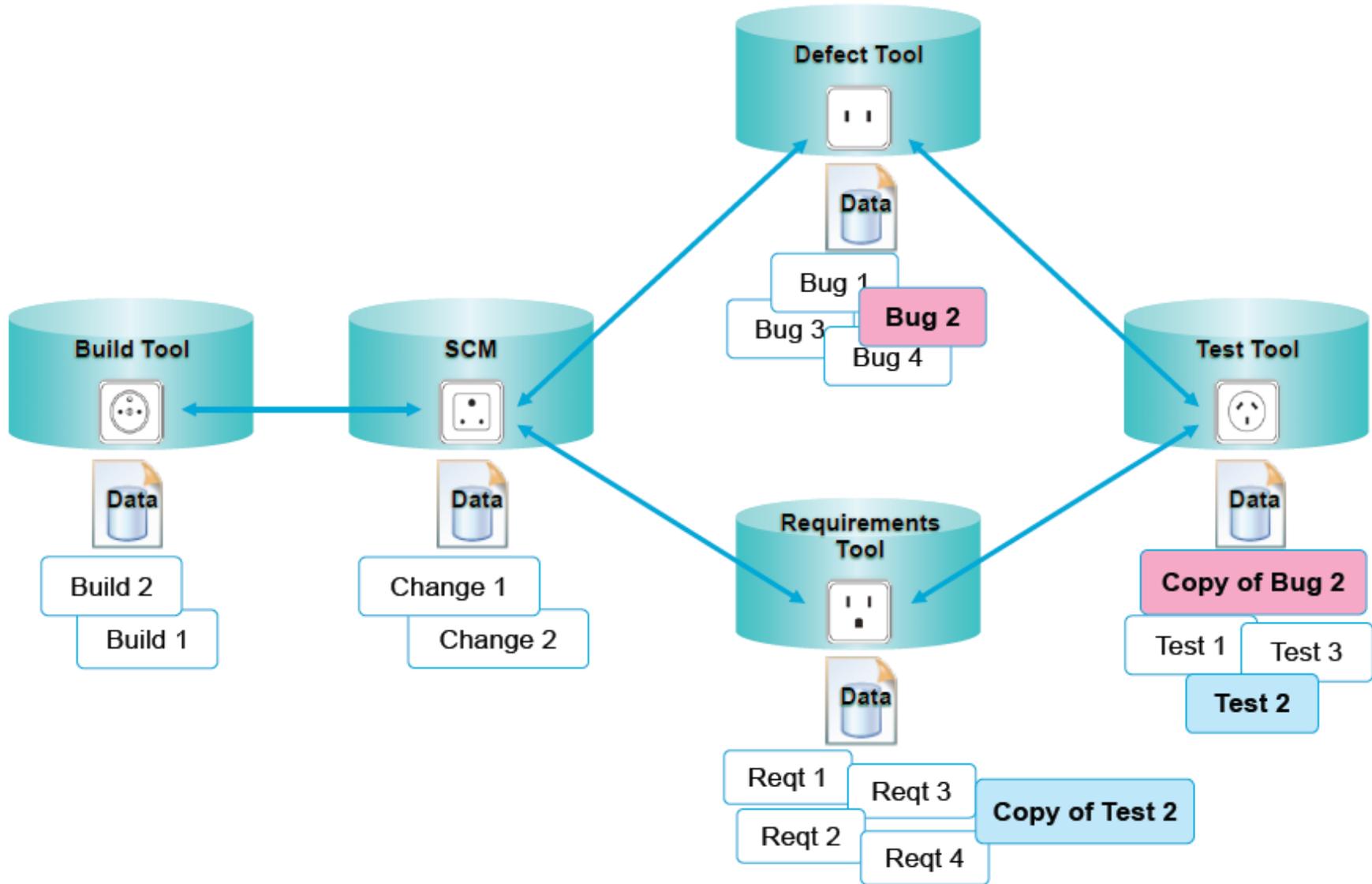
Key Lessons

- APIs are a huge commitment
 - ▶ the tyranny of stable APIs
 - API layers
 - I*2... I*7
- Version challenge for product developers
 - ▶ which API level does our product require and support
 - n-1, n-2
 - ▶ Lockstep version upgrades

Next assignment: A Tools **Integration** Platform

- Common goal
 - ▶ Rich integration - loose coupling
- New goal
 - ▶ Avoid lockstep version upgrade
 - ▶ Independent upgrade - customers must be able to upgrade their products **one at a time** in the order of their choice

Traditional Tools Integration



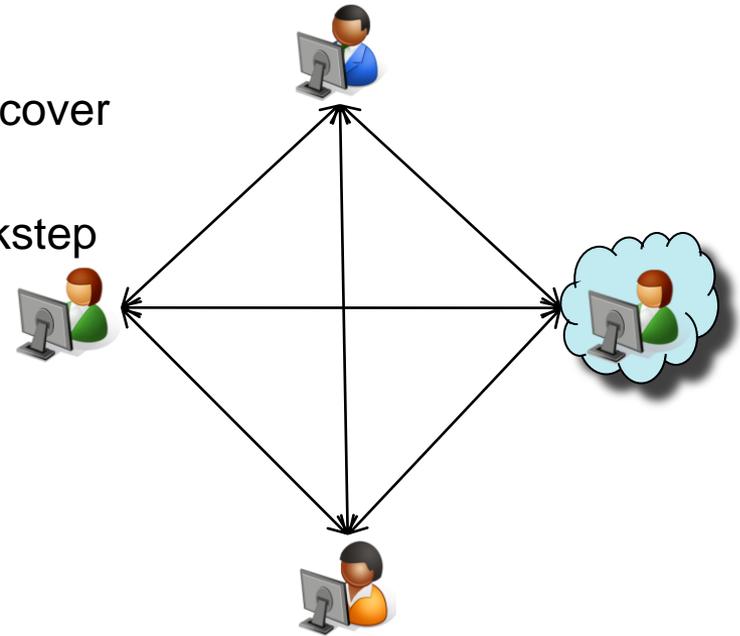
Traditional Tools Integration...

