



WHERE TEAMS ARE **R-HEROES**

Jazz and Rational Team Concert

*Eric Long, Technology Evangelist
IBM Rational Technology Events
Singapore
20 August 2008*

Agenda

- **All that Jazz**
- Rational Team Concert
 - ▶ Overview
 - ▶ Getting Started with a new Team
 - ▶ Software Configuration Management
 - ▶ Build Management
 - ▶ Change and Traceability



The Jazz project

People working together to deliver great software

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



A major investment by IBM to create a scalable, extensible team collaboration platform.

IBM's vision for the future of software delivery -- supporting a workforce that is increasingly organizationally and globally distributed, fluid and dynamic.

An evolution of the Rational portfolio, which will evolve to support the Jazz technology platform over time.

A community at **Jazz.net** – the online venue for open commercial development of select Jazz-based products.



Jazz.net: Delivering greater openness and customer participation in the products they depend on for software delivery

- IBM is opening up the Rational Software Delivery Platform for greater ease of consumption, extensibility and integration to meet the unique usage needs of our customers
- IBM is providing transparent, collaborative customer participation in the development of new Rational technologies through an open commercial community





Create a jazz.net account

Jazz.net registration is now open to everyone! In most cases, we can validate your registration and send you your password right away if you provide us with complete and accurate information.

*Create a jazz.net User ID:

[Check this User ID for availability](#)

Personal Information:

*First name:

*Last name:

*E-mail:

*Re-type E-mail:

*Company or Organization name:

*Select the entry that best describes you:

Mailing Address:

*Address line 1:

Address line 2:

*City:

State/Province (required U.S./Canada):

Zip/Postal Code (required U.S./Canada):

*Country/region:

If you have a [registration code](#), enter it below:

Why register?

Creating a jazz.net account allows you to take part in the Jazz community. You can download product Betas, and other previews of Jazz technology.

You'll be able to interact directly with the Jazz development team and other members of the Jazz community through forums and mailing lists, and help guide the evolution of the Jazz platform.

Still not sure?

Watch the [guided tour](#) of Jazz.net.



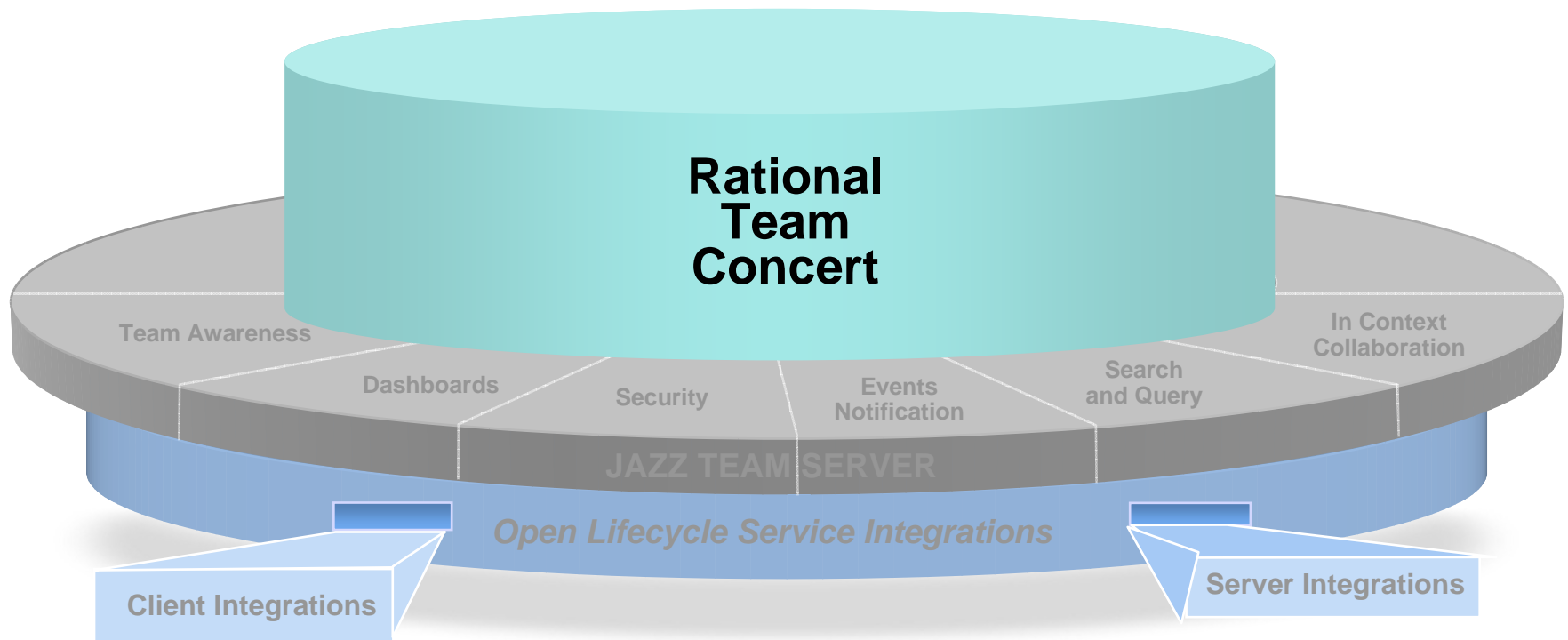
Agenda

- All that Jazz
- **Rational Team Concert**
 - ▶ **Overview**
 - ▶ Getting Started with a new Team
 - ▶ Software Configuration Management
 - ▶ Build Management
 - ▶ Change and Traceability



Today's focus: Rational Team Concert

The first offering built on the Jazz technology platform



Rational. software



Software innovation through collaboration

- **Real time, in-context team collaboration**
 - ▶ Make software development more automated, transparent and predictive
- **"Think and work in unison"**
 - ▶ Integrated source control, work item and build management
- **Assess real-time project health**
 - ▶ Capture data automatically and unobtrusively
- **Automate best practices**
 - ▶ Dynamic processes accelerate team workflow
 - ▶ Out-of-the-box or custom processes
- **Unify software teams**
 - ▶ Integrate a broad array of tools and clients
 - ▶ Extend the value of ClearQuest and ClearCase
 - ▶ Visual Studio Client (4Q08)
 - ▶ Support for System z and System i servers (2H08)

Open and extensible on



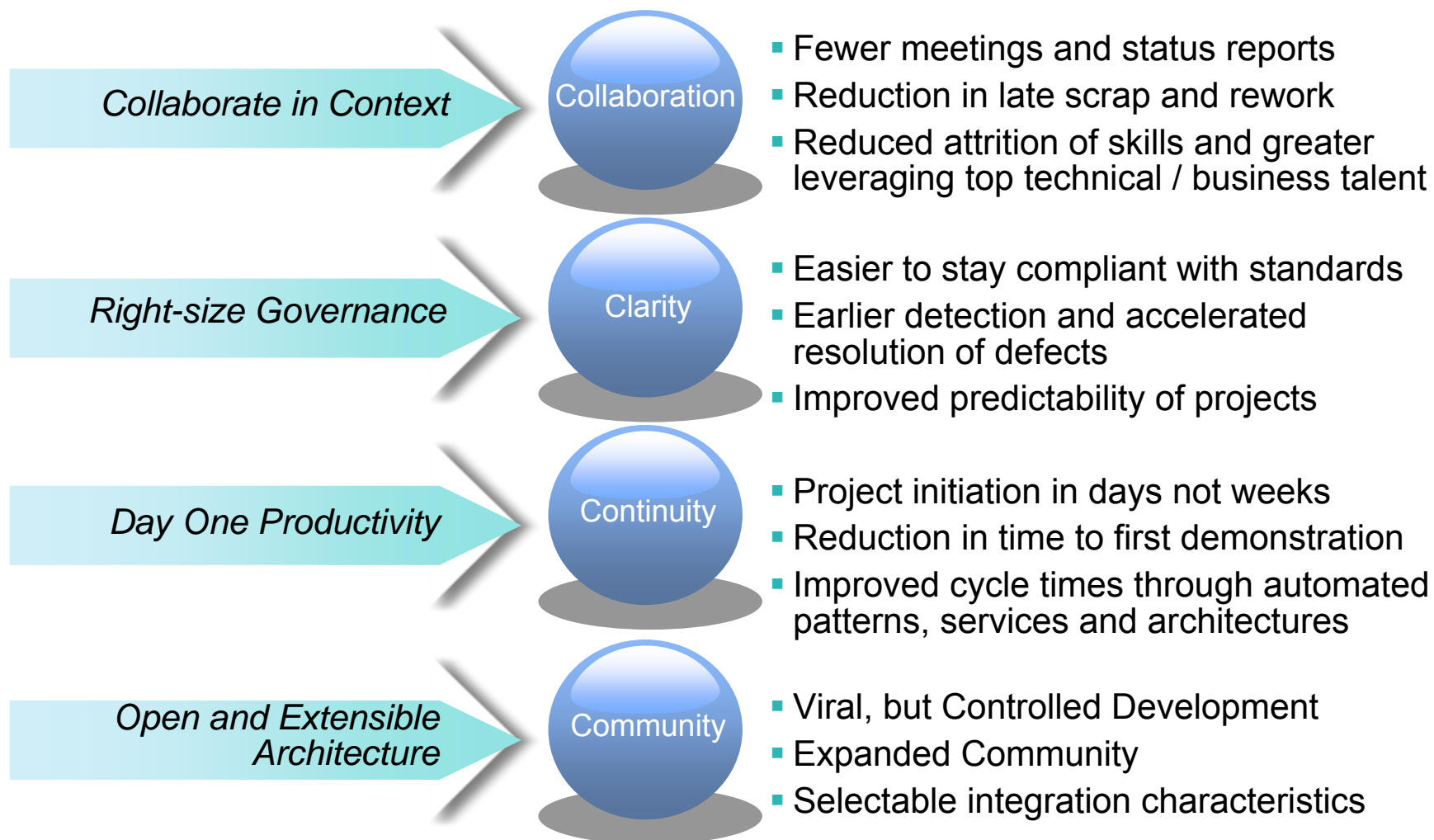
- ✓ Collaborate in context
- ✓ Right-size governance
- ✓ Day one productivity

IBM Rational Team Concert

transparent *integrated presence*
 wikis OPEN real-time reporting
 chat automated hand-offs Web 2.0
custom dashboards automated data gathering
EXTENSIBILITY *Eclipse plug-ins* services
 architecture **FREEDOM TO CREATE**



Translating words to actions



The Rational Team Concert family 2008

Statement of Direction (2009)

Enterprise

Enterprise Deployments

- Enterprise optimized extensions
- Integrated ClearCase, ClearQuest Build Forge
- Enterprise scale, security, governance & provisioning

Standard

Corporate Teams

- Customizable process
- Real-time project health
- LDAP admin
- Enterprise scale and extensibility
- Up to 250 users per server

Express

Mid-sized and Departmental

- Transparent development
- Out-of-the-box processes
- Commercial middleware
- Up to 50 users per server

Express-C

Community and Academia

- In-context collaboration
- Open source middleware
- Up to 10 users per server

Eclipse™, Linux®, MS Windows®, UNIX® - z/OS® (1H 09) & i5/OS (3Q) w/ native asset & work item mgmnt



Rational Team Concert

Iteration Planning

- Integrated iteration planning and execution
- Task estimation linked to key milestones
- Out of the box agile process templates

Project Transparency

- Customizable web based dashboards
- Real time metrics and reports
- Project milestone tracking and status

SCM

- Integrated stream management with flow relationships
- Component level baselines
- Server-based sandboxes
- Identifies component in streams and available baselines
- ClearCase connector

Work Items

- Defects, enhancements and conversations
- Query results view and share queries with team or member
- Support for approvals and discussions
- ClearQuest connector
- Query editor interface

Build

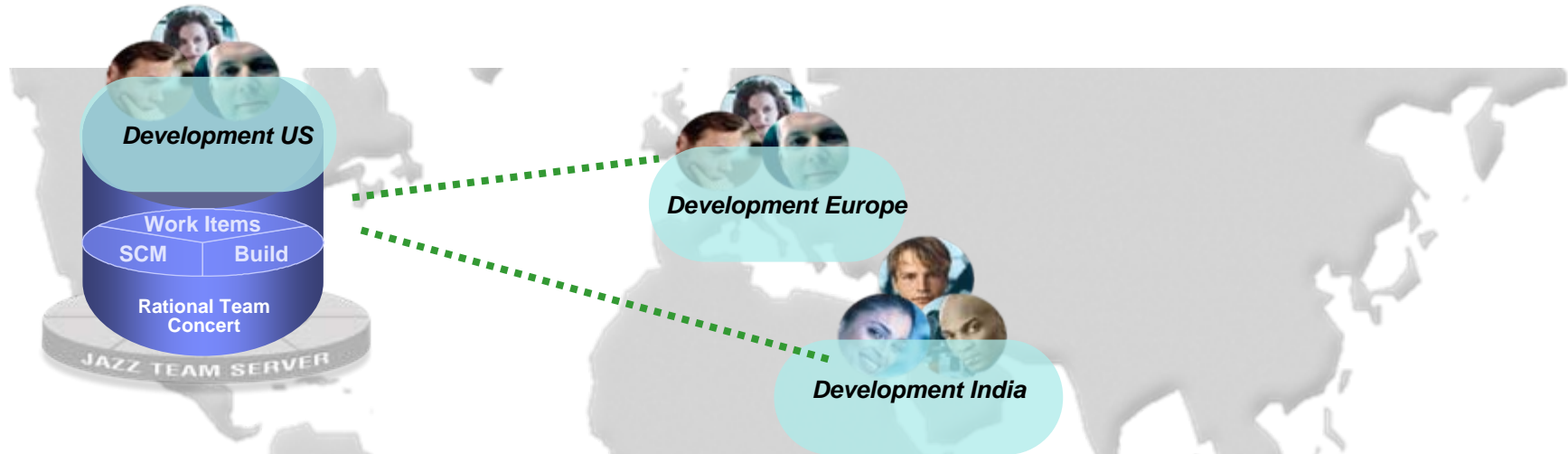
- Work item and change set traceability
- Local or remote build servers
- Supports ant and command line tools
- Integration with build forge
- Build definitions for team and private builds

