

# Jazz & Rational Team Concert:

Development Solution for Collaboration, Automation and Extensibility



# Agenda

## The Project Reality

## Jazz & Rational Team Concert Introduction

- Jazz Conception

- Source Code Management

- Software Change Management

- Build Management

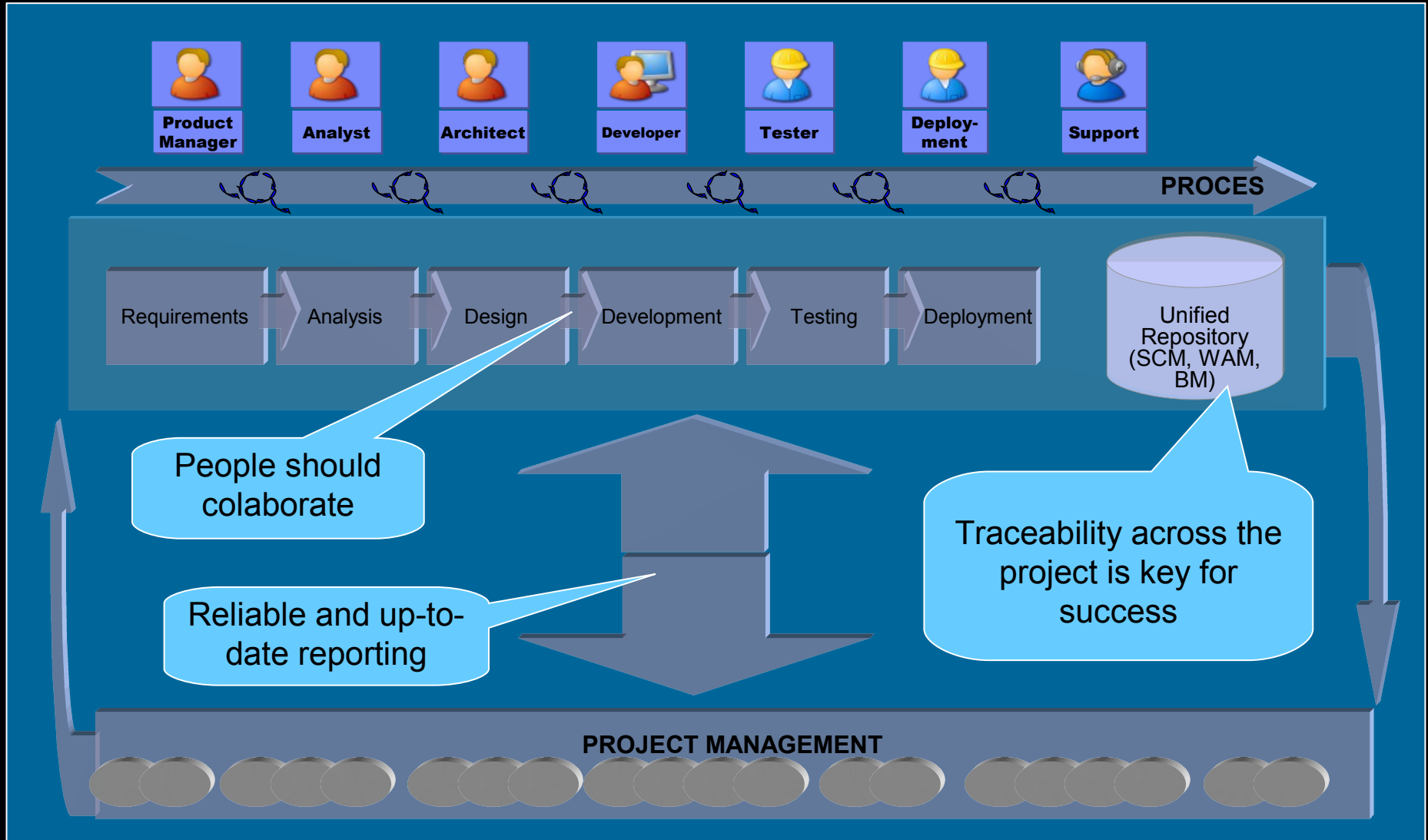
## Using IBM Solution to support project needs

## Benefits of using the solution

# The Project Reality



# The Project Reality



# The Project Reality

## Different tools

Rational Application Developer / Eclipse

WebSphere Integration Developer

MS Visual Studio

...

## Different Teams

Customer team -> Requirements

Different Vendors -> Responsible for the Pieces

Customer teams -> Acceptance Testing & Change Requests

## Different Processes

## Different KPIs

# The Project Reality

## Costs

- Rework (teams are not synchronized)

- Bug fixing

- Administration

## Deadlines

- Lack of synchronization among teams leads to time complications

- Time spent on communication affects the deadlines

## Risks

- Reliability of project reporting?

- Changes impacting the projects?

## Quality

- Unpredictable quality



# Jazz Platform



# Why Are We Doing Jazz?

## The Development Landscape has shifted!

### Late 90's: Focus on **Point Tools**

Who can build the best Java IDE, the best C IDE, the best Web Tool,...

### When we built **Eclipse**: Focus on **One Developer**

Seamless integration across a set of tools to improve the productivity of **one** developer