- **Point-to-point integrations**

- ▶ Limited coverage: there are too many tools to cover more than a small fraction of possibilities
- ▶ Tight dependencies between tools require lockstep upgrades
- ▶ Proprietary APIs create vendor lock-in

- **State of the Art: shared repository**

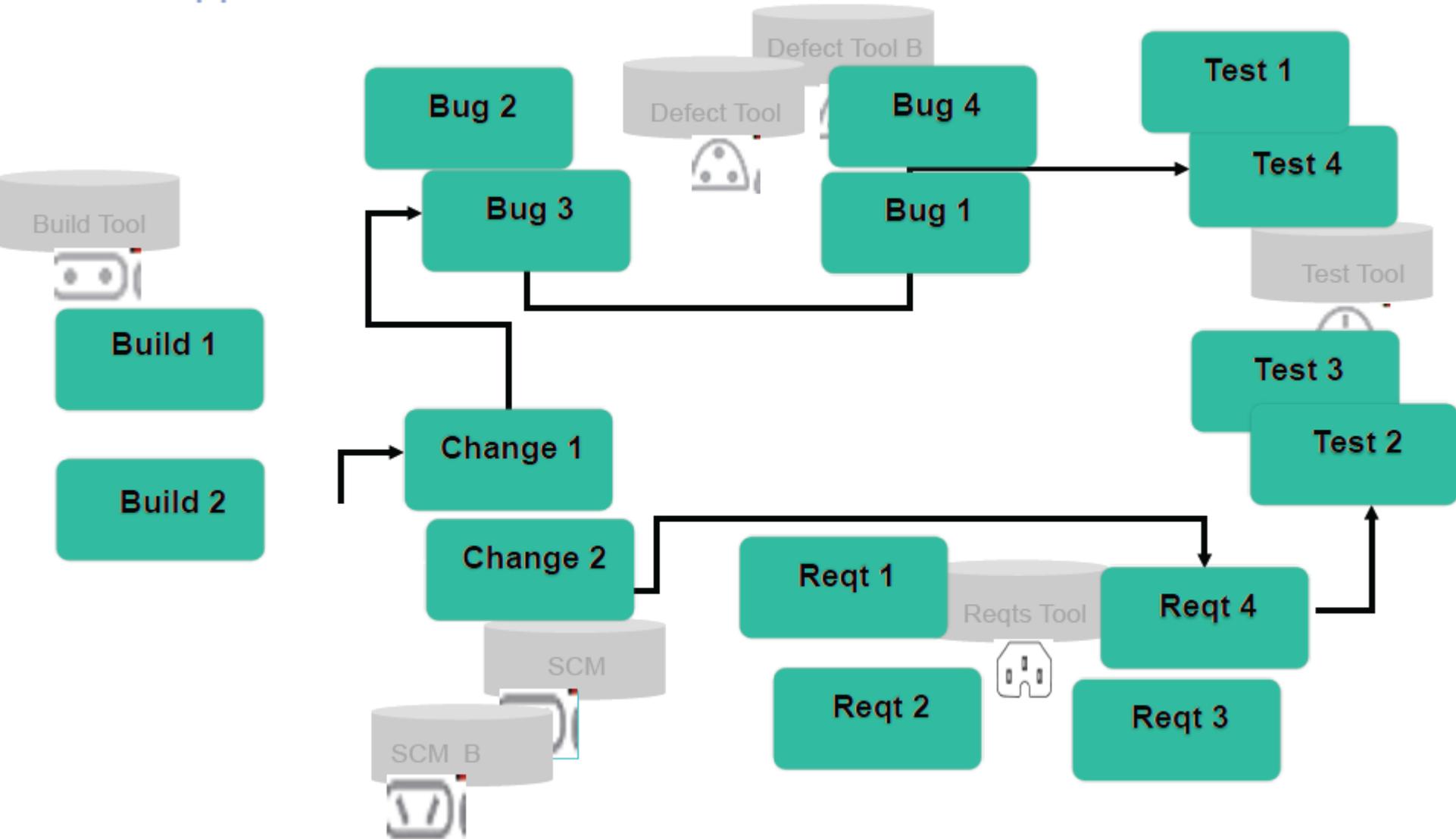
- ▶ Hard to add existing (legacy) tools
- ▶ Difficult to evolve tools individually
- ▶ Limited to a single vendor's tools or affiliates



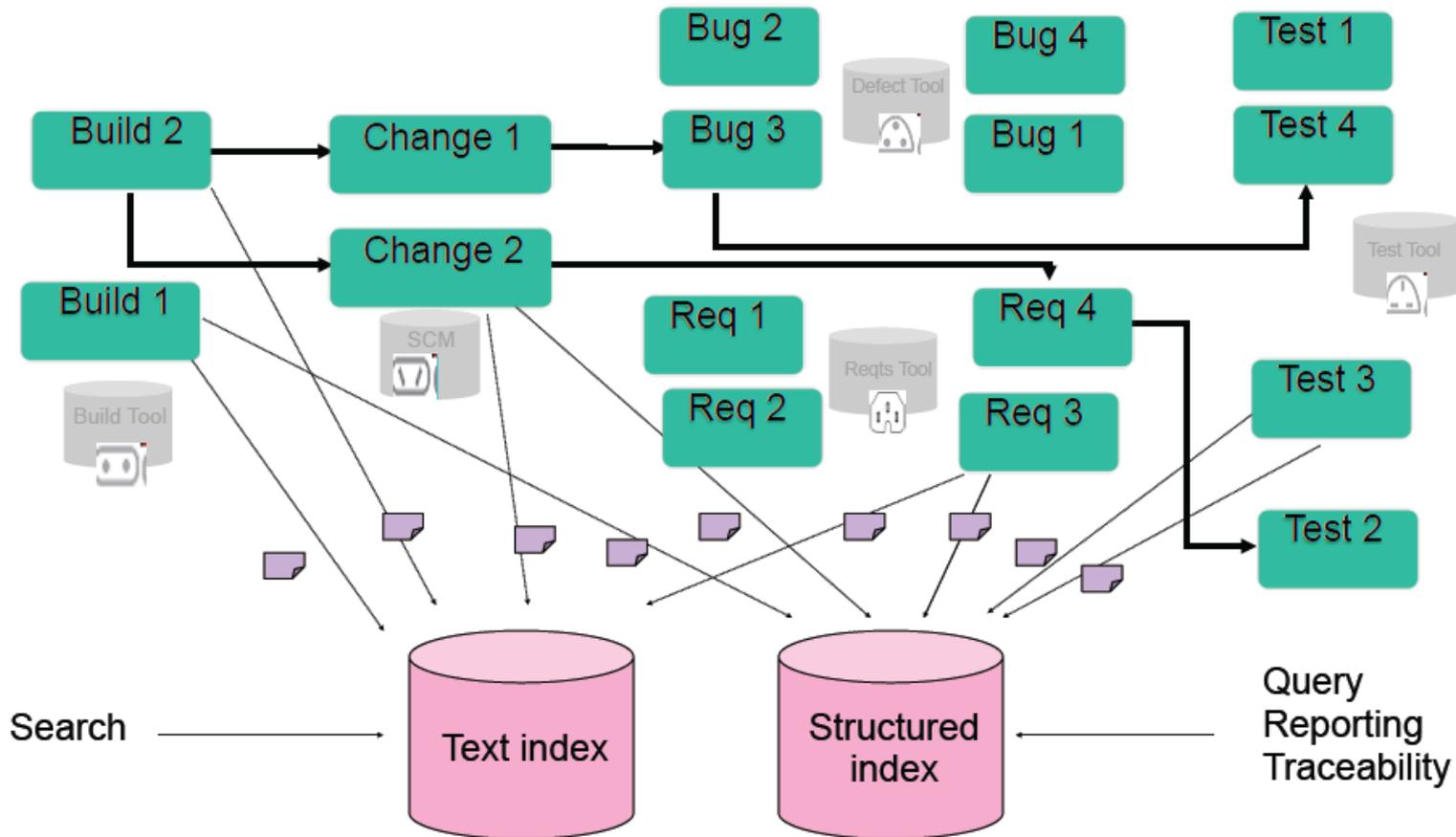
Enter “Linked Data”