Jazz Team Server

- Single structure for project related artifacts
- World-class team on-boarding / offboarding including team membership, sub-teams and project inheritance
- Role-based operational control for flexible definition of process and capabilities
- Team advisor for defining / refining “rules” and enabling continuous improvement
- Process enactment and enforcement
- In-context collaboration shows team members and status of their work



Leveraging Rational Team Concert independently

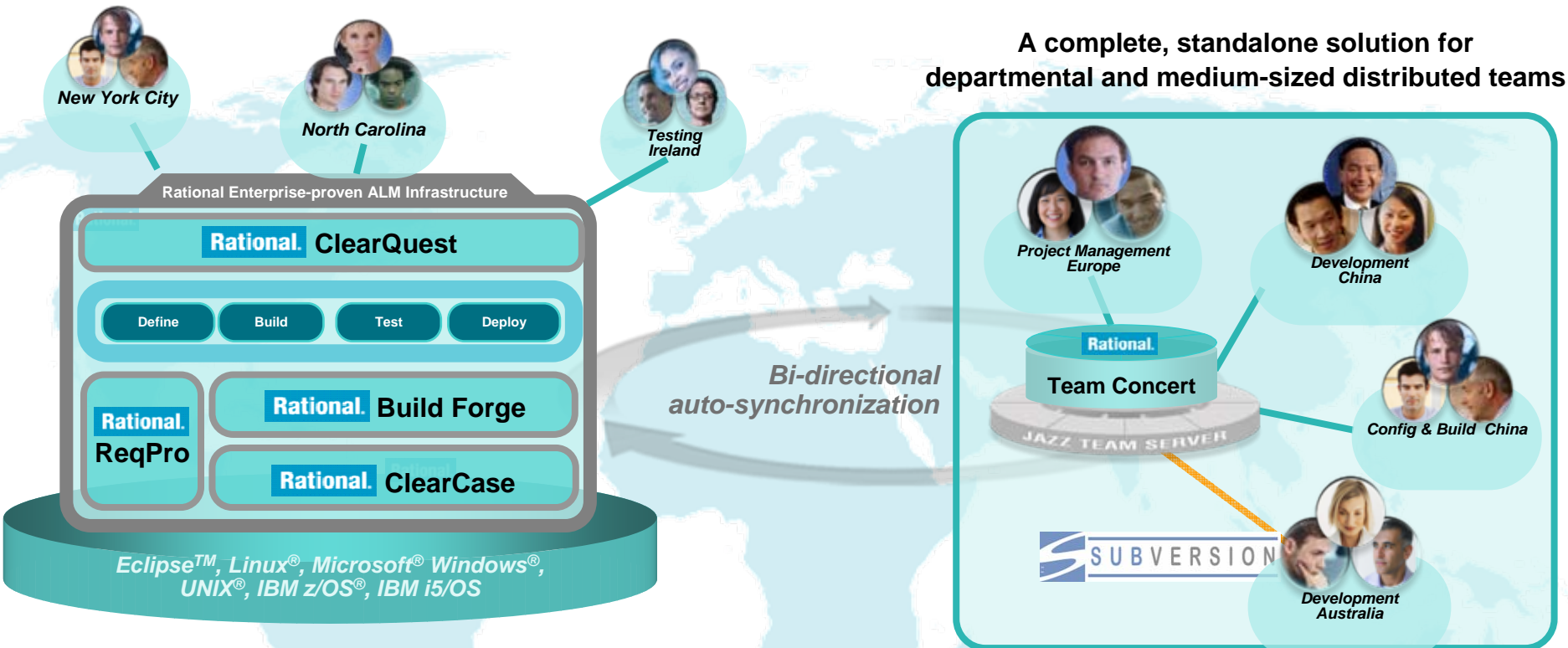


✓ Rational Team Concert Standard, Express, Express-C

- ✓ A standalone development environment optimized for small and mid-sized teams
- ✓ All the collaborative capabilities of the Jazz platform – plus integrated work items, SCM and build management
- ✓ Dashboards and real-time reports
- ✓ Team and Process-aware



Unify distributed teams in mid-sized or enterprise settings



- Supports departmental “satellite” or “agile” projects optionally integrated with existing enterprise-proven Rational ALM infrastructure offerings
- Enhances existing ALM offerings with new collaborative & process capabilities



Source code management

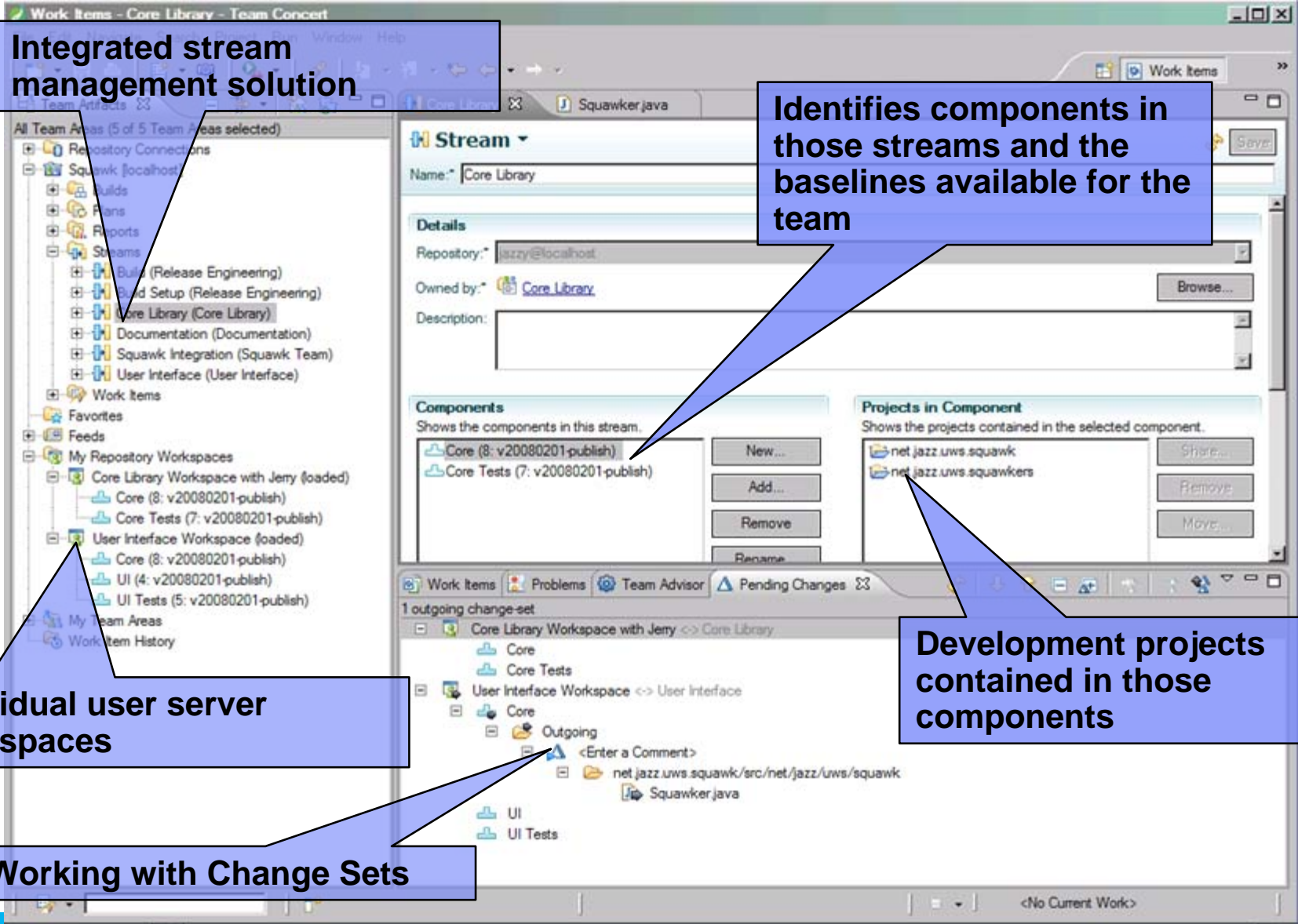
Integrated stream management solution

Identifies components in those streams and the baselines available for the team

Individual user server workspaces

Working with Change Sets

Development projects contained in those components



Work items

Subscribe to work items you're interested in

Predefined, custom and personal queries

Integrated discussion threads

Query results

Understands and persists work item's relationship to SCM and build artifacts

The screenshot displays the IBM Rational Work Items application. The main window shows a defect titled "Defect 12" with the summary "UWS Add JavaDoc to CLI code". The defect is in a "Resolved" state and is owned by "April Blues". The description states: "We need to add JavaDoc to our code. Note that I have sized this at one day and targeted it for M1. This means do it now while Zach gets the build going but only spend at most this much time on it, maybe less. We just need to get a start on it now." The discussion section contains a comment: "We have decided to make the change|".

At the bottom of the application, a query results table is visible, showing the following data:

I	Status	P	S	Summary	Owned By	Created By
12	Resolved			UWS Add JavaDoc to CLI code	April Blues	Jerry Jazz
10	Resolved			Expose functionality via public get method to core temperature ...	Unassigned	Jerry Jazz



Builds

Create build definitions for team and private builds

Supports Ant and command line build tools

Create local or remote build servers

Identify work items and change sets that went into the build

Historical view of the build queue with status

The screenshot displays the IBM Rational Build team interface. On the left, a tree view shows 'Team Artifacts' with 'Builds' expanded to show various build definitions like 'workshop.squawk.core.continuous.build'. The main area shows a detailed view of a completed build: 'Build workshop.squawk.core.continuous.build B20080303-1042-workshop.squawk.core.continuous.build'. It indicates the build is 'Completed' with a duration of 27 seconds, starting at 1:41:40 PM and ending at 1:42:07 PM on March 3, 2008. A status trend bar shows a mix of green and red segments. Below this, a 'Contribution Summary' lists changes, logs, repository workspace, snapshot, and work items. To the right, 'General Information' includes the requested by (Zach Builder), build definition, engine, and history. At the bottom, a 'Summary' tab shows a table of build activities.

Build	Label	Progress	Estimated Completion	Start Time	Duration	Tags
workshop.squawk.core.c...	B20080310-2231...	Completed		March 10, 2008 10:30...	34 seconds	
workshop.squawk.core.c...	B20080305-2325...	Completed		March 5, 2008 11:25...	19 seconds	
workshop.squawk.core.c...	B20080305-2324...	Completed		March 5, 2008 11:24...	33 seconds	
workshop.squawk.core.c...	B20080303-1057...	Completed		March 3, 2008 1:57:0...	24 seconds	
workshop.squawk.core.c...	B20080303-1048...	Completed		March 3, 2008 1:48:1...	28 seconds	
workshop.squawk.core.c...	B20080303-1042...	Completed		March 3, 2008 1:41:4...	27 seconds	
workshop.squawk.core.c...	20080303-1017...	Completed		March 3, 2008 1:17:5...	26 seconds	
workshop.squawk.core.c...	B20080201-1707...	Completed		February 1, 2008 8:06...	30 seconds	v20...
workshop.squawk.core.c...	B20080128-1359...	Completed		January 28, 2008 4:59...	39 seconds	wor...
workshop.squawk.core.c...	B20080123-1623...	Completed		January 23, 2008 7:23...	10 seconds	
workshop.squawk.core.c...	B20080123-1621...	Completed		January 23, 2008 7:21...	18 seconds	



Enables process awareness and guidance

Team Advisor

- Rules can be run when delivering changes to enforce team or organizational standards
- Helps ensure higher quality results through enforcement of agreed-upon standards
- Rules are configurable
- “Quick Fixes” can be specified to simplify corrective action
- Process rules can be defined, refined “on the fly”, enabling continual improvements
- Out of the box processes include OpenUP, Scrum, The Eclipse Way, etc

The screenshot displays the Team Advisor interface. On the left, a tree view shows various Eclipse Way Disciplines and Work Products. The main area shows a list of rules under the heading 'Use this Quick Fix'. The 'Detailed F' view is open, showing a tree of rules for 'Deliver (1 error) [client] 11:38:49 A'. The rules listed are 'No Unused Imports', 'Clean Workspace', and 'Descriptive Change Sets'. The 'Descriptive Change Sets' rule is expanded, showing a sub-rule '<No Description>'. A red error icon is visible next to the error message: 'Change set missing required comment or work item.' Below the error message, a 'Quick Fix:' section lists three options: 'Associate Existing Work Item', 'Associate New Work Item', and 'Associate and Try Again (experimental)'.



Iteration Planning

Understand how well you are progressing against your targets in real-time

UWS Temperature Conversion M1 Plan [1.0 M1]

UWS Temperature Conversion M1 Plan

Team Area: UWS Temperature Conversion Team | Iteration: 1.0 M1 (12/1/07 - 6/20/08) | 5 Closed | 9 Open

Owner	Task	Progress	Estimated	Time	Status	Count
April Blues	UWS Create the temperature conversion CLI package	1 / 17 -15 h	100%	2 days	Unassigned	9
Derek Holt	UWS Define permissions	0 / 8 -7 h	100%	4 hours	Unassigned	5
Jerry Jazz	UWS Define team members	0 / 4 -3 h	100%	4 hours	Unassigned	6
Zach Builder	UWS Create the core temperature conversion package	25 / 37 -10 h	100%	1 day	Unassigned	7
Zach Builder	UWS Define iterations/milestones	0 / 3 0 h	100%	4 hours	Unassigned	3
Zara Intern	UWS Add JavaDoc to core temperature conversion JUnit tests	0 / 17 -16 h	100%	1 hour	Unassigned	11
Zara Intern	UWS Create the core temperature conversion package JUnit tests	0 / 8 0 h	100%	2 days	Unassigned	8
Unassigned	Expose functionality via public get method to core temperature conversion code	8 / 9 h	100%	1 hour	High	23

Group by: Owner

Sort By: Priority

Bars: Progress

Exclude: Future and past items, Resolved items

More Filters... Tags...