### Today, we must focus on the **Team** and its **Collaboration**

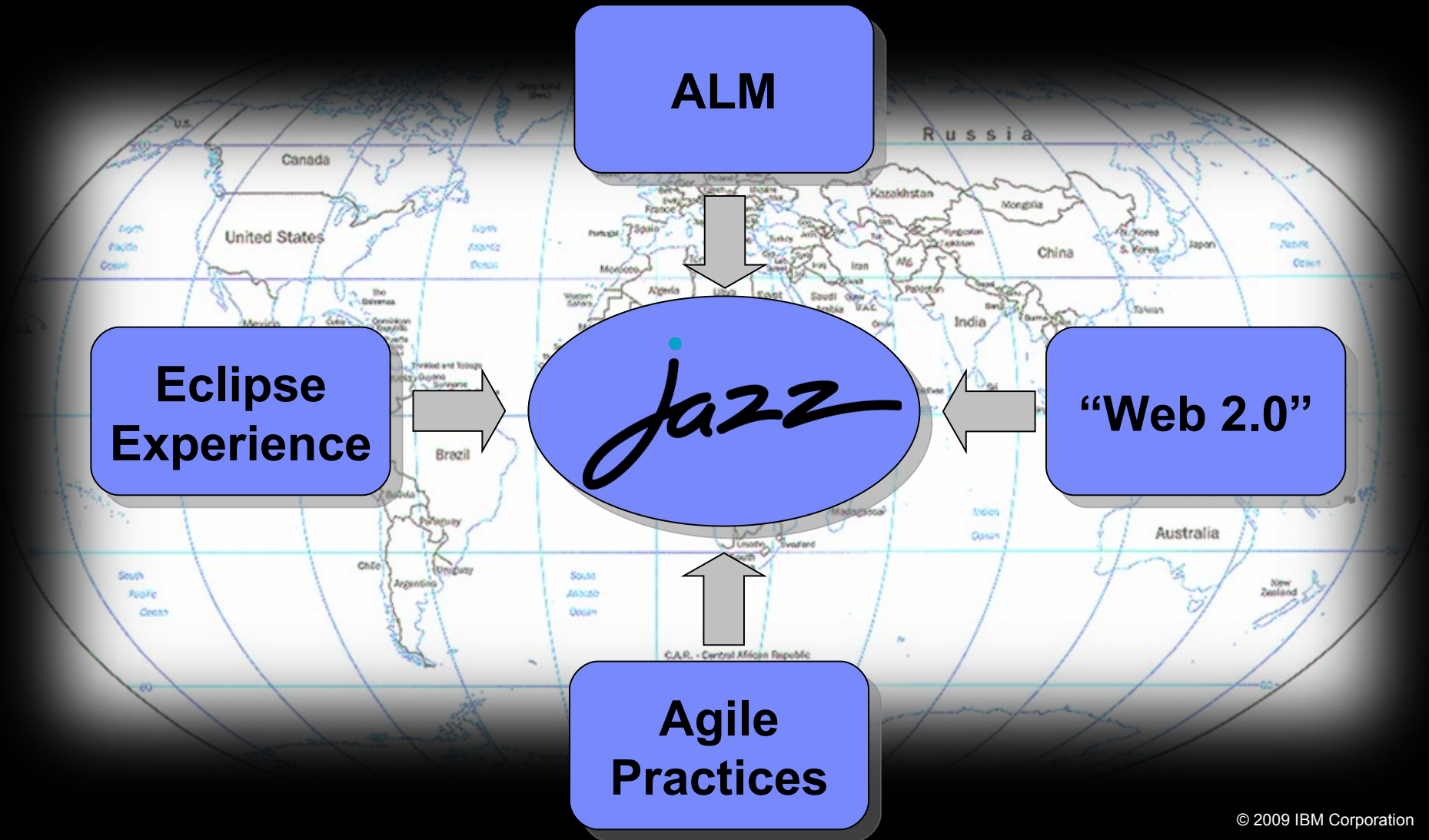
- Geographically Distributed
- Accelerated Delivery Demands
- Agility with Predictability
- Innovation and Repeatability
- Increased Need for Transparency



Seamless integration across  
**All the Phases of the Software Lifecycle**  
to improve the  
**Productivity of the Entire Team.**



# Jazz: Forces in Motion!



# Focus on Application Lifecycle Management

## **Traceability** of relationships between artifacts

*“Managing dependencies between high-priority change requests and ongoing application development efforts “sometimes seems like it isn’t humanly possible.” (US healthcare company)*

## **Automation** of high level processes

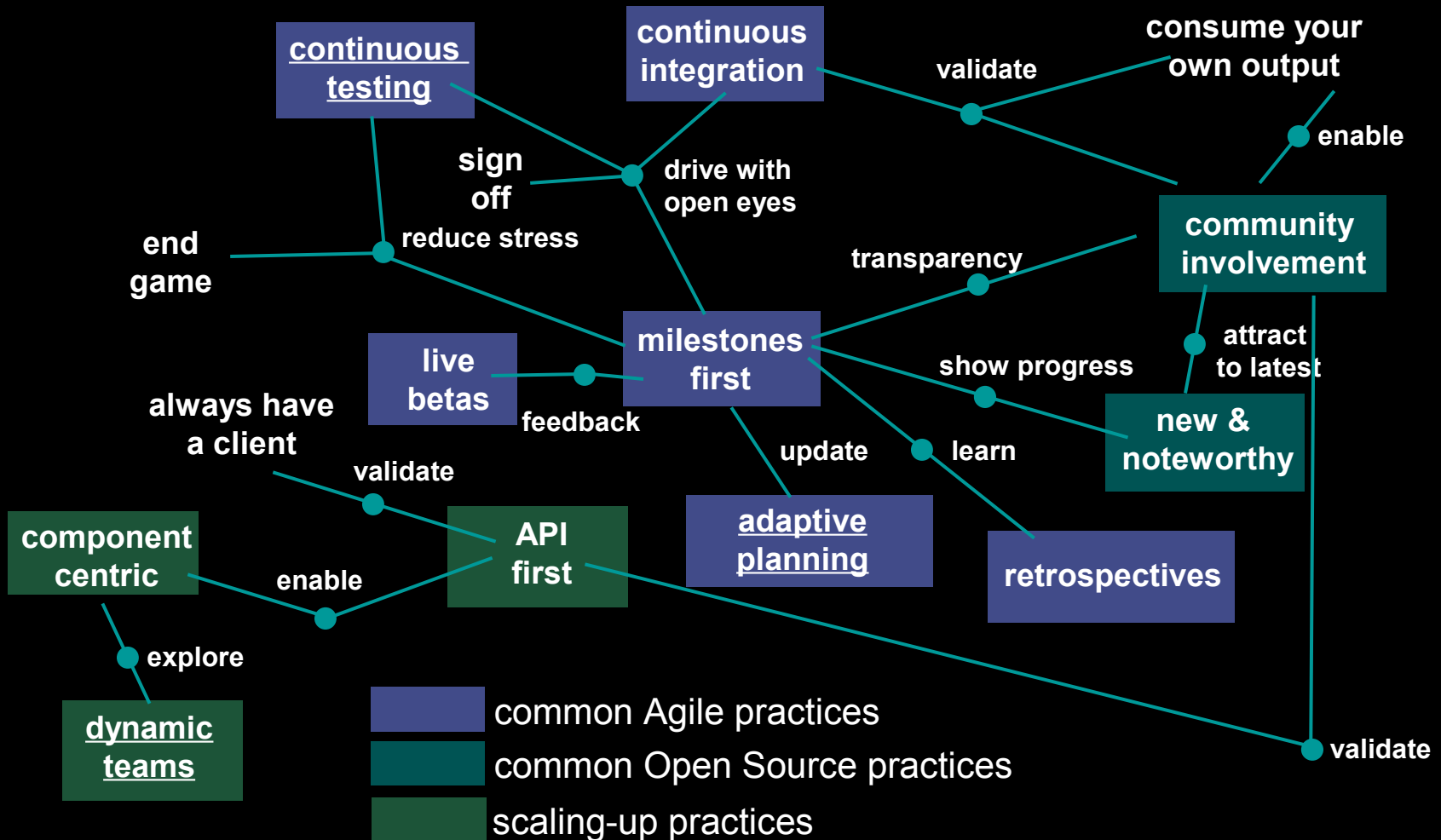
*“We had a consulting company define a methodology for us. We still have it on a shelf somewhere. A process needs to live in the tools we use if it's ever going to be followed.” (Transportation company)*