- Linked Data is an approach, defined by Tim Berners-Lee, to data integration on the Web
 - ▶ <http://www.w3.org/DesignIssues/LinkedData.html>
- Linked data principles
 1. Use URIs as names for things.
 2. Use HTTP URIs so that people can look up those names.
 3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).
 4. Include links to other URIs so that they can **discover** more things.

Linked Lifecycle Data

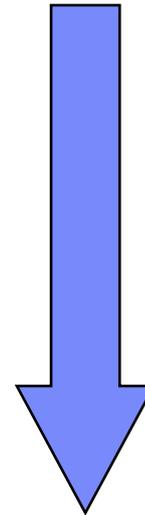


Finding and Analyzing Linked Lifecycle Data



From Linked Data to an Integration Toolbox

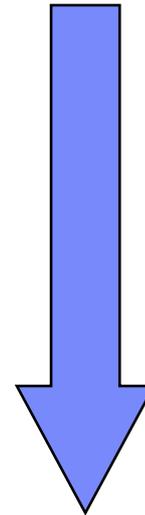
- UI mash-ups – UI integration
 - Provide and consume Open Social gadgets
- Linked data – Data integration
 - Open services for life cycle collaboration
 - Creating linked data
 - Delegated UIs
 - Exploring linked data
 - UI previews
- Leverage Jazz Foundation integration services
- Utilize Jazz Foundation functional services



Deeper Integration

Linked Data is not sufficient – Integration Toolbox

- UI mash-ups
 - Provide and consume Open Social gadgets
- Linked data
 - Open services for life cycle collaboration
 - Creating linked data
 - Delegated UIs
 - Exploring linked data
 - UI previews
- Leverage Jazz Foundation integration services
- Utilize Jazz Foundation functional services



Deeper Integration

OpenSocial www.opensocial.org

- OpenSocial defines a common API for social applications across multiple websites. With standard JavaScript and HTML, **developers can create apps that can embed and be embedded** within a social network itself, or access a social network's data from anywhere on the web.
- We focus on the mash up part
- You can provide new or consume existing gadgets

Open Social Gadgets

Rational Team Concert

Dashboards Project Areas Work Items Plans Builds Reports

Erich Gamma's Dashboard

Home ▾
Team
PMC
New tab
+

Remember The Milk
▾ ○ ✕

[+ Add Task](#)
[Refresh](#) [Settings](#) [Logout](#)

Overdue

Mar 5 ▾ [Make Build Input](#) ■

May 2 ▾ [make hotel reservation](#) ■

Today + May 4

No tasks due today.

News Feed (82 new)
▾ ○ ✕

- ☰ [2] Provide a native iPhone App for RTC/Work Items (113728) 10 minutes ago
- ☰ Import and Export from MS Project (26076) 1 hour ago
- ☰ [16] Provide development environment guidelines and tools (113355) 5 hours ago
- ☰ [9] Track C/ALM 2010 M6 build and install (May 1 - May 7) (113651) 8 hours ago

Page 1 of 11

Bookmarks
▾ ○ ✕

RQM

RRC

Pending approvals for me (2)
▾ ○ ✕

- ☑ 101461: Don't deploy the deprecated Scrum template on new servers
- ☑ 66070: Template Work Item with Blue Prints for Approvals

Open assigned to me (3)
▾ ○ ✕

- 🔄 93642: Help mapping of MCIF practices into RTC
- ☑ 82012: Consider to rename resolution Won't Fix to Works as Designed on jazz.net
- ☑ 62601: RTC 2.0 plan approval tracking

Current Plans for Erich Gamma (10)
▾ ○ ✕

Team Concert iFix3	
C/ALM 2010 *** Deprecated ***	
Team Concert Developer	
Work Item	

Open Social Gadget

Open Social Gadgets

Work items gadget in Open Social Containers (iGoogle, gmail)

iGoogle™

Google Search

I'm Feeling Lucky

Show this page every time I start to browse the web. [Make iGoogle my homepage](#)

Home

- Google Calendar
- CNN.com
- Weather
- IBM Stock
- Jazz Community Ne...
- All Open

Another Tab

- Date & Time
- gm_countdown.xml

Updates

- Friends

Chat

Search, add, or invite

- Dejan Glozic
- Set status here

Chat with friends in iGoogle!

Rather stay

All Open

- 3: CompositeRunner.filter incorrect if child throws NoTestsRema
- 4: BaseTestRunner.getTest() requires class to extend TestCase
- 8: assertThat signature does not match Matcher
- 9: assertThat fails with Class tests (documentation problem)
- 10: Who's responsible to deploy in maven central repository?
- 11: No download link to latest version
- 12: timeout doesn't work properly for >=2 cases in junit4.3?
- 13: Tests on protected methods fail
- 18: distribution cookbook out of date
- 23: suite() method should not matter for JUnit 4 tests

Page 1 of 3

Google Calendar

June 2010						
S	M	T	W	T	F	S
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

Today (Tue, Jun 1)

- 9:00am Jazz Foundation Deep Dive

Tomorrow (Wed, Jun 2)

- 8:00am Jazz Foundation PMC

Wed, Jun 9

- 8:00am Jazz Foundation PMC

Tue, Jun 15

- 9:00am Jazz Foundation Deep Dive

[Today](#) | [Add](#) [Options](#)

Gmail Calendar Documents

Gmail
by Google

[Compose Mail](#)

Inbox

- Buzz
- Starred
- Sent Mail
- Drafts

Personal

Travel

6 more

Contacts

Tasks

+ Dejan Glozic

Search, add, or invite

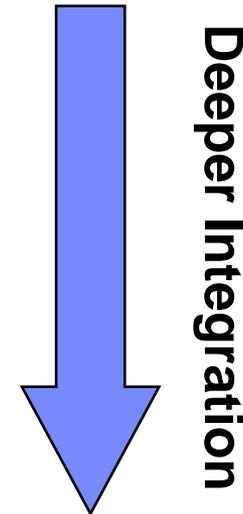
+ Google Calendar

All Open

- 3: CompositeRunner.filter incorrect if child throws NoTestsRema
- 4: BaseTestRunner.getTest() requires class to extend TestCase
- 8: assertThat signature does not match Matcher
- 9: assertThat fails with Class tests (documentation problem)
- 10: Who's responsible to deploy in maven central repository?