Related Work Items: Show Backlog, Unplanned, Closed Items, Next Plans

Overview | Planned

Drag-and-drop work items to change owners/create child parent relationships

Plan and execute iterations while managing team and individual load



In-context collaboration

Team Awareness

Shows team members and their online status

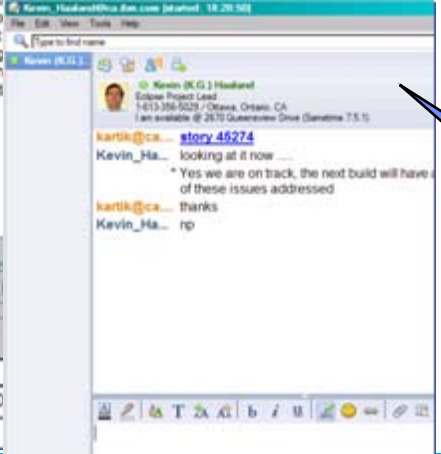
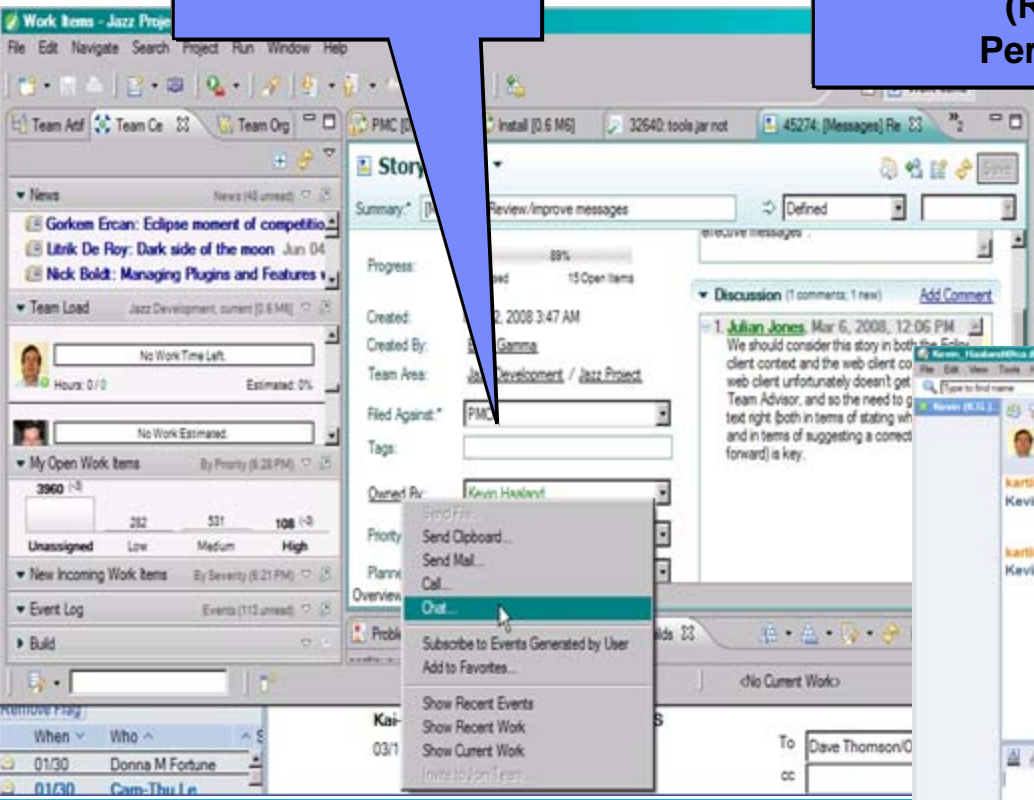
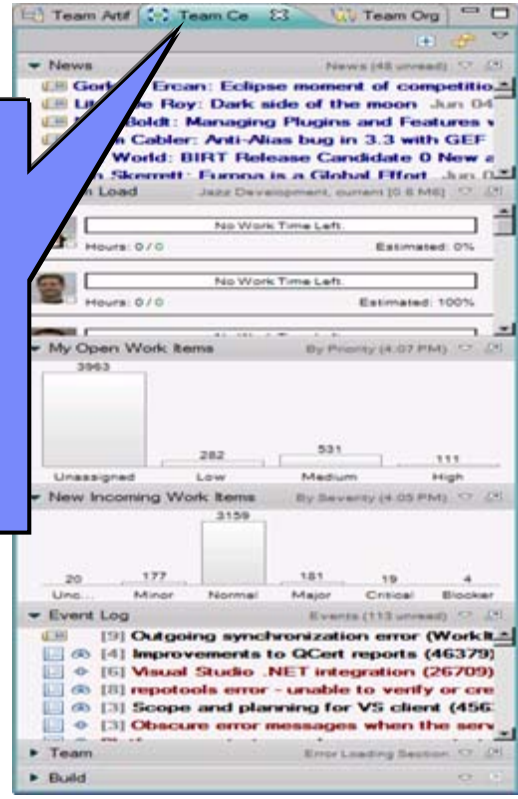
Shows what they are working on

Team Central

- News & events
- Build status
- What's being worked on
- Changes

Configurable (RSS feeds)

Personalizable



Collaborate in Context



Dashboards

Transparency, collaboration and control via customizable dashboards

The screenshot shows a 'Development' dashboard with several sections:

- General** (dropdown menu)
- Builds | Open Defects by Themes | Open Defects by Team | SVT Defects by Team | Defect Trends by Team | Deferred Items by Team
- Endgame Has Started** (Warning icon): Detailed Plan available here: End Game Plan
- Jazz Development Description**: Umbrella area for all Jazz development work. Development work is done in a separate development line named 'development'. Each release is performed in its own project phase. Each component team that actively contributes to the development work is represented with its own child team area. The PMC oversees the development work. Team members listed in this team area are working across all component boundaries.
- Open Track Build Items (1)**: 4841 RC1 status
- Current Development Plans (33)**: Shows a list of components with progress bars. Components include CQ Connector, Component Development, Foundation & Work Items, Install, PMC, Process, Release Engineering, Repository, SCM, UI Design, User Assistance, Web Dashboards, and Web UI. Some have 'No planned work' or 'No estimated work'.
- Risks (10)**:
 - 48863 [Risk] Migrating/importing SVN repositories
 - 48865 [Risk] Migrating from an existing bug tracking system
 - 48868 [Risk] CC Connector
 - 48862 [Risk] We need to build what we ship
 - 48871 [Risk] CQ Connector Set-up
 - 48875 [Risk] Scale to 250 users
 - 48870 [Risk] Oracle Support
 - 48876 [Risk] Scale towards large repositories
 - 48874 [Risk] Derby Support
 - 48872 [Risk] Active Directory
- RFS (28)**:
 - 9619 [RFS] Monitoring a continuous build
 - 9629 [RFS] Create iteration plan, assign work, tracking progress
 - 26827 [RFS] Getting Started Experience
 - 11321 [RFS] Install and configure a Jazz server
 - 9631 [RFS] Process governed work item customization
 - 9625 [RFS] Starting a Project
 - 9627 [RFS] Joining a team
 - 9623 [RFS] Browsing the history of changes
 - 9632 [RFS] Duplicate handling
 - 9908 [RFS] Managing Team Artifacts
- Jazz Development Event Log (2 new)**:
 - Test scenario: Process author creates/refines a Process Template (51513) Now
 - Test scenario: Customer hears about Jazz and installs it (51492) 15 minutes ago

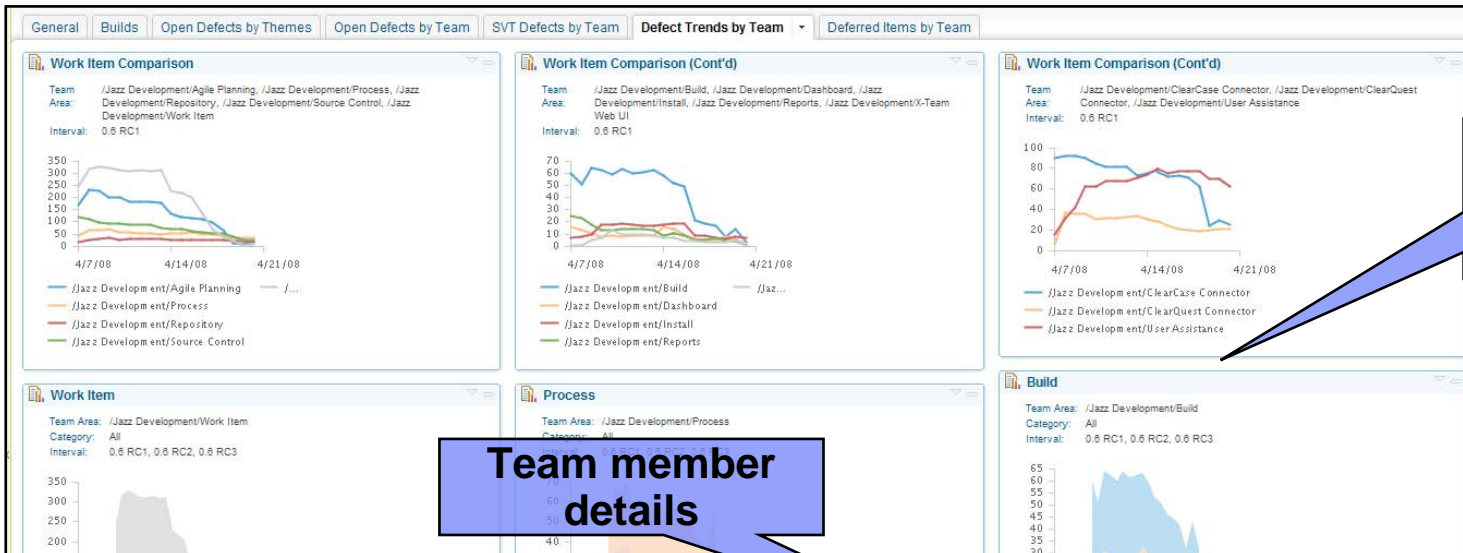
Clearly understand team goals

Risks, issues, challenges surfaced at both the team and project level

Real-time status



Dashboards and reporting



Trending by project or by individual team

Team member details

Current milestone status

This section of the dashboard provides detailed information about work items and team members. It includes a 'Work Item Members' list with names and roles, 'Current Work Item Plans' showing progress bars for different iterations, and a 'Work Item Event Log' with recent activity.

Name	Roles
André Weinand	componentlead, contributor, integrationstreamadmin, integrationbuildmeister
Benjamin Pasero	contributor
Christof Marti	contributor, integrationbuildmeister
Dirk Baeumer	parttime, contributor, integrationstreamadmin, integrationbuildmeister
Erich Gamma	contributor, newcomer, integrationstreamadmin, integrationbuildmeister, dashboardadmin, projectadmin, PMC
Frank Lyner	contributor
Johannes Rieken	parttime, contributor, integrationbuildmeister
Larry Smith	contributor, integrationbuildmeister
Marcel Bihl	contributor
Mark Buquor	newcomer, parttime

Current Work Item Plans (2)
 Current Iteration: 0.6 RC1
 Foundation & Work Items: [Progress bar]

Work Item Event Log (15 new)

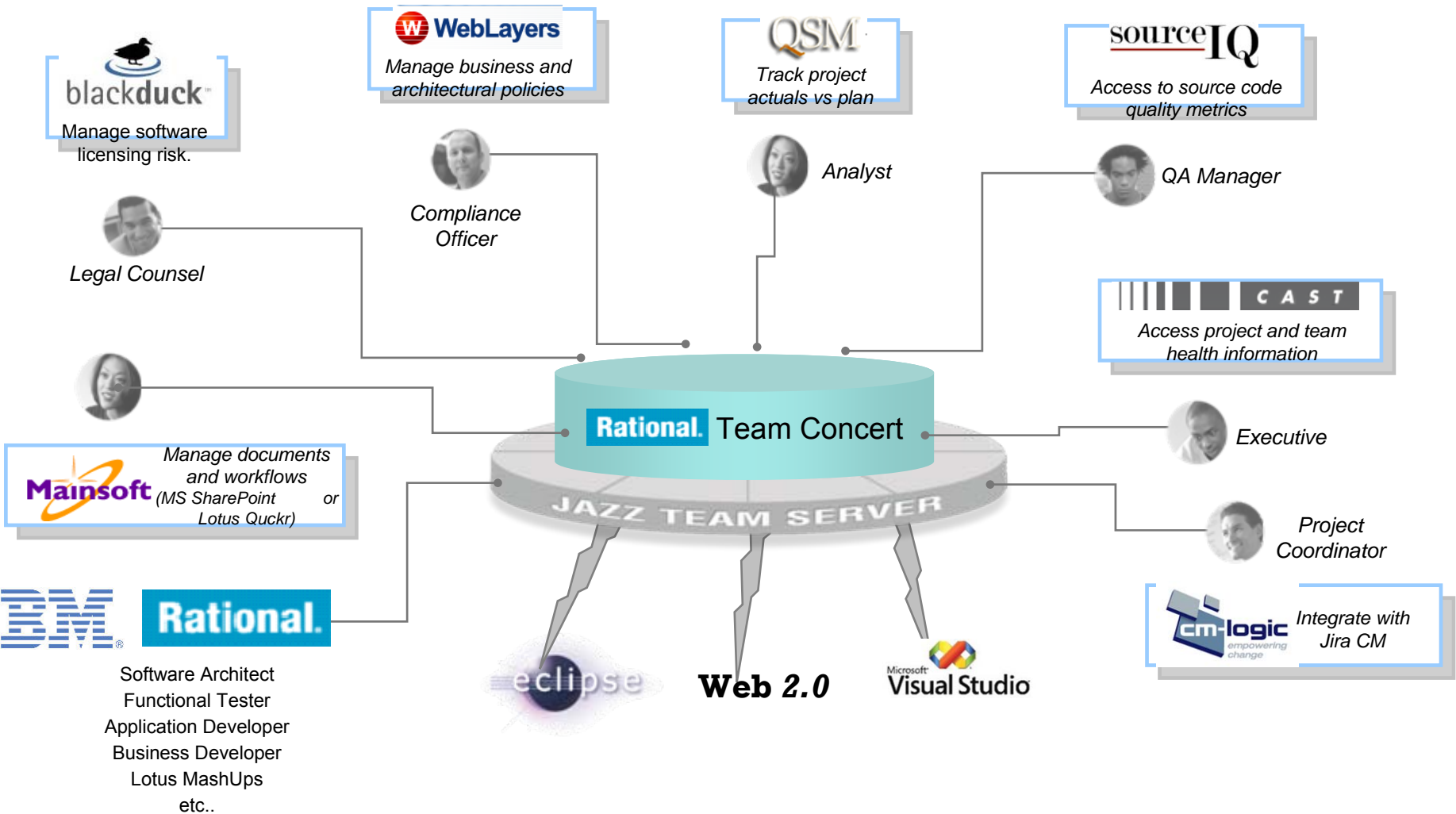
- WebUI - Hide comment is confusing - lost work (51725) 10 minutes ago
- Indexing slow on attachment with long lines (48806) 15 minutes ago
- User photo lost when creating new user (49910) 15 minutes ago
- [3] Deliver new csv icon (52856) 15 minutes ago