*“Automating our build-deploy-test processes would probably save each of our developers about an hour a day.” (Financial services firm)*

## **Reporting** provides visibility

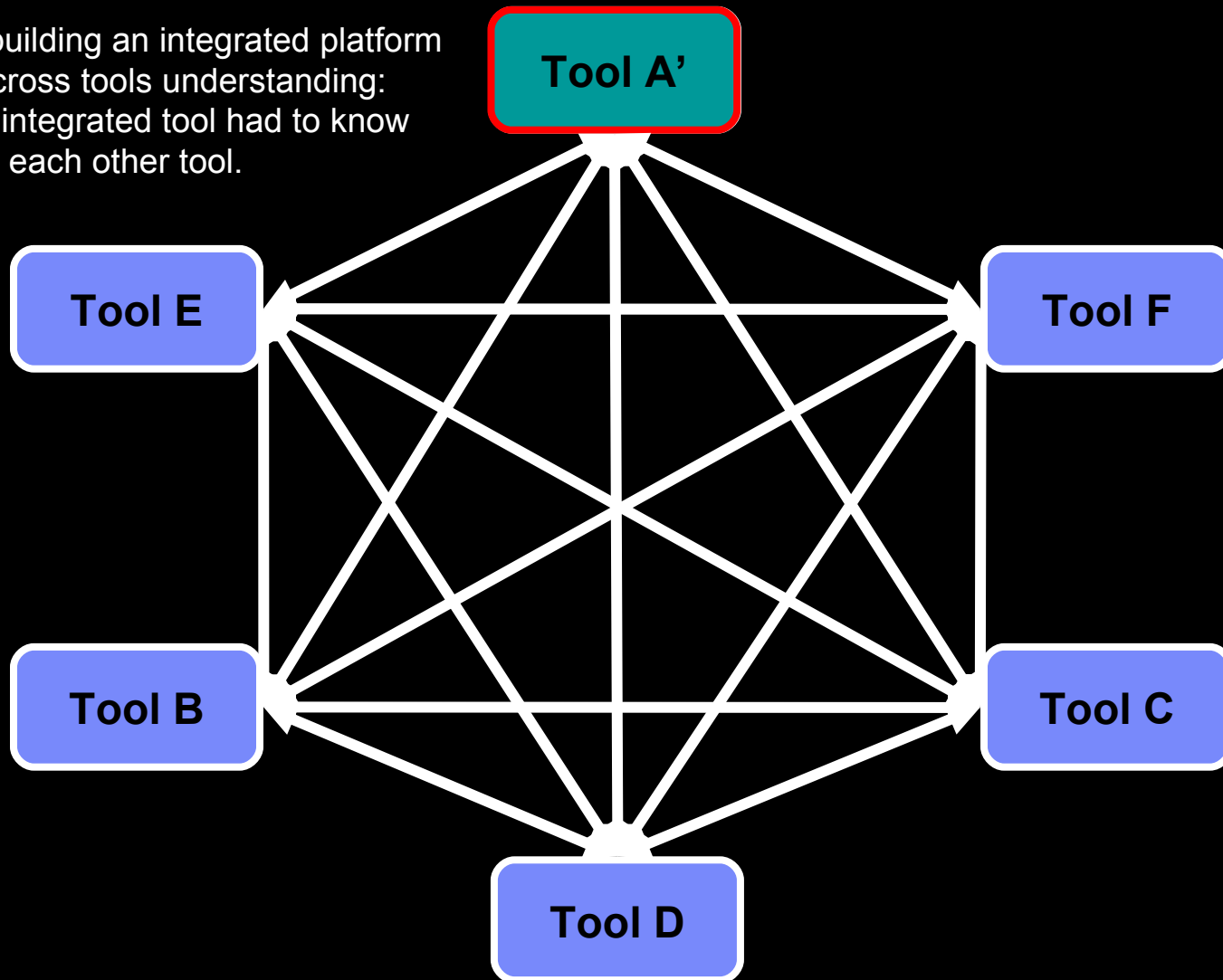
*“We do progress reporting the same way we’ve been doing it for 40 years. It’s all manual: weekly status meetings, progress reports, demonstrations. We’d love to get test results from nightly builds posted somewhere instead of having to run people down to ask them whether things are working yet.” (Bank)*