Linked Data is not sufficient – Integration Toolbox

- UI mash-ups
 - Provide and consume Open Social gadgets
- **Linked data**
 - Open Services for Life Cycle collaboration (OSLC)
 - Creating linked data
 - Delegated UIs
 - Exploring linked data
 - UI previews
- Leverage Jazz Foundation integration services
- Utilize Jazz Foundation functional services



Open Services for Lifecycle Collaboration OSLC

- Applies Linked Data principles to Lifecycle artifacts
 - ▶ Provides specifications for sharing lifecycle data
- A minimalist scenario driven approach
- Effort is divided into domains
 - ▶ Change Management
 - ▶ Quality Management
 - ▶ Requirements Management
- Builds on a Common core specification

Open Services for Lifecycle Collaboration

Community specifications for lifecycle integration

Home
About
Community
Wiki
Learn



Open Services for Lifecycle Collaboration

open community. open interfaces. open possibilities.

Open Services for Lifecycle Collaboration (also known as OSLC or Open Services) is a community effort to help software delivery teams by making it easier to use lifecycle tools in combination. The OSLC community is creating open, public descriptions of resources and interfaces for sharing the things that software delivery teams rely on, like change requests, test cases, defects, requirements and user stories.

By agreeing on common specifications for lifecycle resources and the services to access them, we can eliminate traditional barriers between tools and open the door to new forms of collaboration. OSLC can bring value to software delivery teams and tool providers alike, from the most Agile to the most ceremonial of projects, and for commercially-licensed, open source, and internally developed tools. [More](#).

With OSLC's open and scenario-based approach, businesses benefit from the ability to tie disparate tools together. This collaborative approach gives our consultants the flexibility to make lifecycle tool choices based on specific client project demands.

Randy Vogel, Accenture

Learn more

- **Presentation:** *ALM Integration in a Web 2.0 World*
- **Presentation:** *RESTful Work Items: Opening up Collaborative ALM*
- **Podcast:** *Open Services bears first fruit. A conversation with Steve Abrams, Mik Kersten, and Carl Zetie.*
- **Whitepaper:** *The Case for Open Services*
- **Podcast:** John Wiegand and Steve Abrams introduce the OSLC initiative

News and events

- Implementations delivered for **Change management 1.0** spec ([press release](#))
- **Change management 2.0** spec workgroup expanding participants.
- **Requirements management and Asset management** workgroups draft early specs.
- **Primer** authored for Software Estimation and Measurement
- New **Reporting** workgroup call for participation.

Quick links

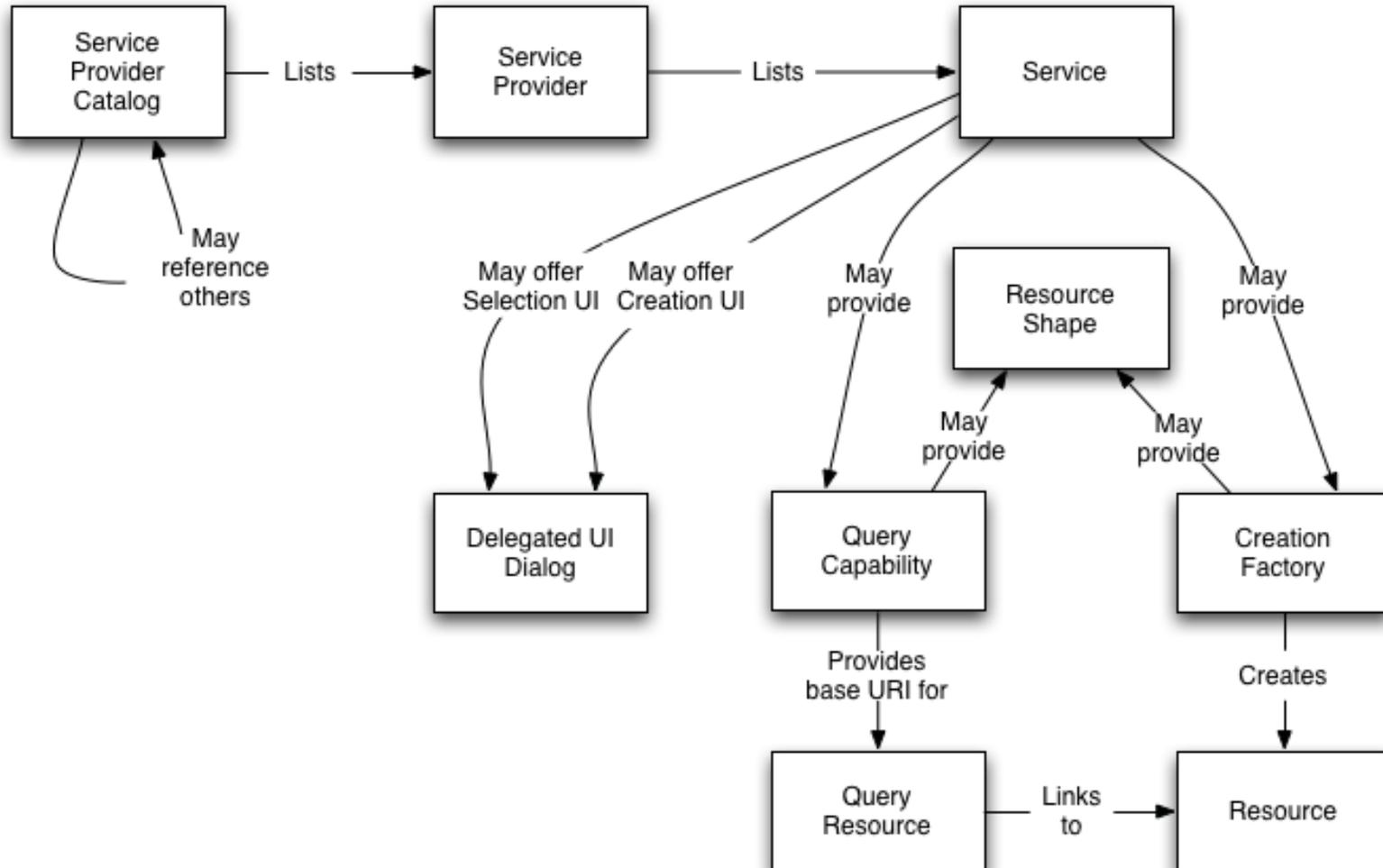
- **Wiki:** Open Services specifications
- **Mailing list:** OSLC community
- **Blog:** *Let's try something different* - Carl Zetie's commentary on OSLC
- Twitter - follow us: [@oslcNews](#)