Page 1 of 2

Work Item Builds
 Succeeded: continuous.workitem.jazz.C20080424-0748 10 minutes ago



Extend team collaboration to a broad range of software professionals



DEMO

- Introduction to Rational Team Concert



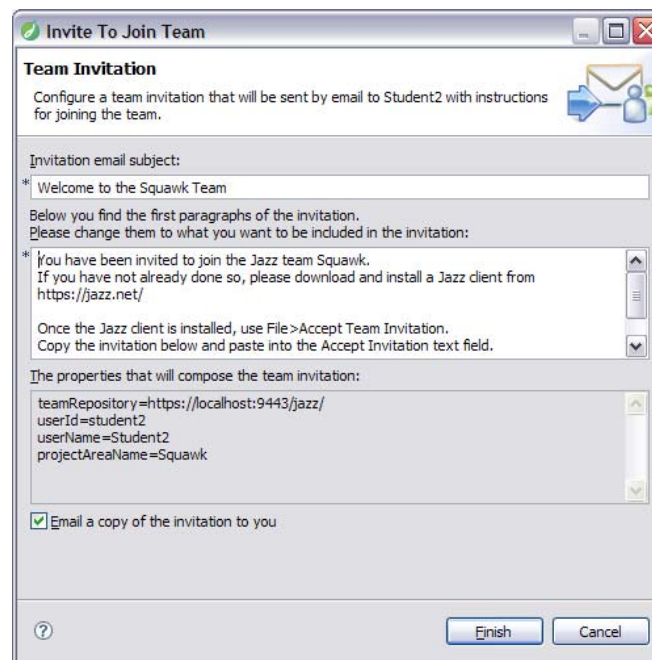
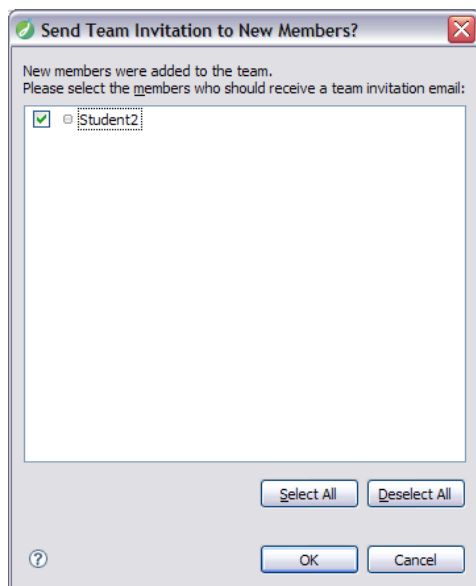
Agenda

- All that Jazz
- **Rational Team Concert**
 - ▶ Overview
 - ▶ **Getting Started with a new Team**
 - ▶ Software Configuration Management
 - ▶ Build Management
 - ▶ Change and Traceability



Joining a Team

- For most environments, joining a project can be complicated
- Team Concert makes this as easy as possible
- Adding a new team member to a project generates a Team Invitation email
- Contents of the email can be used to set up the new team member's access to the project resources in Team Concert



Communication

- Users of Team Concert can use a variety of tools to communicate with team members
 - ▶ E-mail
 - ▶ Instant Messaging/Chat
 - ▶ RSS feeds
 - ▶ Web UI
 - ▶ Team Concert client
- Team members can use all the typical communication mechanisms to keep working together as a team, regardless of where they are physically located. This collaboration allows for a single view of project data
 - ▶ Integrated Instant Messaging/Chat for immediate feedback
 - ▶ RSS feeds to notify you of significant events on the project in real time
 - ▶ The Web UI used for anyone on the team, or who has an interest in the project



DEMO

- Joining a new project
 - ▶ Connect to project
 - ▶ Network with new team



Agenda

- All that Jazz
- **Rational Team Concert**
 - ▶ Overview
 - ▶ Getting Started with a new Team
 - ▶ **Software Configuration Management**
 - ▶ Build Management
 - ▶ Change and Traceability



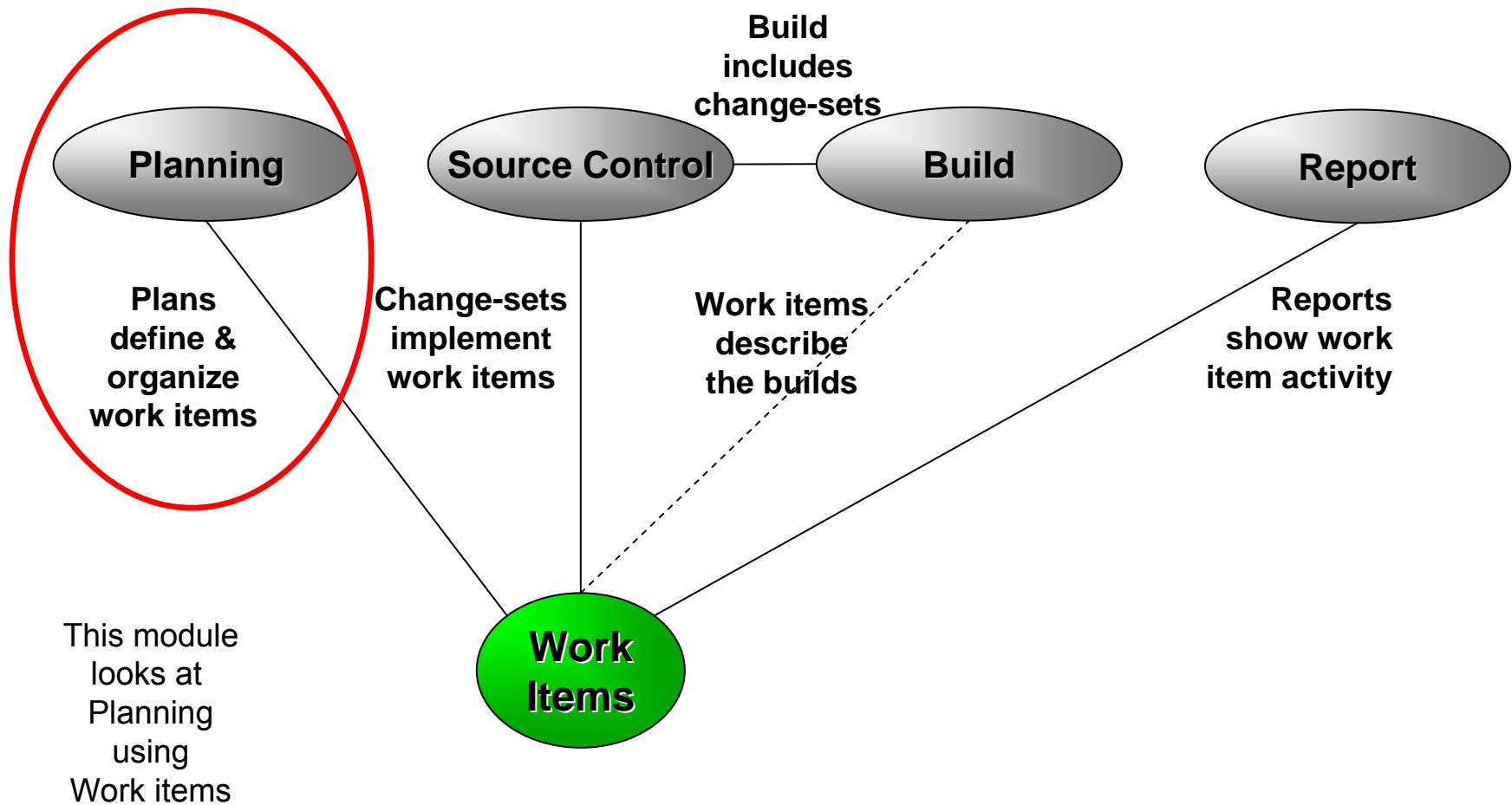
Units of work

- Rational Team Concert allows you to divide your work into **work items**
- The set of work item types is **open-ended**
- Standard types: **Task, Enhancement, Defect**
- The set is **defined by each team**
- The work item **life cycle** is **configurable**
- All work items are **stored** in the **repository**

- The way that Rational Team Concert uses **Work items** means that all your work can be planned, tracked and stored under configuration management. Allowing for both the top down view and the low level view of data and project status.



Work Items are central to Rational Team Concert



Work Item details

37: Create a Lion Squawker

Enhancement 37

Summary: * Create a Lion Squawker

New

Details

Type: Enhancement

Severity: Normal

Found In: Unassigned

Created: 04-Apr-2008 17:20

Created By: student1

Team Area: Core Library / Squawk

Filed Against: * Squawk/Core

Tags:

Owned By: student1

Priority: High

Planned For: -> 1.0 M2

Estimate: 20 m Correction:

Time Spent:

Due: None

Quick Information

<No information>

Description

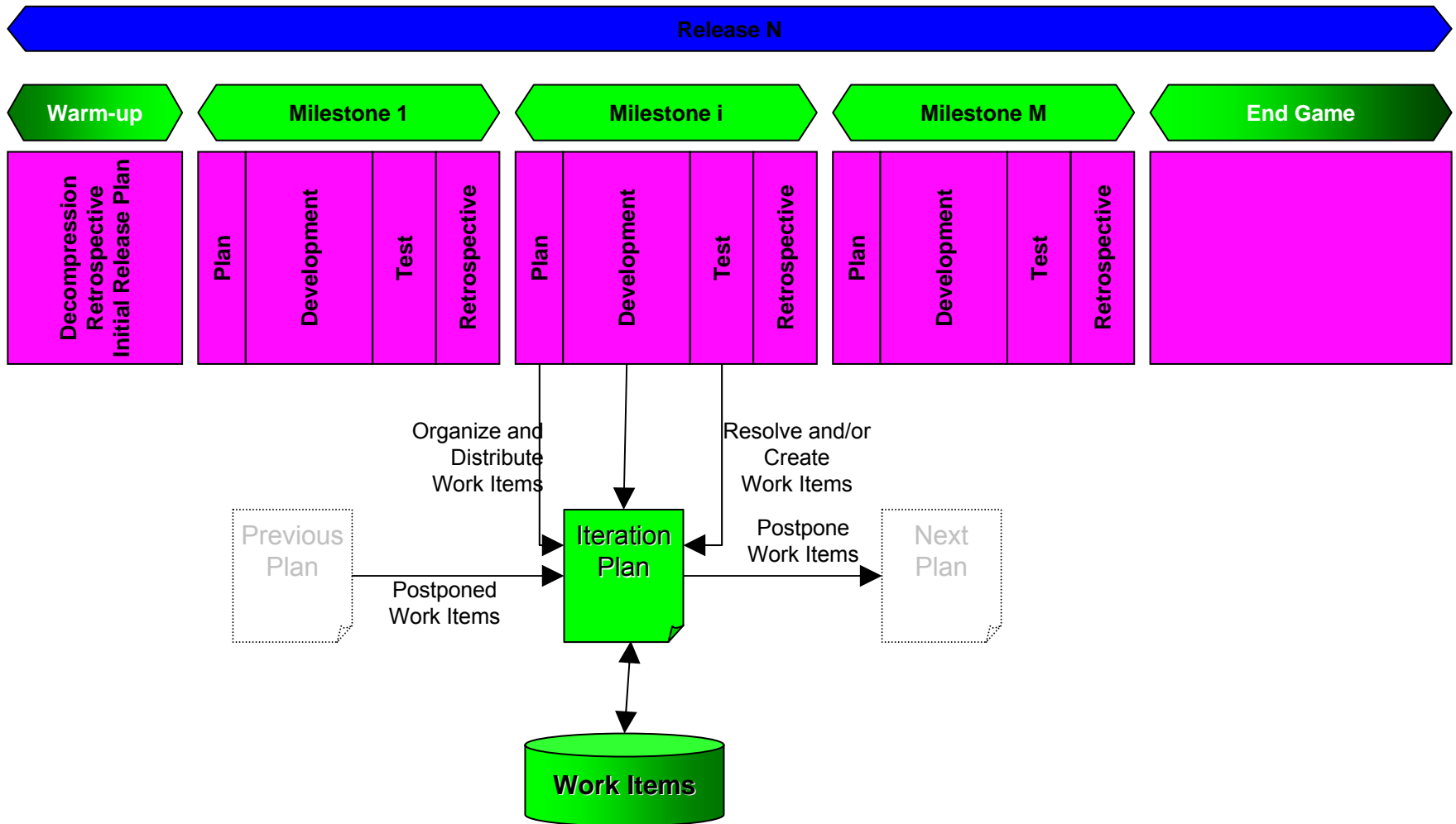
Create a Lion Squawker

Discussion [Add Comment](#)

Overview Links Approvals History



Iteration planning in Rational Team Concert



The Iteration Plan

- A collection of work items...
 - ▶ Assigned to a given milestone
 - ▶ For a given team
- Plans are live
 - ▶ Changing work items changes the plan
 - ▶ Changing the plan directly changes the work items
 - ▶ Create new work items from the plan
- Plan structure is dynamic
 - ▶ Easily be grouped by owner, category, duration, priority, etc.
- Plans are visible
 - ▶ Available to everybody on the team
 - ▶ Observable by interested outsiders



Work Item Queries

- Provides real-time project health information and transparency of status through automated data gathering.
- Rational Team Concert provides a query mechanism to find work items in a Project Area allowing for more project transparency.
 - ▶ The query scope for work items is the project area.
- The user interface includes
 - ▶ an editor for building structured work item queries
 - ▶ an end-user configurable work item view to browse the query results.