# Focus on Application Lifecycle Management



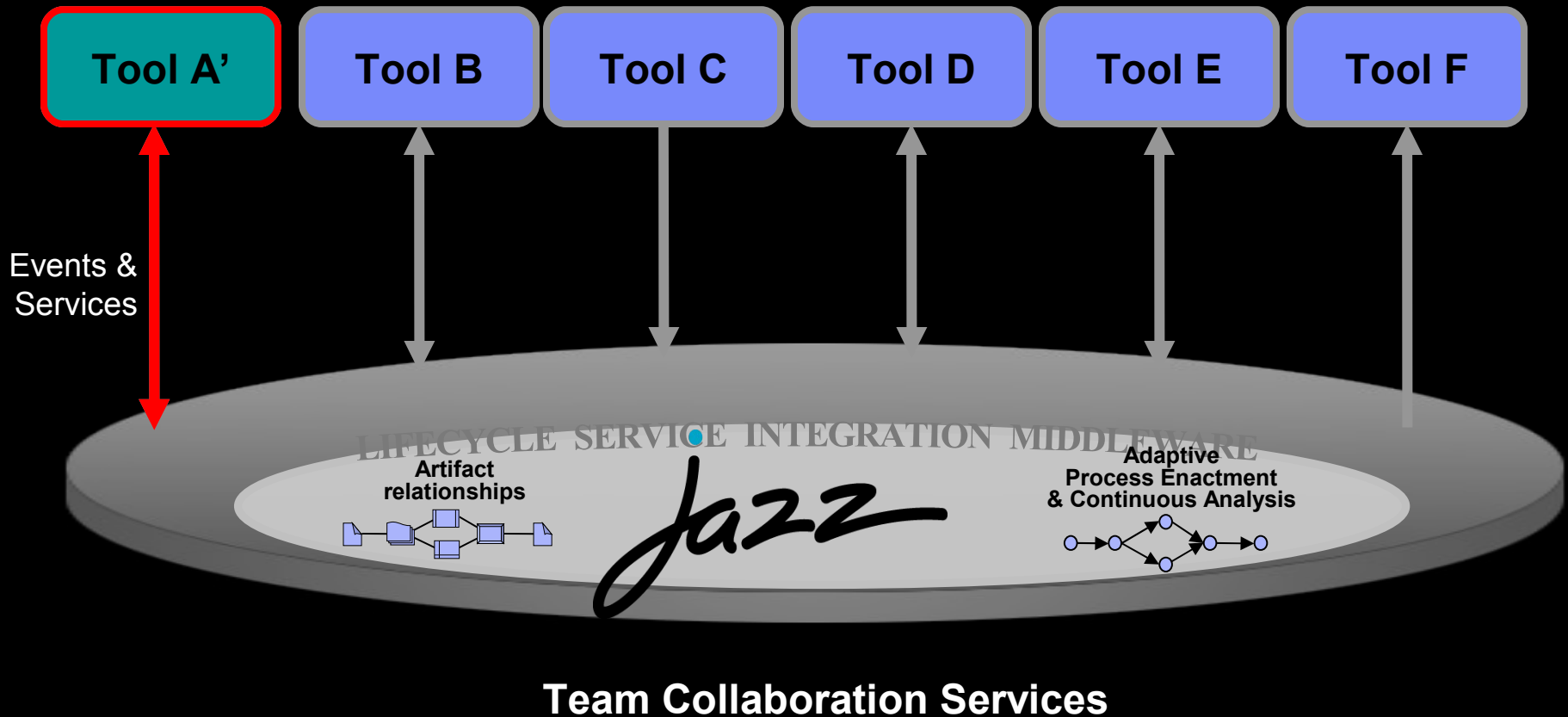
# Collaboration Obviously Means Complexity

Until now, building an integrated platform required a cross tools understanding:  
Each integrated tool had to know about each other tool.

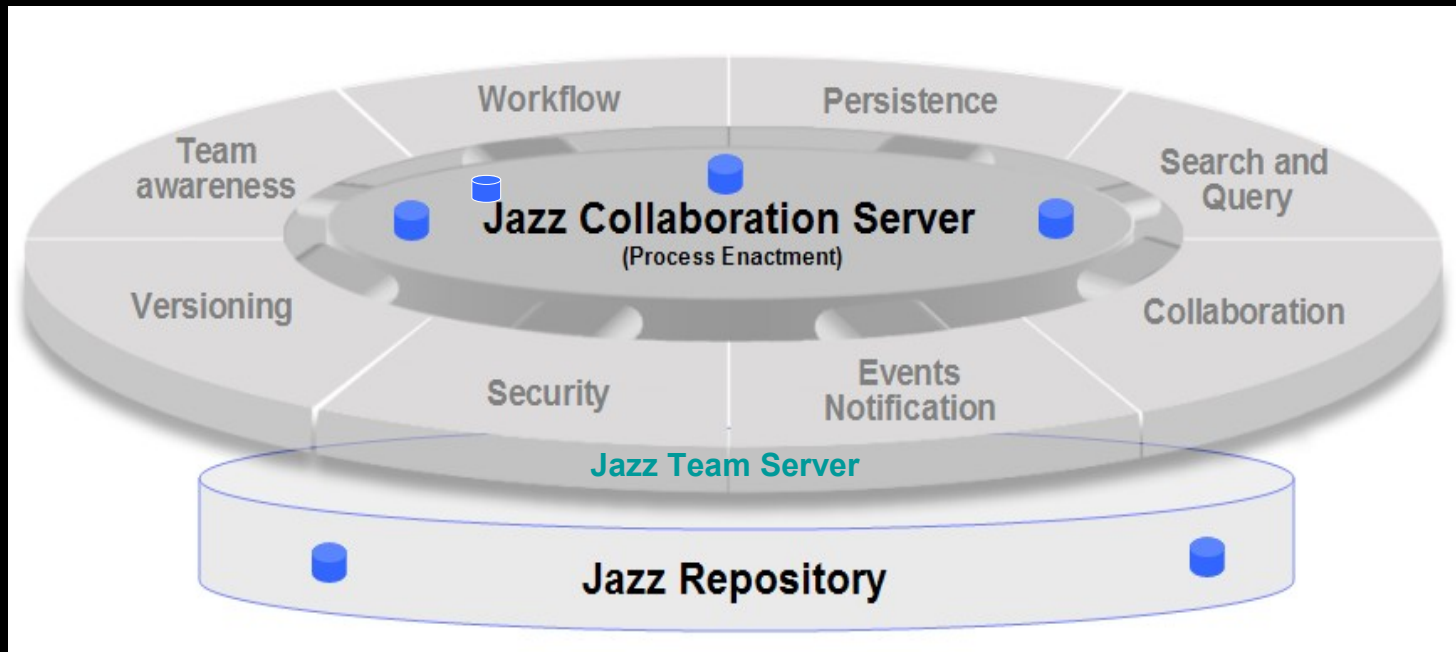


# Jazz as an Integration Middleware

- With the Jazz platform, the tools communicate only with the platform:
  - By listening to normalized/standardized events from the platform
  - By sending normalized/standardized events to the platform