Terms of Use
Privacy
Feedback

Suppose tools exposed their data in a consistent way?

- Open community of individuals interested in improving lifecycle integration.
- Goals
 - 1. Make life better for software and product delivery teams**
 - 2. Reduce the complexity and cost for tool providers in integrating tools together**
 - 3. Open up new possibilities in the marketplace by opening up the way lifecycle tools and data can be used in ALM, PLM and outside**
- Creating open, public specifications that describe resources and interfaces for sharing the things that software and product delivery teams rely on.

OSLC Core concepts



OSLC Core provides Guidance for

- Resource representations
 - ▶ must provide an RDF/XML representations of a resource
 - ▶ may provide JSON, Turtle, ATOM
- Link modeling
- Partial updating of resources
- UI Previews
- Delegated UIs
- Specification versioning

Specification Example: Change Management

<http://open-services.net/bin/view/Main/CmSpecificationV2>

Document

[CM RESTful Services](#)

[CM Change Request Resource Defin](#)

[CM Simple Query Syntax](#)

[CM JSON Format](#)

[CM Delegated Resource Selection ar](#)

[CM Service Description](#)

Resource URIs and Methods

Resource	URI	GET	POST	PUT	DELETE	Description
Collection of Change Requests	{CR Collection URI}	Y	*	N	N	A collection of change requests
Change Request	{CR URI}	Y	N	Y	Y	An identifiable change request, by a permanant URI

* - the collection MAY support creation on its URI, see [Create a new Change Request](#)

N - in the HTTP verb column indicates that a Service Provider MUST return a 405 Not Supported response

For a complete list of [HTTP Response Codes](#)

URIs for working with Change Requests

The following table outlines the key items that are exposed in the Change Management Service Discovery Document. Details of each of these capabilities will follow in subsequent sections.

Purpose	Discovery Element	URL*	Section	Support
Resource Creation	<factory>	{Resource Creation URL}	Create a new Change Request	REQUIRED
Resource Query	<simpleQuery>	{Simple Query URL}	Get a Collection of Change Requests	REQUIRED
Resource Selection UI	<selectionDialog>	{Selection Dialog URL}	Resource Selection	REQUIRED
Resource Creation UI	<creationDialog>	{Creation Dialog URL}	Resource Creation	REQUIRED

Retrieving a Defect

```
GET https://rtc.com:9443/jazz/resource/itemOid/com.ibm.team.workitem.WorkItem/_0J39QJu-Ed6cerS91b5AWw
Accept: application/x-oslc-cm-change-request+xml
```

```
<?xml version="1.0" encoding="UTF-8"?>
<oslc_cm:ChangeRequest
  xmlns:rtc_cm="http://jazz.net/xmlns/prod/jazz/rtc/cm/1.0/" xmlns:oslc_disc="http://open-services.net/xmlns/disc
  xmlns:dc="http://purl.org/dc/terms/" xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/"
  xmlns:jp="http://jazz.net/xmlns/prod/jazz/presentation/1.0/" xmlns:jd="http://jazz.net/xmlns/prod/jazz/discover
  xmlns:oslc_cm="http://open-services.net/xmlns/cm/1.0/" xmlns:atom="http://www.w3.org/2005/Atom"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:calm="http://jazz.net/xmlns/prod/jazz/calm/1.0/">
  <dc:type rdf:resource="https://rtc:9443/jazz/oslc/types/_gasc4Ju-Ed6cerS91b5AWw/defect"/>
  <dc:identifier>9</dc:identifier>
  <dc:created>2009-09-07T14:59:06.333Z</dc:created>
  <dc:creator rdf:resource="https://rtc:9443/jazz/oslc/users/_6I8ZMJu9Ed6cerS91b5AWw"/>
  <dc:title>My First Bug</dc:title>
  <dc:description>This is my first bug</dc:description>
  <dc:subject/>
  <dc:modified>2009-09-07T14:59:06.348Z</dc:modified>
  <oslc_cm:priority rdf:resource="https://rtc:9443/jazz/oslc/enumerations/_gasc4Ju-Ed6cerS91b5AWw/priority/priori
  <oslc_cm:severity rdf:resource="https://rtc:9443/jazz/oslc/enumerations/_gasc4Ju-Ed6cerS91b5AWw/severity/severi
```

Retrieving JSON Representation of a Defect

```
GET https://rtc.com:9443/jazz/resource/itemOid/com.ibm.team.workitem.WorkItem/_0J39QJu-Ed6cerS91b5AWw?oslc
Accept: application/x-oslc-cm-change-request+json
```

```
{
  "dc:title": "My First Bug",
  "rdf:resource": "https://rtc:9443/jazz/resource/itemOid/com.ibm.team.workitem.WorkItem/_0J39QJu-Ed6cerS91b5AWw",
  "rtc_cm:comments": [
    {
      "rdf:resource": "https://rtc:9443/jazz/oslc/workitems/_0J39QJu-Ed6cerS91b5AWw/rtc_cm:comments/0"
    },
    {
      "rdf:resource": "https://rtc:9443/jazz/oslc/workitems/_0J39QJu-Ed6cerS91b5AWw/rtc_cm:comments/1"
    }
  ]

  "rtc_cm:com.ibm.team.workitem.linktype.relatedworkitem.related": [
    {
      "rdf:resource": "https://rtc:9443/jazz/resource/itemOid/com.ibm.team.workitem.WorkItem/_CD62QJu-Ed6cerS91b5AWw",
      "oslc_cm:label": "10: My Second Bug"
    },
    {
      "rdf:resource": "https://rtc:9443/jazz/resource/itemOid/com.ibm.team.workitem.WorkItem/_725cAJvDEc",
      "oslc_cm:label": "11: My Third Bug"
    }
  ]
}
```