DEMO

- Planning your work
 - ▶ Create a new Enhancement
 - ▶ Create a new Iteration



Real time collaboration

- The modules before showed how the project team plans the work for an iteration.
 - ▶ But how does the project keep track of all the planned work items?
 - ▶ How do I see who may help me with my actual problem?
 - ▶ How do I get the most recent status of the project?

What if your tool knows the actual status of your team's work?

Rational Team Concert stores all artifacts for the development project in one repository and provides powerful query capabilities to retrieve and display data.



Real time collaboration

- Query and Reporting capabilities are available in the Eclipse Client and the Web UI.
 - ▶ Define queries on Work Items to find your work and the work of others.
 - ▶ See who is online and ready to collaborate with you.
 - ▶ See the event log for build or work item events that are interesting to you and follow RSS feeds for News.
 - ▶ Generate, display and export reports on the status and health of the project.
- Rational Team Concert displays the information in automatically refreshed views that are configurable, so that you are up to date with the information you need in real time.



Reports – Displaying the project status

- Rational Team Concert uses the BIRT* reporting engine
- A huge variety of report formats are designed and available to display an actual overview of your projects:
 - ▶ Reports for the health of your builds
 - ▶ Reports for viewing the team load and the distribution of work items
 - ▶ Reports for your code
 - ▶ Etc.
- Reports can be arranged in the Web UI to Dashboards
- Reports can be exported to: .pdf, .xls, .doc, .ppt Formats

*BIRT is an open source Eclipse-based reporting system that integrates with your Java/J2EE application to produce compelling reports.



Reports – Displaying the project status

You are logged in as Robin Bater | [Logout](#)
The Jazz Project

Home | **Dashboards** | Work Items | Iteration Plans | Reports

Create New Dashboard | Find Dashboard

Shared Dashboards
My Profile
Jazz Project

Jazz Project Dashboard

Home | Trend Reports | Feeds

M6 Started

Current iteration changed to M6.

1 2

Welcome

Welcome to the Jazz Project dashboard. You can visit this space often to monitor all the vital signs of the Jazz Project at a glance, fed by the live data from the repository.

Jazz Project Description

Process enabled team collaboration platform

The goal of the Jazz project is to build a scalable, extensible team collaboration platform for seamlessly integrating tasks across the software lifecycle. The new technology is called the Jazz Platform. The Jazz project is defining the overall architecture of the Jazz Platform, and building an initial set of Jazz components that plug in to the Jazz Platform.

The work in the Jazz project is divided into development work and maintenance work. Development work produces a consecutive line of releases. Maintenance work produces bug fix releases of the released produced by the development work.

Development as well as maintenance work is overseen by the PMC. The team members listed in this project area constitute the PMC.

Server Status

Database: **Connected**
 Services: **Ok**
 Memory: 28% free of 982MB allocated / 1000MB Max
 Version: 0.6.0.I20080307-1427
 Uptime: 2 days, 2 hours

Jazz Project Teams (44)

- Jazz Development
- Jazz Maintenance

Useful Links

Wiki

- Planning
- Release Engineering
- Developer Setup
- Project Documents

Jazz Team Blog (10, 9 new)

See you at EclipseCon 2008 11 days ago
The Team Concert User Workshop: behind the scenes 14 days ago
Refining the Jazz user interface 18 days ago
A brief history of the Jazz Team Server interface: Our journey from a J2EE server towards a RESTful server 24 days ago
 Page 1 of 3



Reports – Displaying the project status

You are logged in as Robin Bater | [Logout](#)
The Jazz Project

Home | **Dashboards** | Work Items | Iteration Plans | Reports

Create New Dashboard | Find Dashboard

Shared Dashboards
My Profile
Jazz Project

Jazz Project Dashboard

Home | Trend Reports | Feeds

Recently modified (129) Tags

10.5(1) actionbuttons(2) admin(1) blank(1)
bundle(1) cli.discard(2) engine(1)
errorhandling(3) from_wi(1) fvt(2)
globalization(6) icon(1)
ignores(1) leftpanel(1) m2d4patch5(1)
m5patchcandidate(1) m6feature(2)
m8pmc(1) mac(1) nls(1) os(1)
performance(1) polish(2) production(1)
svt(4) svt_spt(4) usability(1)
ux_blueprint(1) verify(2)
webui_alignment(3)

Show More

Open Work Items by Type (All)

Team Area: All
Category: All
Interval: All

8000
6000
4000
2000
0

2/1/07 8/1/07 2/1/08

com.ibm.team.apr.common.wo...
Plan Item
Enhancement
Task

Open vs Closed Work Items (All)

Team Area: All
Category: All
Interval: All

35000
30000
25000
20000
15000
10000
5000
0

2/1/07 2/1/08 2/1/09

Open Closed

Open Work Items by Priority (All)

Team Area: All
Category: All
Interval: All

8000
6000
4000
2000
0

2/1/07 8/1/07 2/1/08

Unassigned
Low
Medium
High

Blocking Work Items (All)

Team Area: All
Category: All
Interval: All

16
14
12
10
8
6
4
2
0

2/1/07 8/1/07 2/1/08

New Work Items by Severity (All)

Team Area: All
Category: All
Interval: All

700
600
500
400
300
200
100
0

6/6/06 1/1/07 5/25/07 11/1/07

Unknown Major
Unclassified Critical
Minor Blocker
Normal

Closed Work Items by Priority (All)

Team Area: All
Category: All

Display your choice of reports and queries in your own dashboard, e.g. to control the flow of work items.



Reports – Displaying the project status

jazz
You are logged in as **Robin Bater** | [Logout](#)

Home | **Dashboards** | Work Items | Iteration Plans | **Reports**

My Reports

Shared Reports

- Build
 - Build Health (Continuous)
 - Build Health (Weekly)
 - Code Coverage
 - Code Health
 - Frequent Test Failures
 - Frequent Test Regressions
- Data Warehouse
 - Data Warehouse Metrics
- Repository
 - Integral Range Statistics
 - Latest Metrics
 - Latest Metrics by Namespace
 - Metrics
 - Metrics by Namespace
 - Range Statistics
 - Scalar Statistics
- Source Control
 - Project Activity
 - Stream Size Counts
- Work Items
 - Blocking
 - By Owner (Live)
 - By Priority
 - By Team Area (Live)
 - Closed Daily by Severity
 - Comparison
 - Defect Backlog
 - Defect Discovery Rate
 - Deferred Work Items
 - New by Severity
 - Open by Type
 - Open vs Closed

Build Health (Weekly) Save As... Save

Parameters

Team Area: /Jazz Development	Build Definition: weekly.integration.jazz	Duration: Last Month
Include Incomplete Builds: No		

Build health

Latest Build is Clean (Mar 4, 2008 3:41:30 PM)

● Build
— Number of Tests
Clean Build Range
Failed Build Range

The build was clean during 67% of this time.
34% of builds (13 of 38) were clean in the specified time.

Build duration

For a detailed Report open the Reports page and choose from a variety of available reports

Jazz/RTC Tech Event

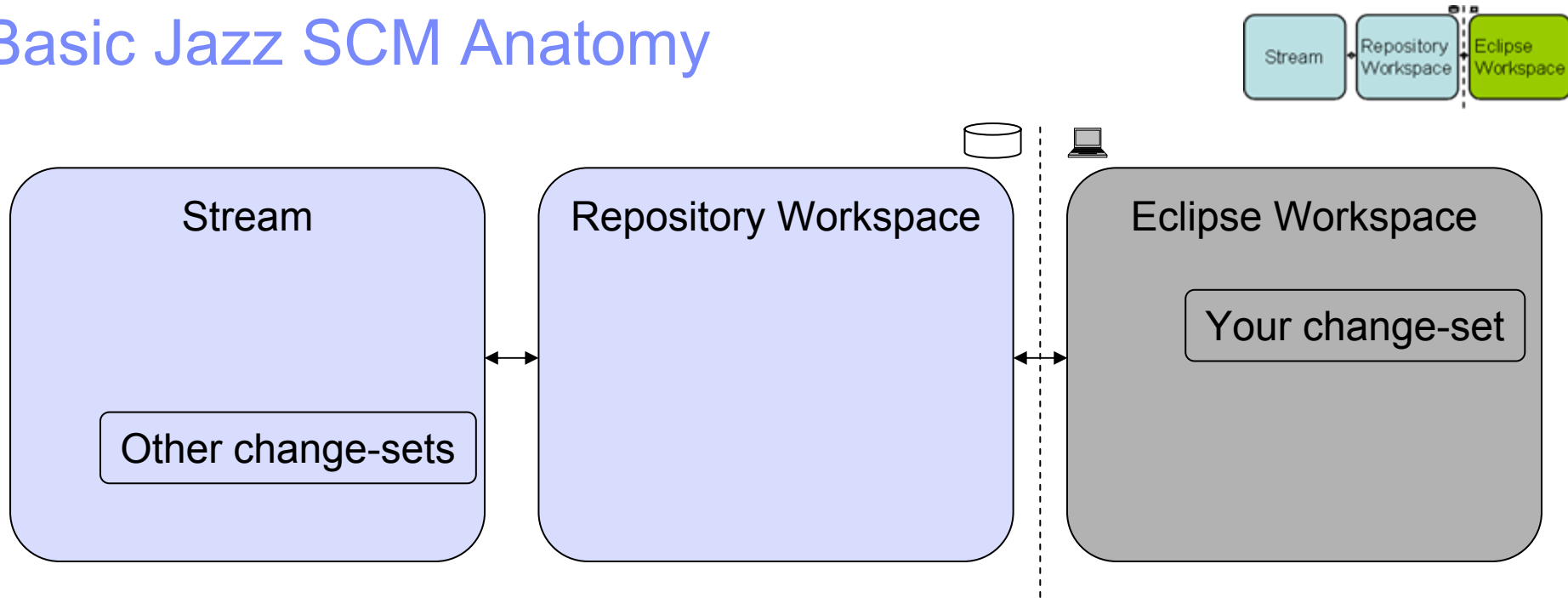
41

Agenda

- All that Jazz
- **Rational Team Concert**
 - ▶ Overview
 - ▶ Getting Started with a new Team
 - ▶ **Source Configuration Management**
 - ▶ Build Management
 - ▶ Change and Traceability



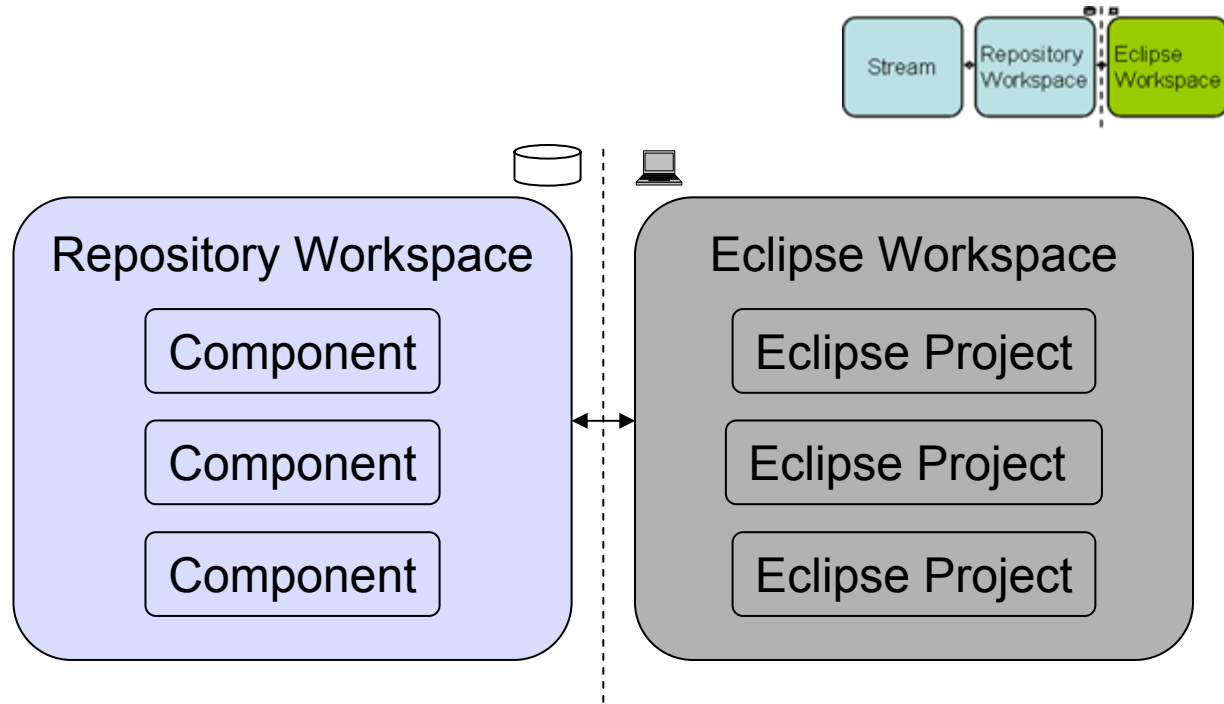
Basic Jazz SCM Anatomy



- Streams are for sharing
- Repository workspaces are your personal space
- Eclipse workspaces are where you work
- Change-sets flow back and forth
- Subsequent modules will cover each of these



Components...