# Jazz as an Integration Middleware





Zdeněk Borůvka – Jazz & Rational Team Concert  
05/27/2009

---



# Rational Team Concert



# Introducing IBM Rational Team Concert

Enables **“real-time, in-context” collaboration** for distributed project teams – making software development more automated, transparent and predictive

Integrates **Source Control, Work Item, Reporting and Build** capabilities which “think and work in unison”

Provides **real-time Project Health** information and **Transparency** of status through automated data gathering

Supports out-of-the-box and custom defined **Processes** which break down functional & organizational barriers

Allows choice of client tools and extends the value of **ClearQuest & ClearCase** in enterprise deployments

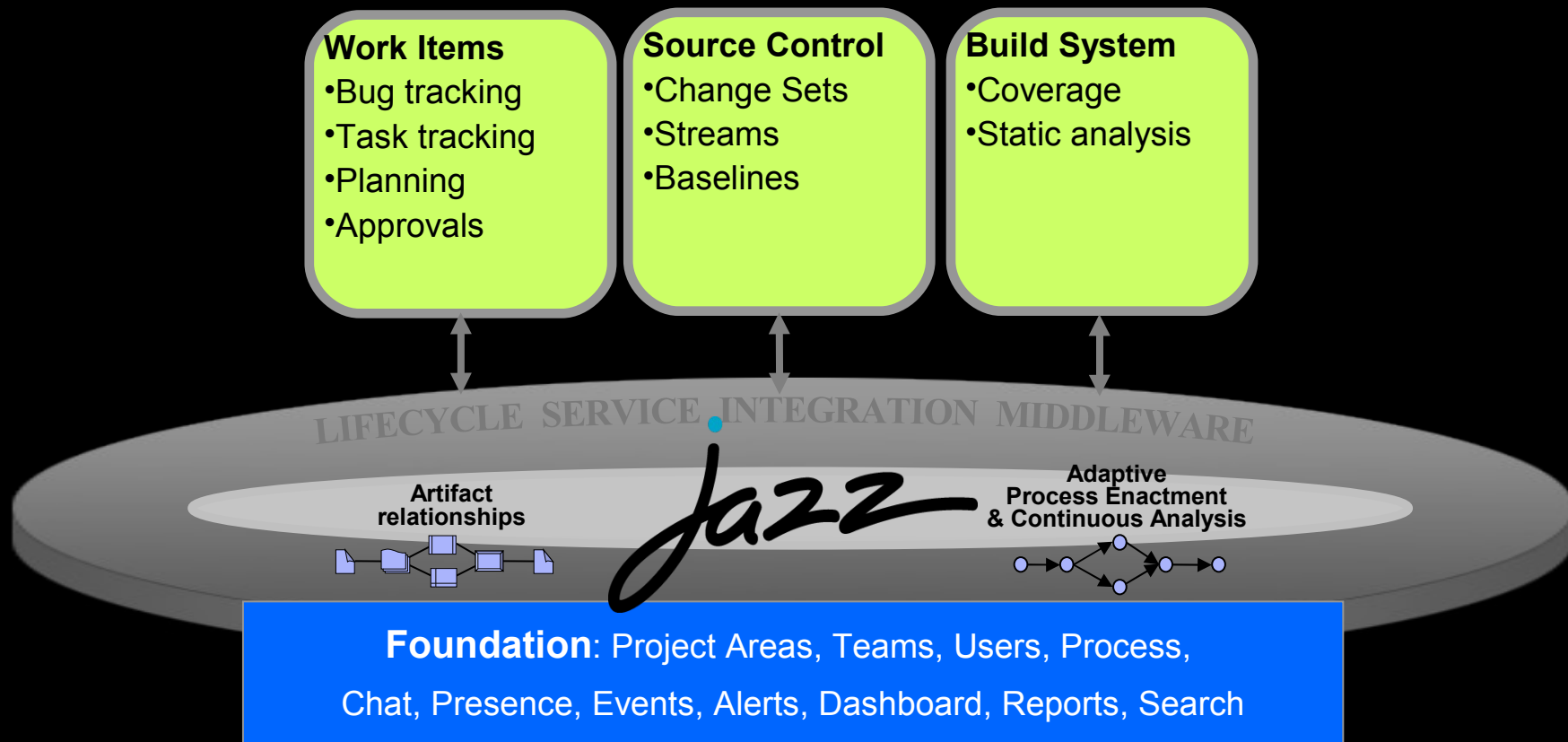
The graphic features a blue header with the text 'IBM Rational Team Concert'. Below the header is a row of five circular icons: a musical note, and four portraits of diverse individuals. Underneath this row is a large image showing silhouettes of people in a meeting, with a glowing globe in the background. At the bottom of the graphic is a white box containing the following text: '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 TEAM SERVER'.

Open and extensible on



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

# IBM Rational Team Concert



Team Concert provides an integrated Collaborative development environment

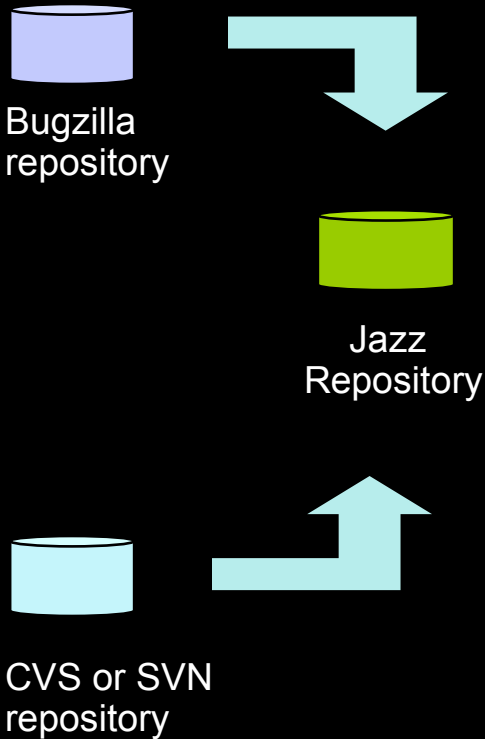
Using a single repository

Integrated Version Control, Work Item and a Build System

# IBM Rational Team Concert

## Import

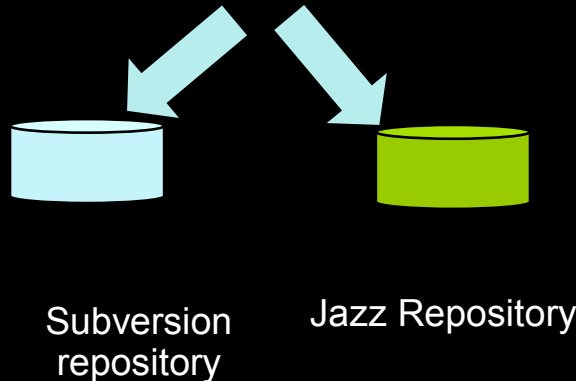
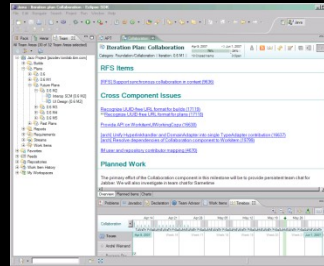
One way data migration