Linking

- Allow to establish relationships between your resources and resources provided by others
 - ▶ Consume resource pickers provided by others
 - ▶ Support OLSC linking protocols to establish links
 - ▶ Support resource pickers that can be consumed by others

picker dialog:

The 'Add Existing Defect' dialog box is shown. It has a title bar with a close button. The main area contains the following fields and controls:

- Project Area:** A dropdown menu with 'Smarter Real Estate' selected.
- Type:** A dropdown menu with 'Defect' selected.
- Use Work Item ID or Words Contained in the Text:** A text input field containing 'test'. To the right of this field, it says '2 result(s)'.
- Matching Work Items:** A list box containing two items: '10: Failing Test Case "Test"' and '11: 2 Failing Test Case "Test"'. The list box has a scroll bar on the right.
- Buttons:** 'OK' and 'Cancel' buttons at the bottom right.

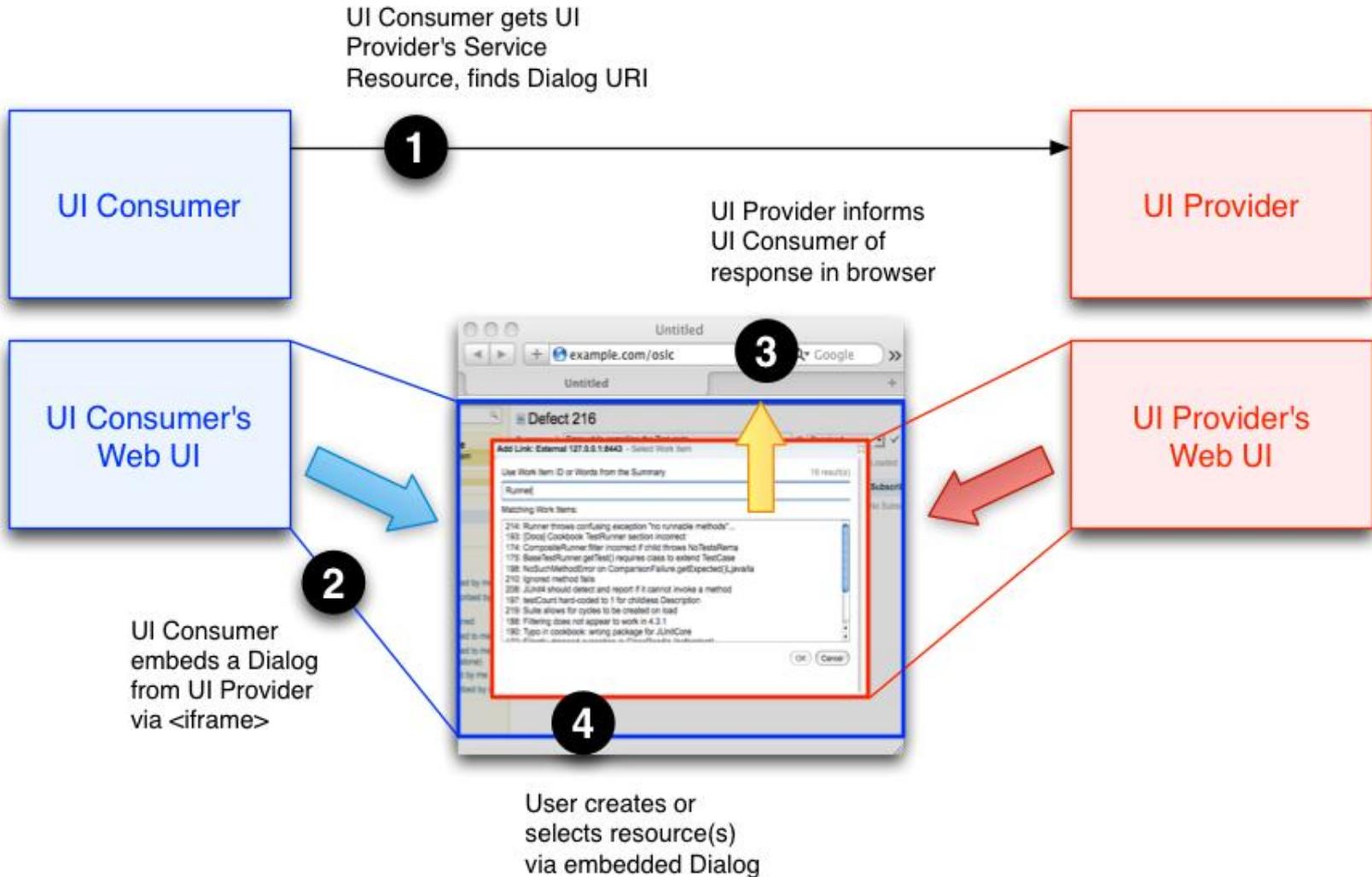
creation dialog:

The 'Create Defect' dialog box is shown. It has a title bar with a close button. The main area contains the following fields and controls:

- Title:** 'Create Defect New Defect'.
- Defect/samDb00000081:** A dropdown menu with a small icon to its right.
- Buttons:** 'Save' and 'Cancel' buttons at the top right.
- Tabs:** 'Main', 'Attachments', and 'Customer' tabs. 'Main' is selected.
- ID:** A text input field containing 'samDb00000081'.
- State:** A dropdown menu with 'Submitted' selected.
- *Headline:** A text input field.
- Project:** A dropdown menu.
- *Severity:** A dropdown menu.
- Priority:** A dropdown menu.
- Owner:** A dropdown menu.
- Keywords:** A text input field with a pencil icon to its right.
- Symptoms:** A text input field with a pencil icon to its right.
- Description:** A large text area.
- Template:** A dropdown menu at the bottom with a 'Load' button to its right.