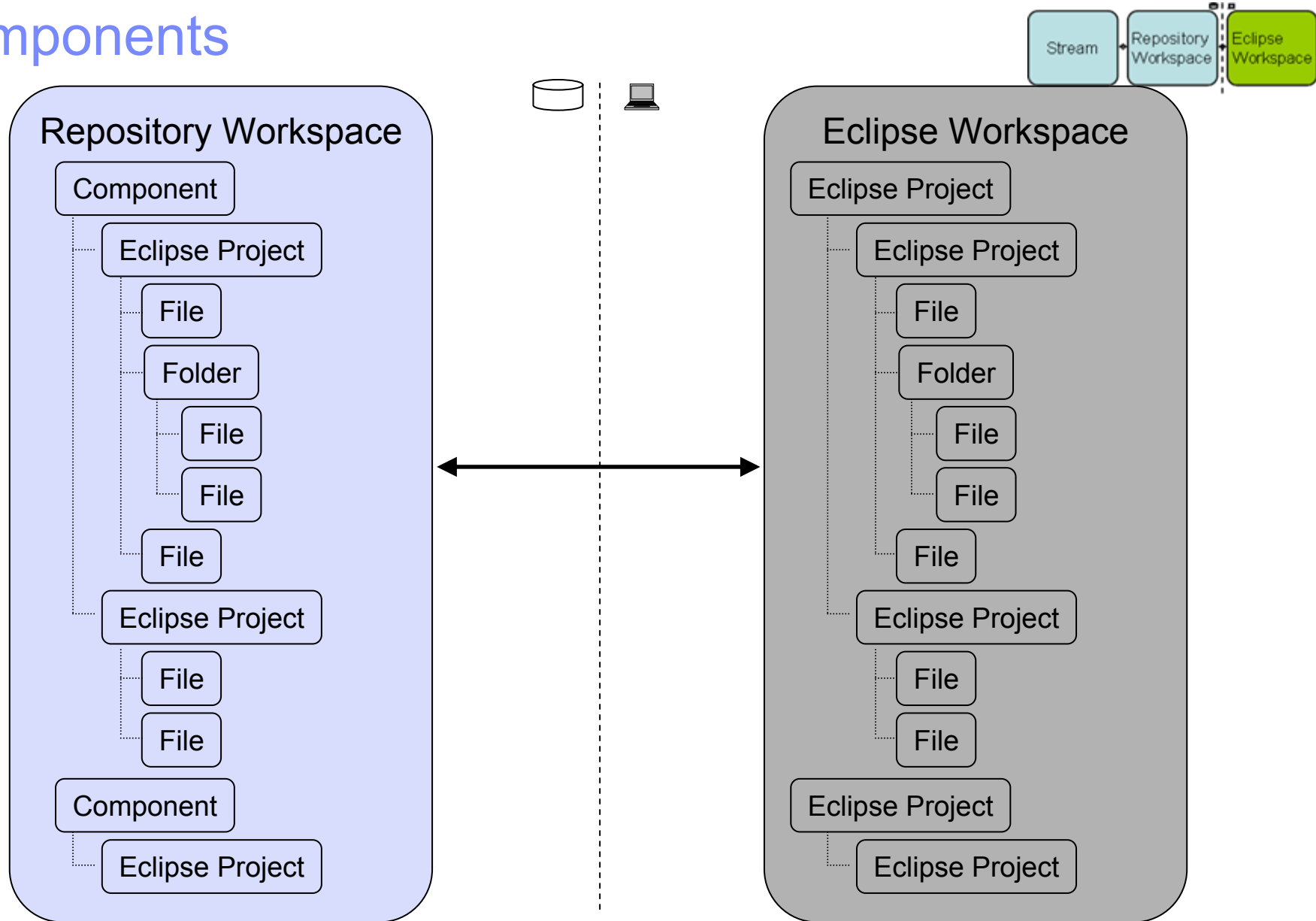


- Repository Workspaces

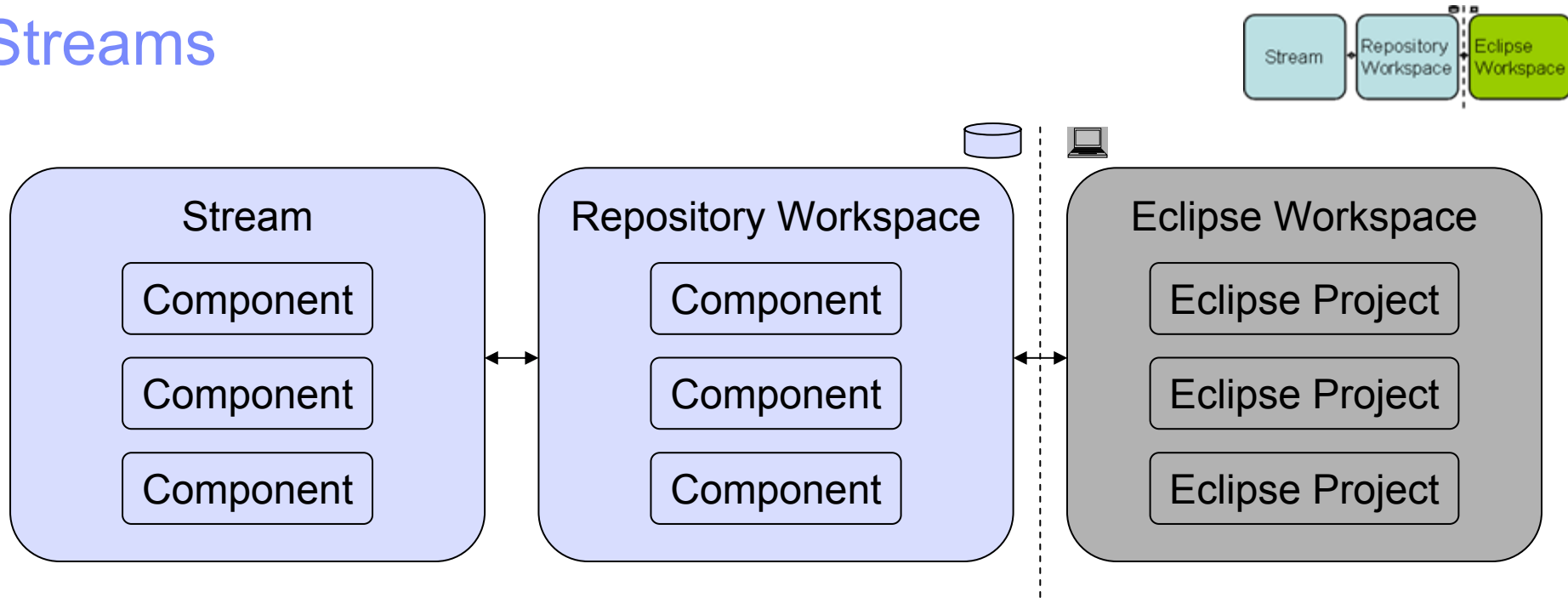
- ▶ Contain components
- ▶ Jazz understands the structure of your project
- ▶ Components map to Eclipse Projects in an Eclipse Workspace
 - Jazz will create these Eclipse Projects for you



Components



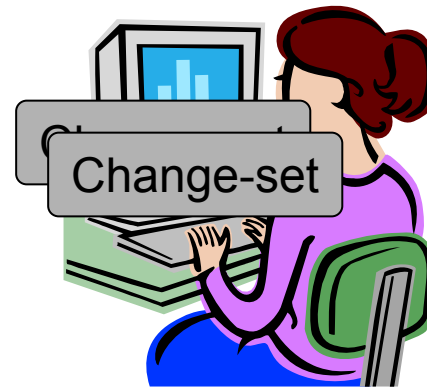
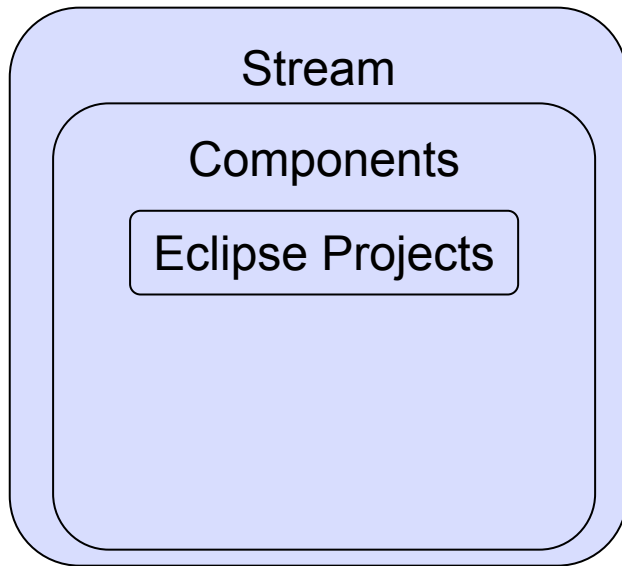
Streams



- Eclipse Workspace
 - ▶ Where a developer actively adds and changes source
- Repository Workspace
 - ▶ A developer's private space in the repository
- Stream
 - ▶ A place to share source with your team



Change-sets



- Components Track Changes
 - ▶ Collection of files and folders is built up from the steady build up of change-sets
 - ▶ Each change-set builds on what came before
- Change History
 - ▶ A time-ordered sequence of change-sets
 - ▶ Describes how the component's content was built from nothing



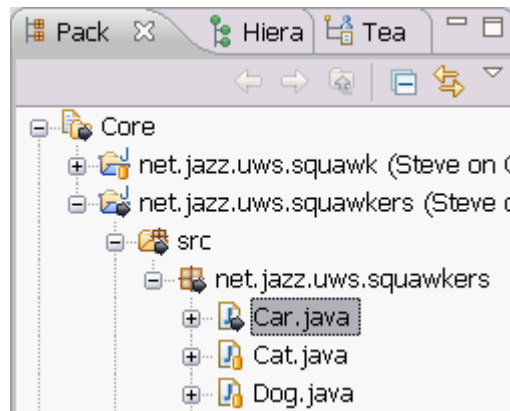
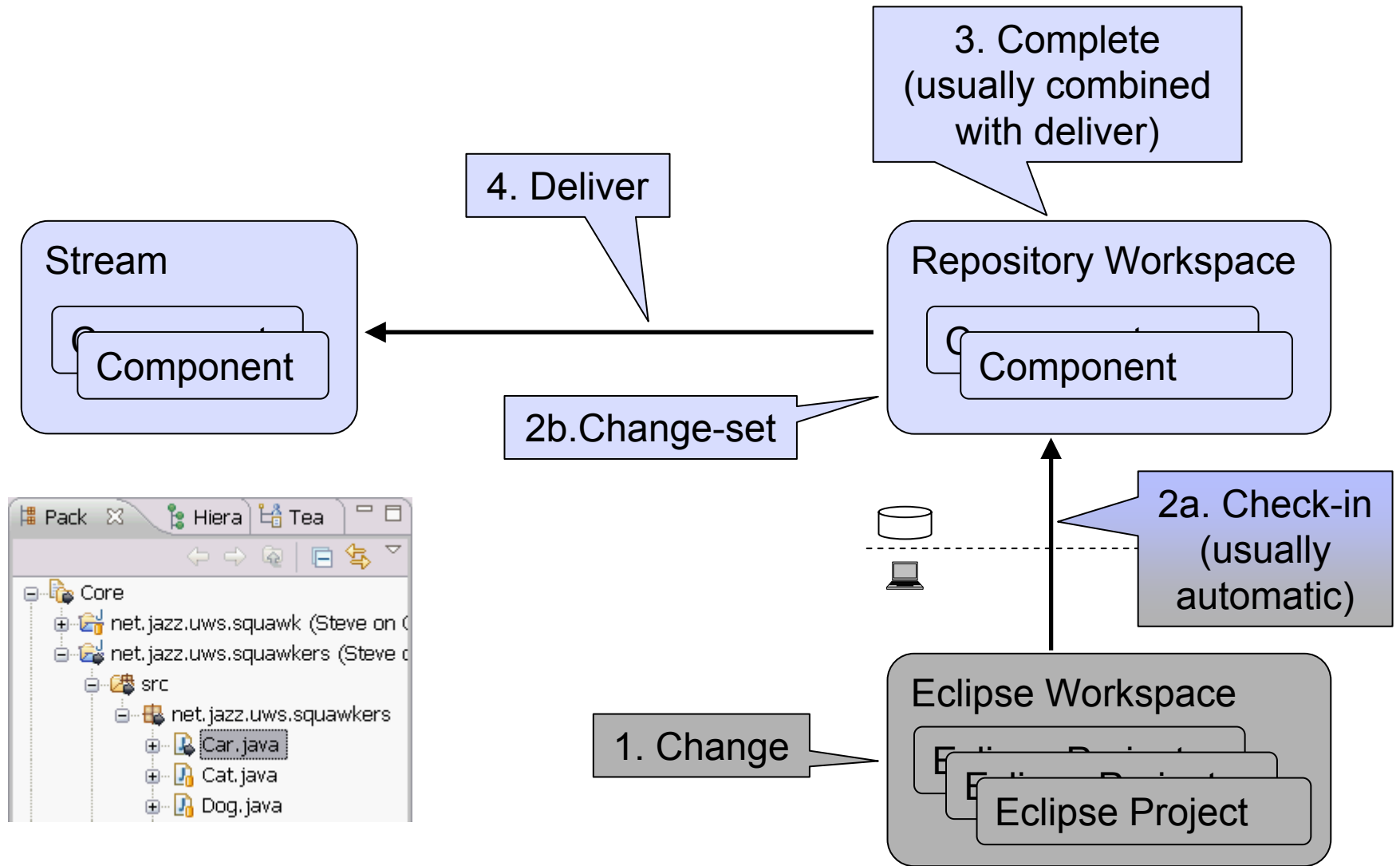
Change-set Details

- Composed from a collection of changes to one or more files and folders
 - ▶ Manage dependencies in an integrated and transparent fashion
 - ▶ Change-set orientation is another example of Jazz understanding your project
 - ▶ A change-set that affects multiple resources is committed as a single atomic unit
- Indicates the reason for the changes
 - ▶ Traceable artifacts
 - ▶ Collaborative in nature via a comment, and/or
 - ▶ By referencing the relevant work item
- Can be shared with another team member in real time to increase productivity
 - ▶ Via a stream, or
 - ▶ From your repository workspace via a work item