## Bridge

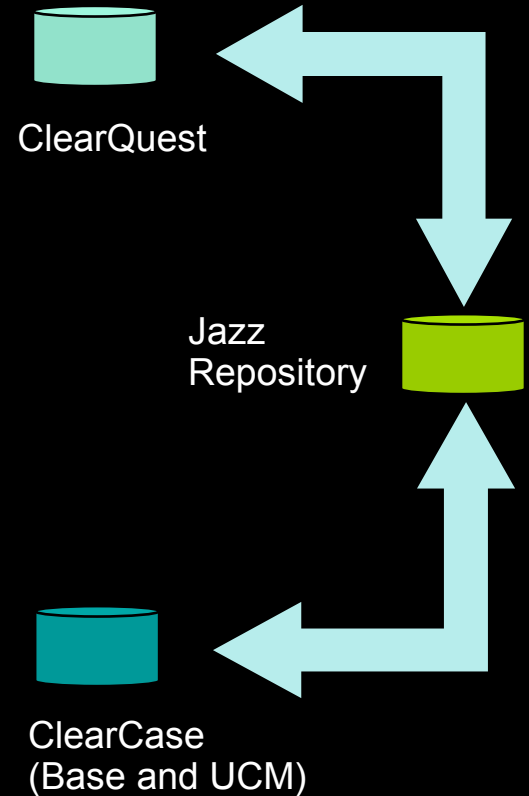
**SVN Bridge** links work items with change sets.