Delegated UI Dialogs

For resource creation and selection



Delegated UIs: Embedded Editors

- Using Open Social for composing UIs

The screenshot displays the IBM Quality Management (QM) interface. At the top, there is a navigation bar with tabs for Dashboards, Requirements, Planning, Construction, Lab Management, Builds, Execution, Reports, and Defects. A search bar and user profile are also visible. Below the navigation bar, there are several tabs for different views: Dashboards, Classic CD V1, Add CD to Shopping C..., Add CD To Cart, Execution Result, and Work Items: CD Class... The main content area shows a 'Defect 81' entry. The summary is 'Failing Test Case "Add CD to Shopping Cart"'. The defect is in a 'New' state. The 'Details' section includes fields for Type (Defect), Filed Against (...c (Change Management)), Severity (Normal), Found In (Unassigned), Project Area (CD Classic (Change Management)), Team Area (CD Classic (Change Management)), Creation Date (Oct 5, 2010 12:14 AM), and Created By (eg). The 'Quick Information' section lists various relationships: Subscribers (1): e, Affects Test Execution (1): 1, Blocks Test Execution (1): 1, Affects Plan Item (1): 1, Affects Requirement (1): 1, Tracks Test Case (1): 1, and Tracks Test Plan (1): 1. The 'Description' section is partially visible at the bottom, showing 'Test Plan: Classic CD V1', 'Test Case: Add CD to Shopping Cart', 'Test Script: Add CD To Cart', and 'Project Area: CD Classic (Quality Management)'.

UI Preview (was Compact Rendering)

- Allows to show information about linked resources

The screenshot displays the CD Classic (Change Management) interface. The main window is titled "CD Classic (Change Management)" and has tabs for "General" and "Testing". Under the "Testing" tab, there is a section for "Defects blocking Tests (1)" which contains a list item "81: Failing Test Case 'Add CD to Shopping Cart'". Below this, a specific test case "4: Add CD to Shopping Cart" is highlighted. A preview window for this test case is open, showing the following details:

Last Result:	ID:	4
Owner:	Project Area:	CD Classic (Quality Management)
Test Case:	Environment:	Add CD to Shopping Cart
Weight:	Modified:	100 10/5/10 12:13:48 AM CEST

At the bottom of the preview window, there is a "Show More" link.

UI Preview

- GET with Accept header application/x-oslc-compact+xml
- Return is an RDF/XML document, return media type is application/x-oslc-compact+xml
- <http://open-services.net/bin/view/Main/OslcCoreUiFreview>

```
<?xml version="1.0" encoding="UTF-8"?>

<oslc:Compact
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/terms/"
  xmlns:oslc="http://open-services.net/xmlns/oslc#"
  rdf:about="http://example.com/bugs/12345">

  <dc:title> 12345: &lt;s&gt;Null pointer exception during startup&lt;/s&gt; </dc:title>
  <dc:name> 12345 </dc:name>
  <oslc:icon rdf:resource="http://example.com/icons/defect.jpg" />

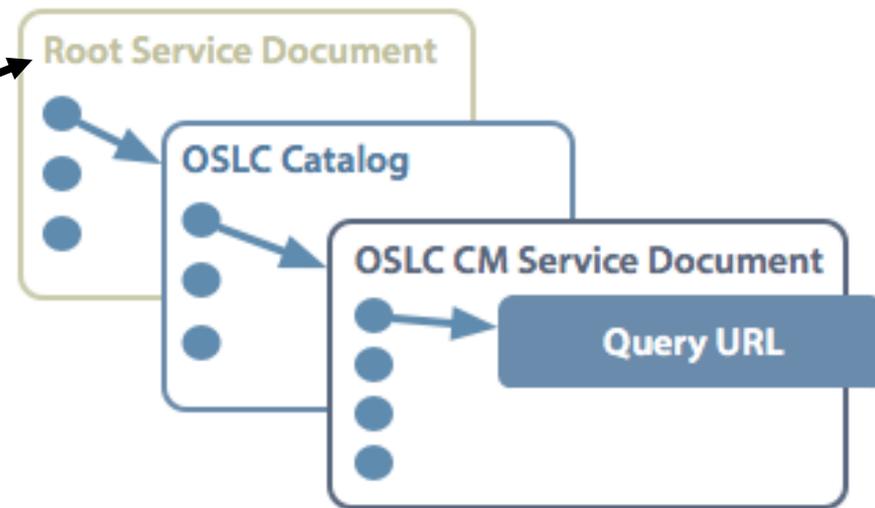
  <oslc:smallPreview>
    <oslc:Preview>
      <oslc:document rdf:resource="http://example.com/bugs/12345?hover=small" />
    </oslc:Preview>
  </oslc:smallPreview>

  <oslc:largePreview>
    <oslc:Preview>
      <oslc:document rdf:resource="http://example.com/bugs/12345?hover=large" />
      <oslc:hintWidth> 60em </oslc:hintWidth>
      <oslc:hintHeight> 20em </oslc:hintHeight>
    </oslc:Preview>
  </oslc:largePreview>

</oslc:Preview>
```