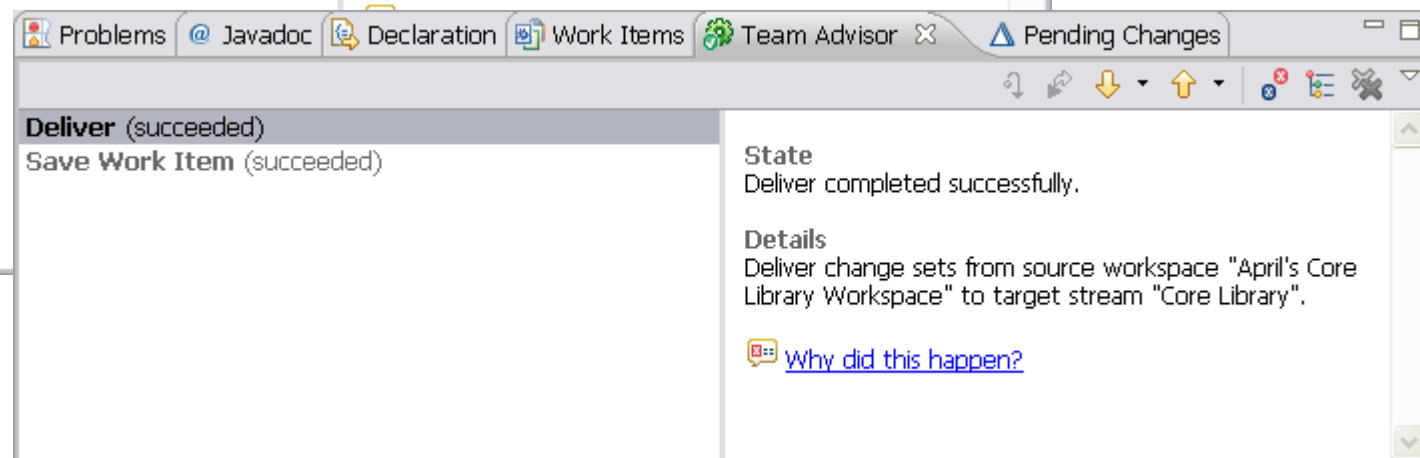
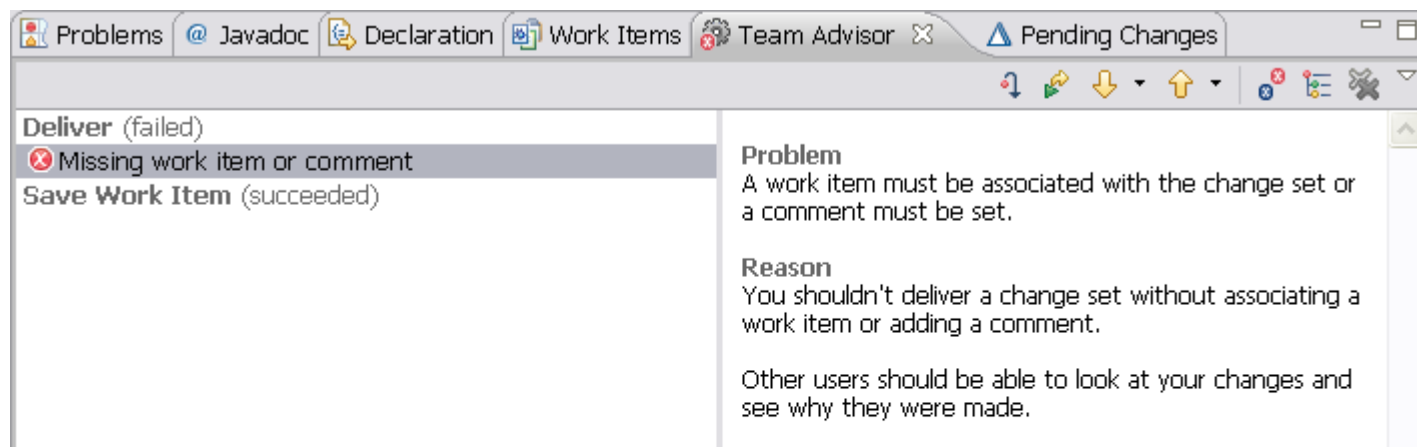


Typical Journey For A Change-set



Change-set Delivery is Process Enabled

- The deliver operation is process-enabled, allowing the team's process to check and enforce delivery rules automatically



Delivery Notifications

Team Artifacts Team Central

Event Log Events (28 unread)

- [2] 'Lion Documentation' (12) 30 minutes ago
 - Geoff Clemm delivered 1 change-sets to Documentation 30 minutes ago
- [5] 'Lion Implementation' (11) 1 hour ago
 - Geoff Clemm delivered 1 change-sets to Core Library 3 hours ago
 - Geoff Clemm delivered 1 change-sets to Core Library 3 hours ago
- [3] Current iteration changed: Yesterday
- Team Area created: User Interface Yesterday
- Steven Wasleski created stream 'Documentation' Yesterday
- Steven Wasleski created stream 'Core Library' Yesterday

News

Build 0 events

My Open Work Items (15:47 updated)

Change-sets Delivered

Geoff Clemm delivered 1 change-sets to Documentation

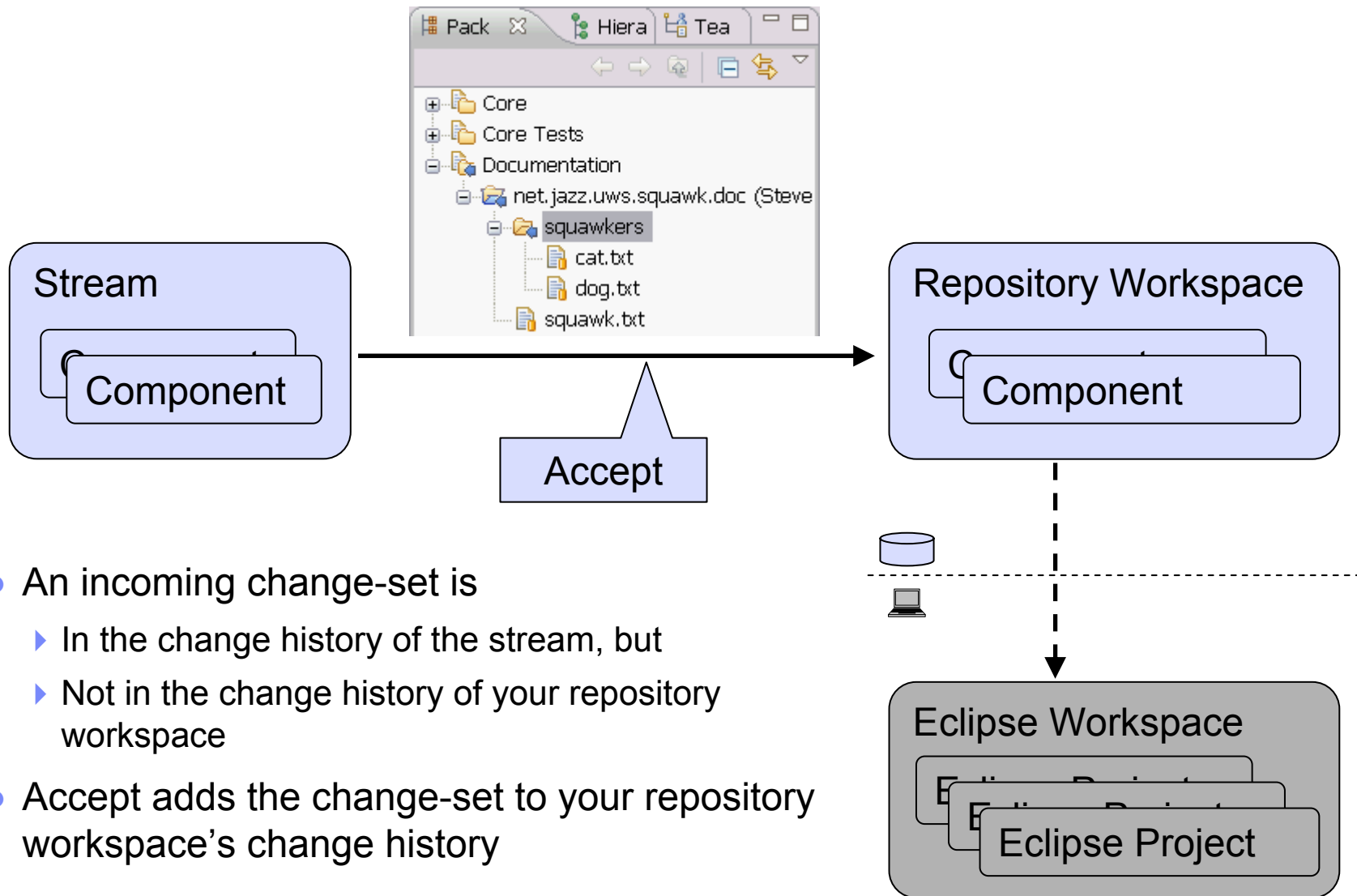
Work Item Problems Team Adv Team Org Pending C

1 incoming change-set

- Steve on Core Library <-> Core Library
- Steve on Documentation <-> Documentation
 - Documentation
 - Incoming
 - Geoff Clemm 12: Lion Documentation - <No Comment>
 - net.jazz.uws.squawk.doc/squawkers
 - lion.txt
 - 12: Lion Documentation



Getting Teammates' Delivered Work



- An incoming change-set is
 - ▶ In the change history of the stream, but
 - ▶ Not in the change history of your repository workspace
- Accept adds the change-set to your repository workspace's change history



DEMO

- Sharing work across your team



Agenda

- All that Jazz
- **Rational Team Concert**
 - ▶ Overview
 - ▶ Getting Started with a new Team
 - ▶ Source Configuration Management
 - ▶ **Build Management**
 - ▶ Change and Traceability

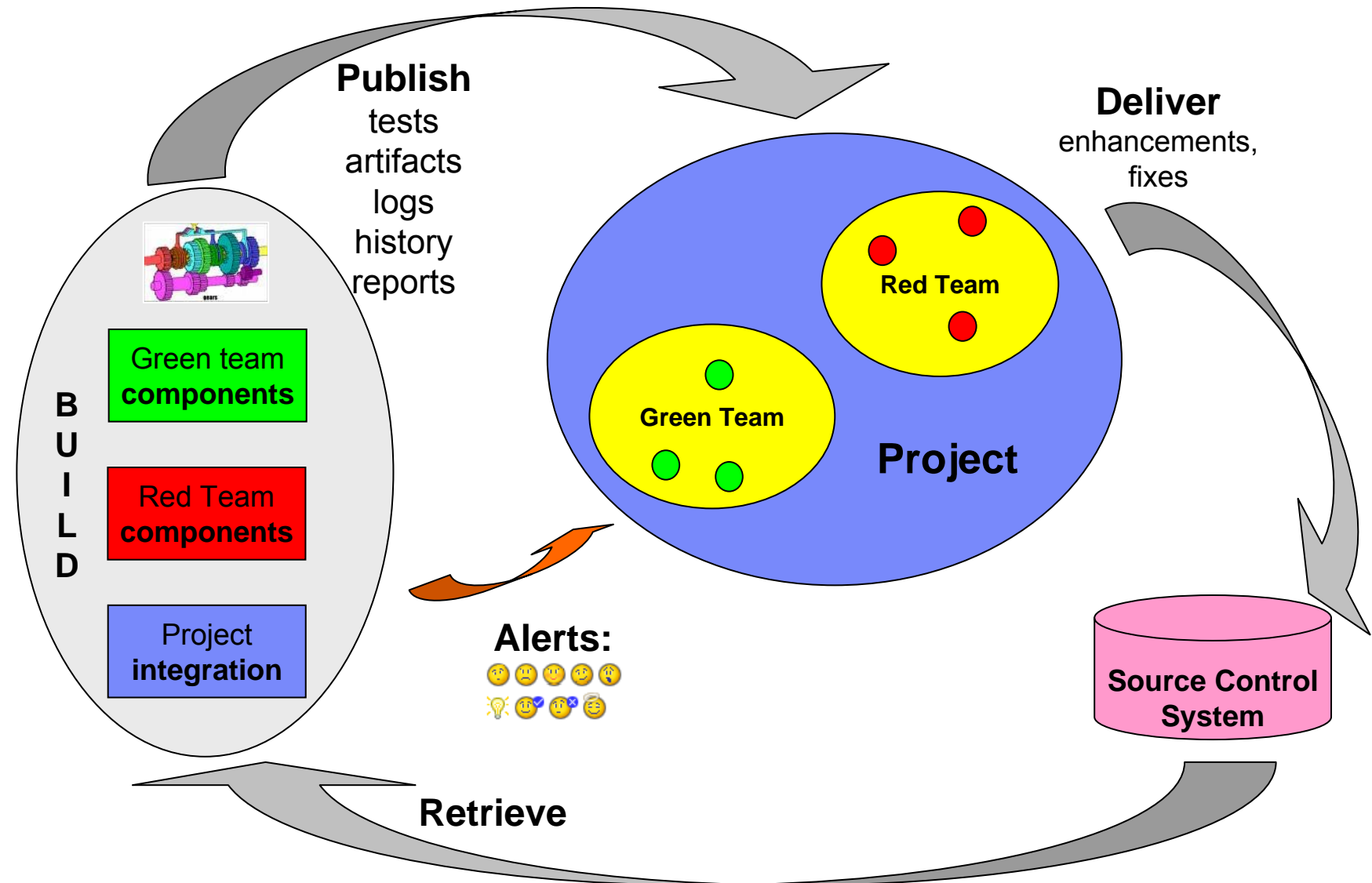


Build Challenges and their impacts

- Silos of information contribute to reducing team productivity
- Inconsistent policies and processes for project stakeholders
- Difficult to facilitate communication, visibility, sharing, and leveraging globally distributed teams without mass chaos
- Limited to no traceability among the artifacts
- Difficulty tracking status of multiple builds
- Delays and missed deadlines for projects and releases
- Unpredictable product/project release cycles
- Costly to support/integrate/troubleshoot multiple systems to avoid mass chaos
- Failed audits, additional work needed for tracking
- No reliable status of project health and no trend analysis



Build in the World of Agile Team Development



Rational Team Concert Build

- Is an integral part of the project infrastructure
 - ▶ Consistent, repeatable process throughout the project
- Brings awareness of build progress and results to developers
 - ▶ Easy sharing of information
- Links build results to related Jazz artifacts
 - ▶ Integrated experience, traceability and tracking “baked in“
- Allow developers to have a private build area
 - ▶ Build and test code before delivering to the main branch
- Accommodates existing build technologies (Ant, CruiseControl , Build Forge, Maven, ...)
 - ▶ Leverages technology that fits your project best



Build is very visible to the user

The screenshot displays the IBM Rational Team Concert (RTC) interface. The main window shows the details of a build: **Build workshop.squawk.integration.build B20080116-1351-workshop.squawk.integration.build**. The build status is **Completed** with a duration of 10 seconds, starting on Wed, 16 Jan 2008 at 13:51:04. A callout bubble labeled **Published build** points to the build title. The **Contribution Summary** shows 0 errors, 0 warnings, 2 downloads, 1 link, 17 tests, 0 failures, and 0 errors. The **General Information** section lists the requested by (Jerry Jazz), build definition, engine, and history. A callout bubble labeled **My builds** points to the left-hand tree view showing repository connections and build engines. A callout bubble labeled **Recent builds** points to the right-hand pane showing recent build events. A callout bubble labeled **Alerts** points to the bottom right corner showing a notification: **Build finished successfully workshop.squawk.core.continuous.build B20080116-1350-workshop.squawk.core.continuous.build**. At the bottom, a **History** table lists recent builds.