**Mylin repository connector (incubator)**

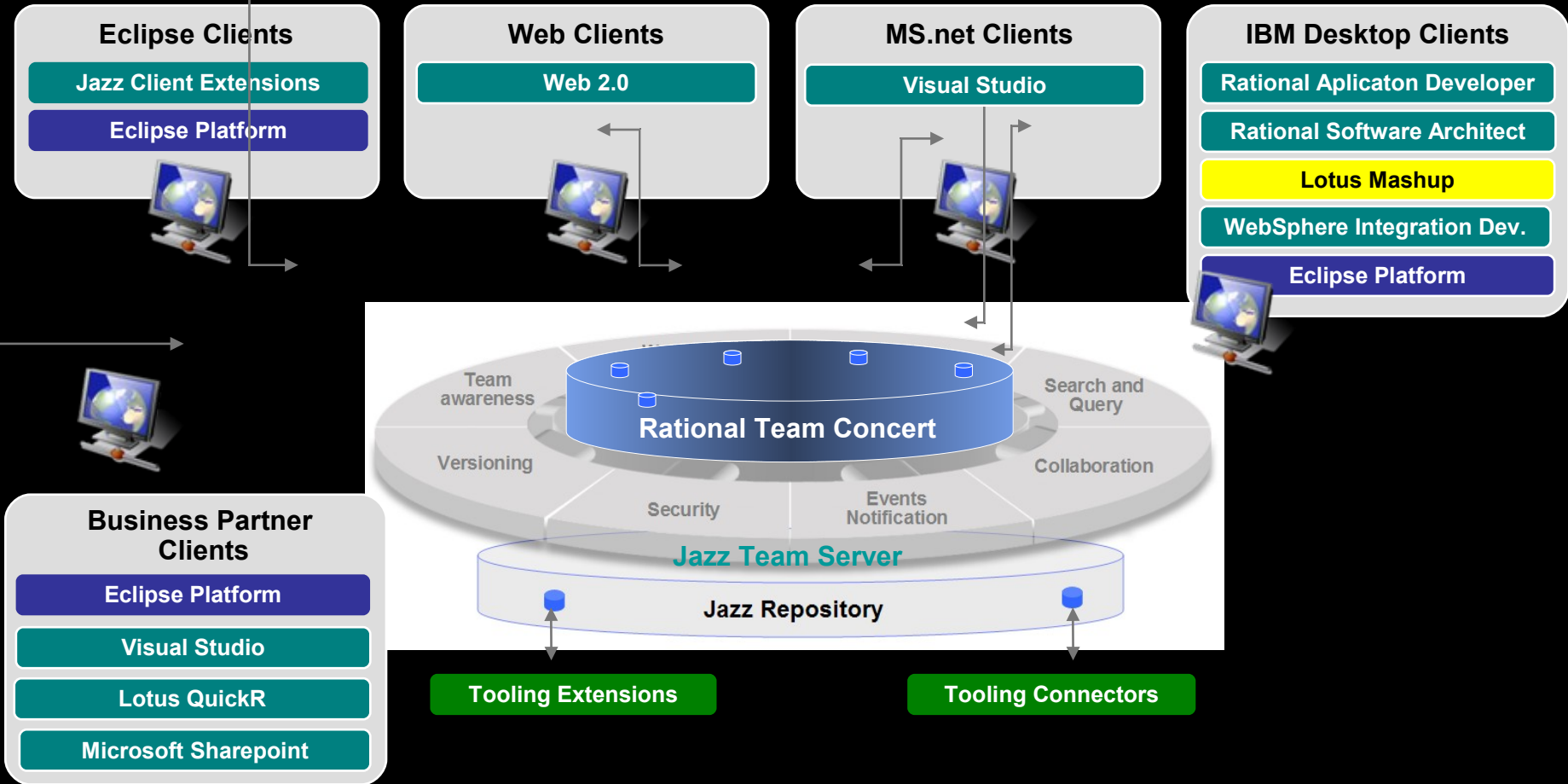


## Connectors

Two way data synchronization



# Rolling Team Concert out to End Users





# Applying Jazz & Team Concert

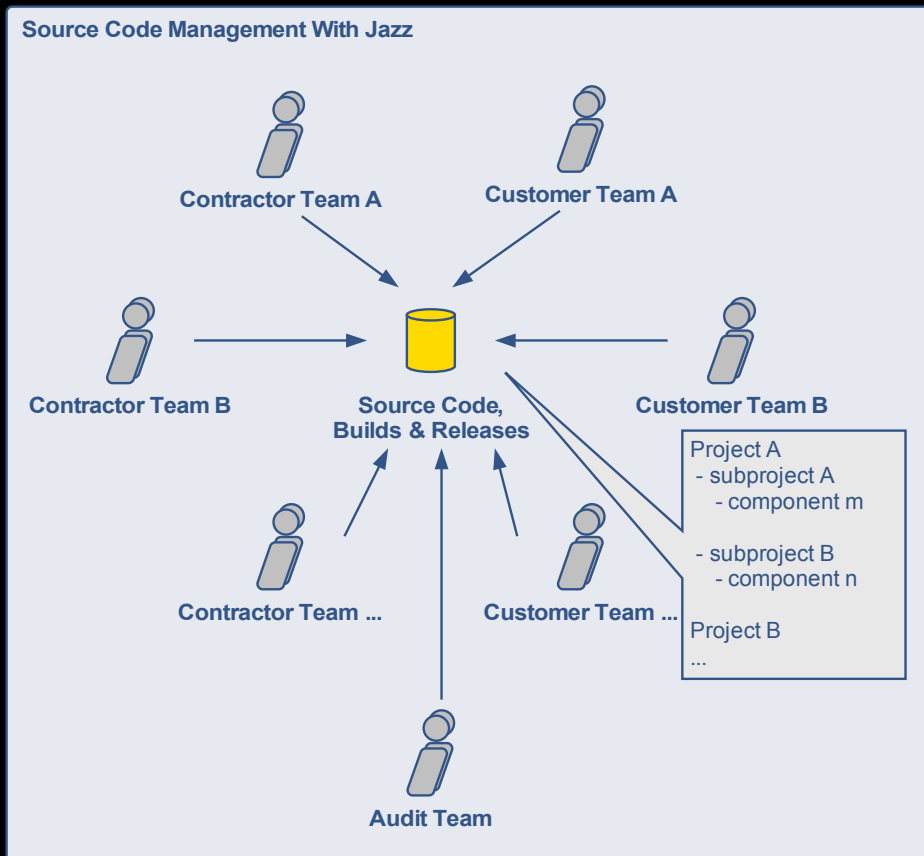




# Benefit one: Unifying Source Code Mng.

One Single Repository for all Products' Releases

Unified Source Code Sharing Principles

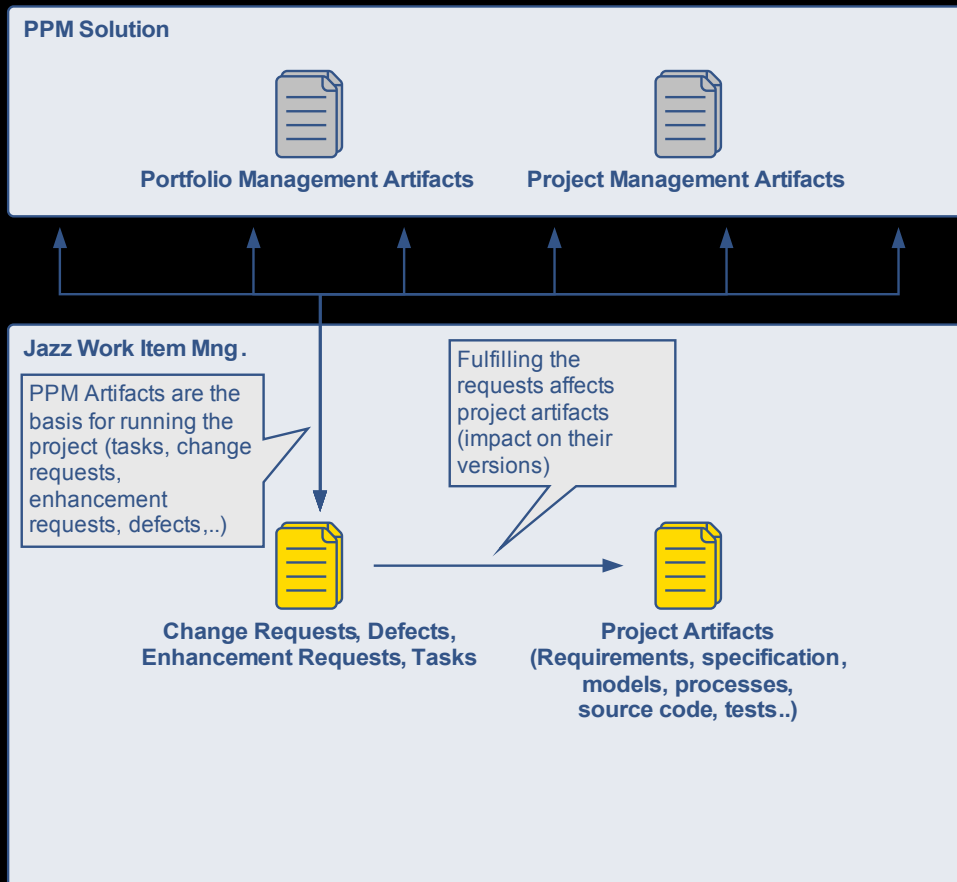


- Store different products or products component releases
- Track source control changes
- Get auditable repository of source code related to software change management process