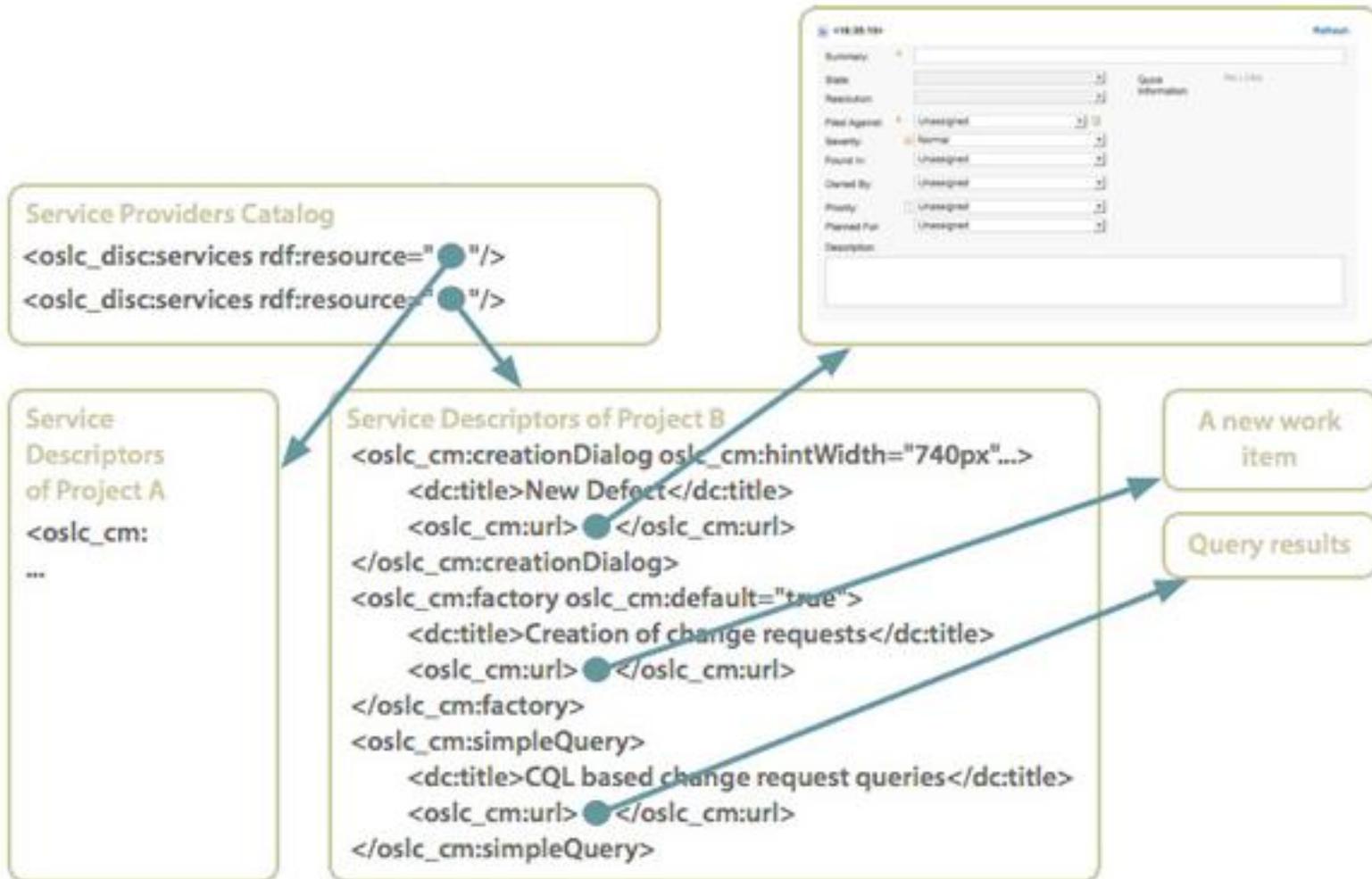
Service Discovery

1. Discover the existence of the Change Management system itself, known URL
 - ▶ E.g. <https://rtc:9443/rtc/rootservices>
2. Discover the contexts (e.g. projects) in which change requests may exist, e.g project
3. Discover the services that are provided within that context

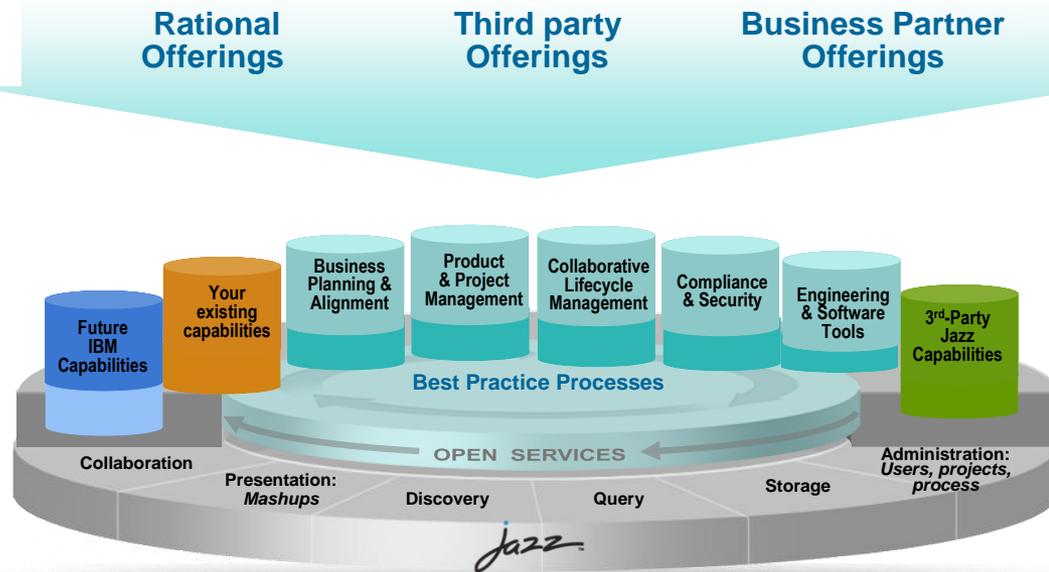


```
<rdf:Description rdf:about="https://rtc:9443/jazz/rootservices">
  ...
  <oslc_cm:cmServiceProviders rdf:resource="https://rtc:9443/jazz/oslc/workitems/catalog"/>
  ...
</rdf:Description>
```

Discovering the Creation Dialog



Jazz is a platform for transforming software delivery



Jazz is a platform for *transforming how people work together* to deliver greater value and performance from their software investments.

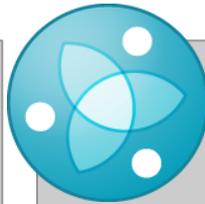
Jazz is...

- Our vision of the future of systems and software delivery
- A scalable, extensible team collaboration platform
- An integration architecture enabling mashups and non-Jazz products to participate
- A community at Jazz.net where Jazz products are built

Collaborative ALM



*Rational
DOORS Requirements
Professional*



*Rational
Team Concert*



*Rational
Quality Manager*

Requirements project

- Requirements
- Requirements Collections

Development Project

- Work items
- Plans
- Streams, Change Sets
- Builds

Test Project

- Test Cases
- Test Plans
- Test Executions

DEMO

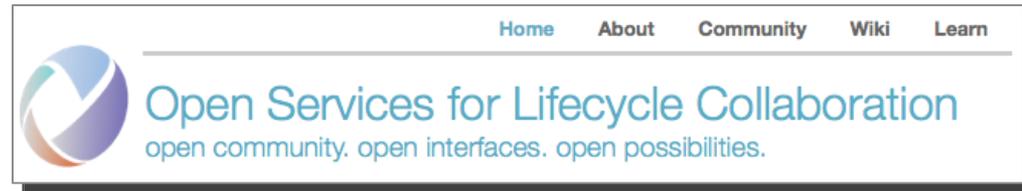
Eclipse vs. Jazz Integration Architecture

- Eclipse
 - ▶ a tools platform, implement new tools
 - ▶ Java APIs
 - ▶ Java
 - ▶ Integrate by writing new plug-ins
 - Plugin.xml
 - Factories
 - ▶ Desktop application
 - ▶ Fine grained integration
 - ▶ Independent upgrade difficult
- Jazz integration architecture
 - ▶ Integrate existing tools
 - ▶ REST based specifications
 - ▶ Many languages
 - ▶ Integrate by providing REST implementations
 - Service documents
 - Discovery, Factory URLs
 - ▶ Web (but can integrate with desktop apps)
 - ▶ Coarse grained integration
 - ▶ Supports independent upgrade

References

■ OSLC

- www.open-services.net



■ Jazz.net

- ▶ www.jazz.net