Build	Label	Progress	Est	Start Time
workshop.squawk.integration.build	B20080116-1351...	Completed		Wed, 16 Jan 2008 13:...
workshop.squawk.core.continuo...	B20080116-1350...	Completed		Wed, 16 Jan 2008 13:...
workshop.squawk.integration.build	B20080115-1638...	Completed		Tue, 15 Jan 2008 16:3...
workshop.squawk.core.continuo...	B20080115-1637...	Completed		Tue, 15 Jan 2008 16:3...



Private builds

- Private or Personal Builds allow you to build your changes before delivering them to the stream. This can provide you with some assurance that your changes will not disrupt the team when you deliver them.

The screenshot displays the IBM Rational Build System interface for a specific build. The window title is "B20080516-1631-workshop.squawk.core.continuous.build". The main heading is "Build workshop.squawk.core.continuous.build (personal build by student1)".

Completed
Duration: 37 seconds
Start Time: May 16, 2008 4:30:44 PM
Completed: May 16, 2008 4:31:21 PM

Status Trend:

Reported Work Items

- None reported against this build
- [Create a new work item](#)
- [Associate an existing work item](#)

Contribution Summary

- Downloads: [1 download](#)
- External Links: [1 link](#)
- JUnit: [4 tests, 0 failures, 0 errors](#)
- Logs: [1 log](#)
- Repository Workspace: [Student1 Core Library Workspace](#)
- Compile: [0 errors, 0 warnings](#)

General Information

- Requested by: student1
- Build Definition: [workshop.squawk.core.continuous.build](#)
- Build Engine: [workshop.build.engine](#)
- Build History: [23 builds](#)
- Tags:
- Deletion allowed

Navigation tabs at the bottom: Summary | Build Activities | Compilation | JUnit | External Links | Logs | Downloads | Properties



Builds and Snapshots

- A build can request a snapshot
 - ▶ If there are any changes in a component since the last build
 - A new baseline is created with the same name as the snapshot name
 - ▶ Convenient for reproducing build problems

The screenshot displays a web browser window with the URL `workshop.squawk.core.continuous.build_20080115-1637`. The page title is "Snapshot" and it includes a "Save" button. The "Name" field contains the same URL. The page is divided into several sections:

- Details:** Shows "Created by: build", "Created on: Jan 15, 2008 4:37 PM", "Modified on: Jan 15, 2008 4:37 PM", and a "Description" field with the text "Snapshot created by automated build".
- Links:** Provides four actions: "Create a new repository workspace", "Create a new stream", "Compare with snapshot", and "Compare with repository workspace or stream".
- Components:** Lists three components: "Build Scripts (4: workshop.squawk.com)", "Core (6: v20071108-Bow-wow)", and "Core Tests (6: v20071108-Bow-wow)".
- Projects in Component:** A section intended to show projects within the selected component, currently empty.

Agenda

- All that Jazz
- **Rational Team Concert**
 - ▶ Overview
 - ▶ Getting Started with a new Team
 - ▶ Source Configuration Management
 - ▶ Build Management
 - ▶ **Change and Traceability**



Builds

The screenshot displays the IBM Rational Team Concert interface. The main window shows the details of a build: **Build workshop.squawk.core.continuous.build** (ID: B20080314-1507-workshop.squawk.core.continuous.build). The build status is **Completed** with a duration of 28 seconds, starting on March 14, 2008 at 2:07:10 PM and completing at 2:07:39 PM. A status trend bar shows a sequence of red and green bars. The **Contribution Summary** section lists various metrics: 1 download, 1 external link, 1 log, repository workspace, snapshot, 0 errors/warnings, 4 tests/0 failures/0 errors, changes, and 3 work items included in the build. A callout box points to the '3 included in build' link with the text: **Identify work items and change sets that went into the build**. The **Associated Release** section notes that released builds are available as choices in the work item 'Found In' field. The **General Information** section shows the build was requested by April Blues, defined by workshop.squawk.core.continuous.build, using the workshop.build.engine build engine, with a history of 16 builds. A 'Deletion allowed' checkbox is checked. At the bottom, a table lists work items found in the build:

I	Status	P	S	Summary	Owned By	Created By
35	Resolved		Ignore		Zach Builder	Jerry Jazz
32	Resolved		Problem with build B20080306-1029-squawk.m1.maint...		Jerry Jazz	Jerry Jazz
31	Resolved		Clean up Dog squawker		Zara Intern	Jerry Jazz



Work Items

The screenshot displays the 'Work Items - Squawk - Team Concert' application window. The interface includes a menu bar (File, Edit, Navigate, Search, Project, Run, Window, Help), a toolbar, and a left-hand navigation pane. The main area is divided into a 'Details' section for a specific defect and a 'Discussion' section. At the bottom, there is a table listing multiple work items.

Defect 32 Details:

- Summary: Problem with build B20080306-1029-squawk.m1.maintenance.build
- Type: Defect
- Severity: Critical
- Found In: Unassigned
- Created: Mar 6, 2008 12:31 PM
- Created By: Jerry Jazz
- Team Area: Core Library / Squawk
- Filed Against: Squawk/Core
- Owned By: Jerry Jazz
- Priority: Unassigned
- Planned For: Patches
- Resolved: Mar 13, 2008 5:17 PM
- Resolved By: Jerry Jazz

Discussion:

- 1. Jerry Jazz, Mar 13, 2008, 5:11 PM: Reassigned it back to myself.

Work Items List:

I	Status	P	S	Summary	Owned By	Created By
35	Resolved	Ignore		Ignore	Zach Builder	Jerry Jazz
32	Resolved	Critical		Problem with build B20080306-1029-squawk.m1.maintenance.b...	Jerry Jazz	Jerry Jazz
31	Resolved	Unassigned		Clean up Dog squawker	Zara Intern	Jerry Jazz

List associated work items

Drill down into the details of a work item



Change Sets

Defect 32

Summary:* Problem with build B20080306-1029-squawk.m1.maintenance.build

Resolved Fixed Save

Attachments

Id	Name	Created	Created by	Size	Type

Subscribers

Links

- Change-Sets
 - Changes in Core - Jerry Jazz 3/13/08 5:16 PM
 - Changes in Core Tests - Jerry Jazz 3/13/08 5:16 PM
- Included in builds
 - squawk.m1.maintenance.build M20080313-1516
 - workshop.squawk.core.continuous.build B20080314-1507-workshop.squawk.core.continuous.build
 - workshop.squawk.integration.build I20080314-1507-workshop.squawk.integration.build
- Reported against builds
 - squawk.m1.maintenance.build B20080306-1029-squawk.m1.maintenance.build

Overview Links Approvals History

Work Items Problems Builds Change Explorer

<No Comment >

- net.jazz.uws.squawkers/src/net/jazz/uws/squawkers
 - Dog.java

Review change sets that make up the build

Change Explorer lists files that were modified for a given change set



Compare Changes

Compare changes between versions of a file

```

Java Structure Compare
  - Compilation Unit
    - Dog
      - BOW_WOW : String

Java Source Compare
Dog.java
  /**
   * @author wasleski
   */
  public class Dog extends Squawker {
    private static final String BOW_WOW = "Bow-wow";

    /* (non-Javadoc)
     * @see net.jazz.uws.squawk.Squawker#squawk()
     */
    @Override
    public String squawk() {
      return BOW_WOW;
    }
  }

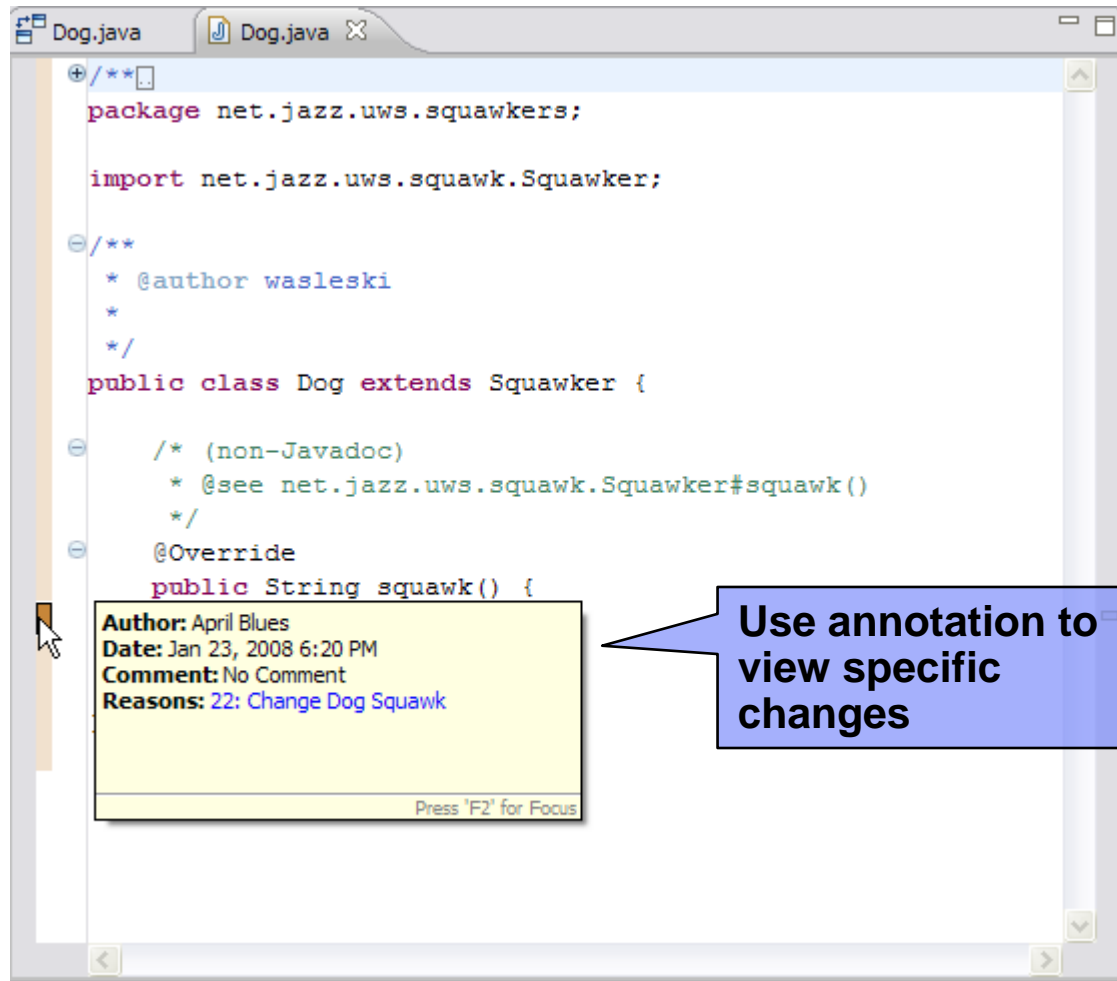
Dog.java
  /**
   * @author wasleski
   */
  public class Dog extends Squawker {
    private static final String BOW_WOW = "Bow-wow!";

    /* (non-Javadoc)
     * @see net.jazz.uws.squawk.Squawker#squawk()
     */
    @Override
    public String squawk() {
      return BOW_WOW;
    }
  }
  
```

Comment	Creator	Date Created	Merges
32: Problem with build B20080306-1029-squawk.m1.mai...	Jerry Jazz	3/13/08 5:16 PM	
31: Clean up Dog squawker	Zara Intern	3/6/08 12:26 PM	
22: Change Dog Squawk	April Blues	1/23/08 6:20 PM	
Share projects	Jerry Jazz	1/23/08 2:05 PM	



Visualize Change History



The screenshot shows an IDE window with two tabs for 'Dog.java'. The code in the editor is as follows:

```
package net.jazz.uws.squawkers;

import net.jazz.uws.squawk.Squawker;

/**
 * @author wasleski
 */
public class Dog extends Squawker {

    /* (non-Javadoc)
     * @see net.jazz.uws.squawk.Squawker#squawk()
     */
    @Override
    public String squawk() {
```

A yellow popup window is displayed over the 'squawk()' method, showing the following change history:

- Author:** April Blues
- Date:** Jan 23, 2008 6:20 PM
- Comment:** No Comment
- Reasons:** 22: Change Dog Squawk

At the bottom of the popup, it says "Press 'F2' for Focus". A blue callout box with a pointer to the popup contains the text: "Use annotation to view specific changes".



Wrap-up

- Jazz Overview
- Rational Team Concert
 - ▶ Overview
 - ▶ Getting Started with a new Team
 - ▶ SCM
 - ▶ Build Management
 - ▶ Change and Traceability
- GET STARTED WITH Jazz TODAY AT
 - ▶ Jazz.net





**THANK
YOU!**