# Benefit two: Unifying Work Item Mng.

One Single Repository for all Auditable Work Items

Project Management Basis

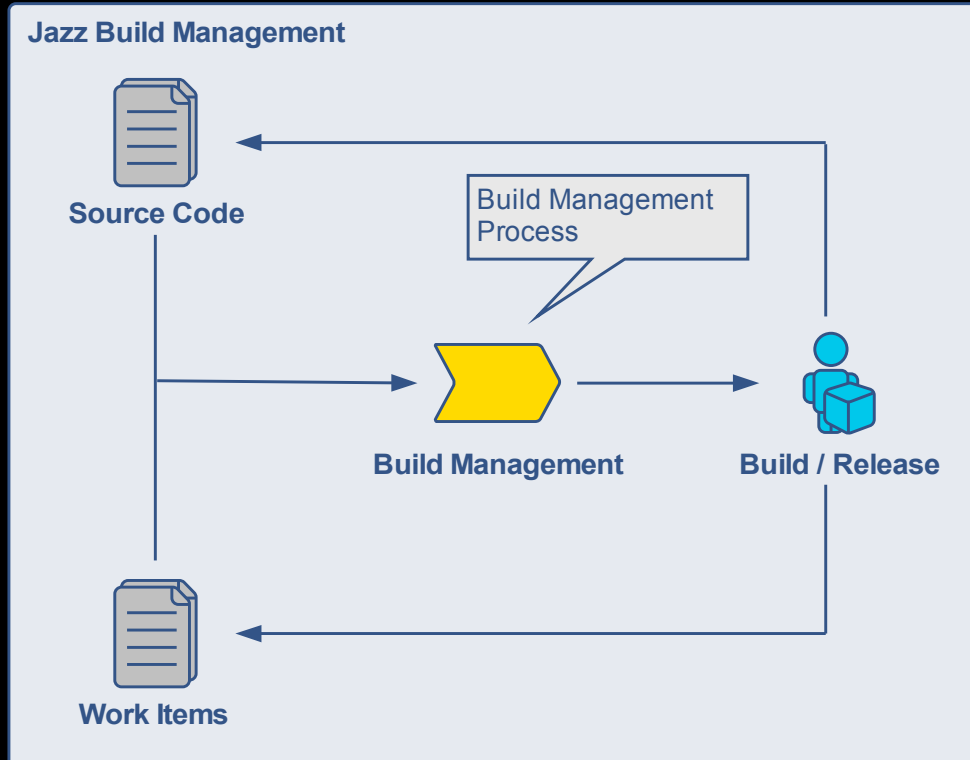


- Address the process needs (planning, tracking, audit & reporting)
- Support key processes (Request Management, Change Management, Defect Management)
- Relate process elements (changes, defects etc.) to concrete release
- Leverage reliable reporting (tasks are basis for project management)

# Benefit three: Unifying Build Management

One Single Repository for all Builds and Releases

Automating the Build Process across many technologies

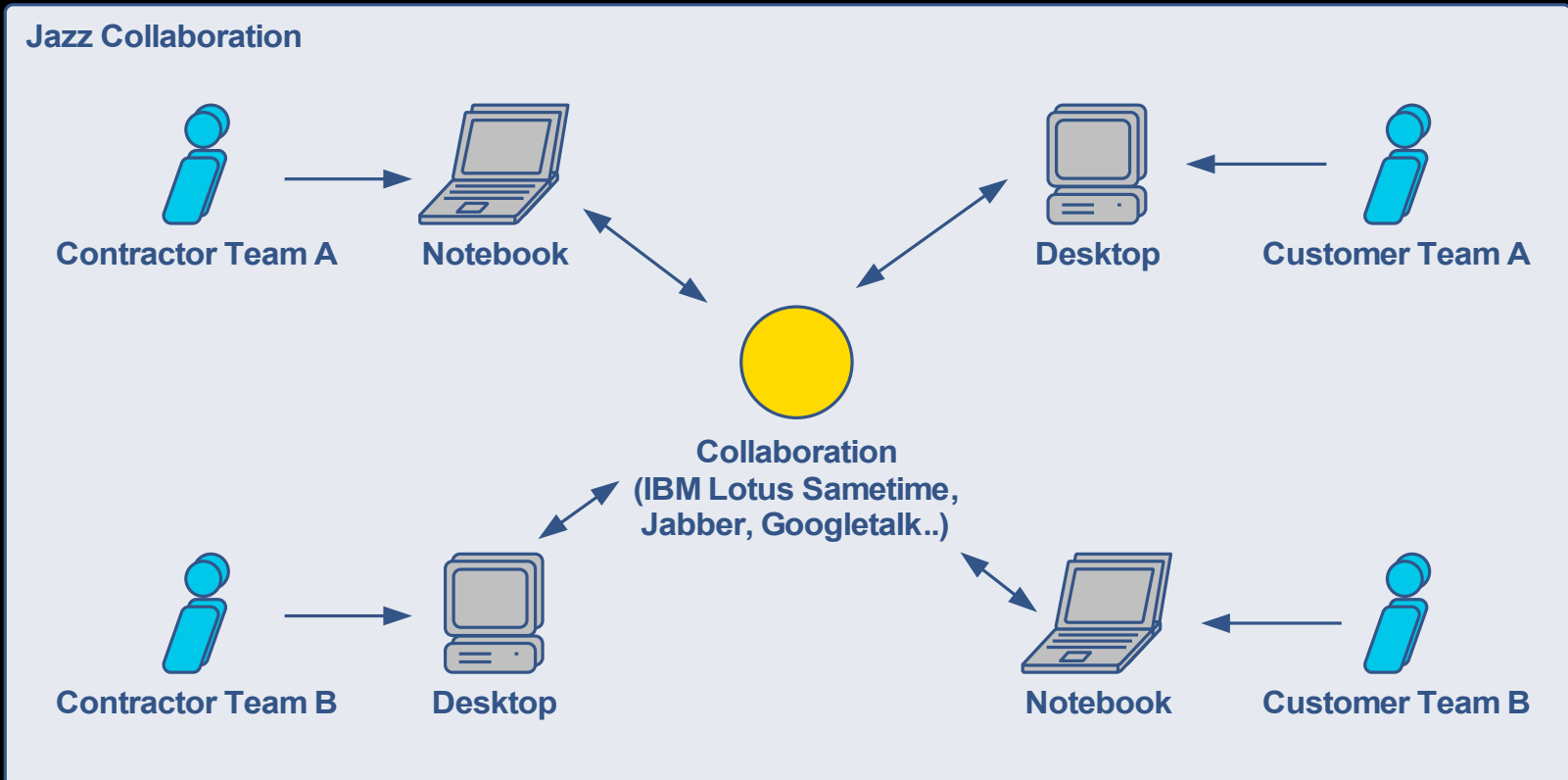


- Build different projects compilations and get the relationships automatically
- Track builds and releases to changes, requests, defects and source control

# Benefit four: Unifying Collaboration

Solve Issues Immediately with Right Person Across Projects

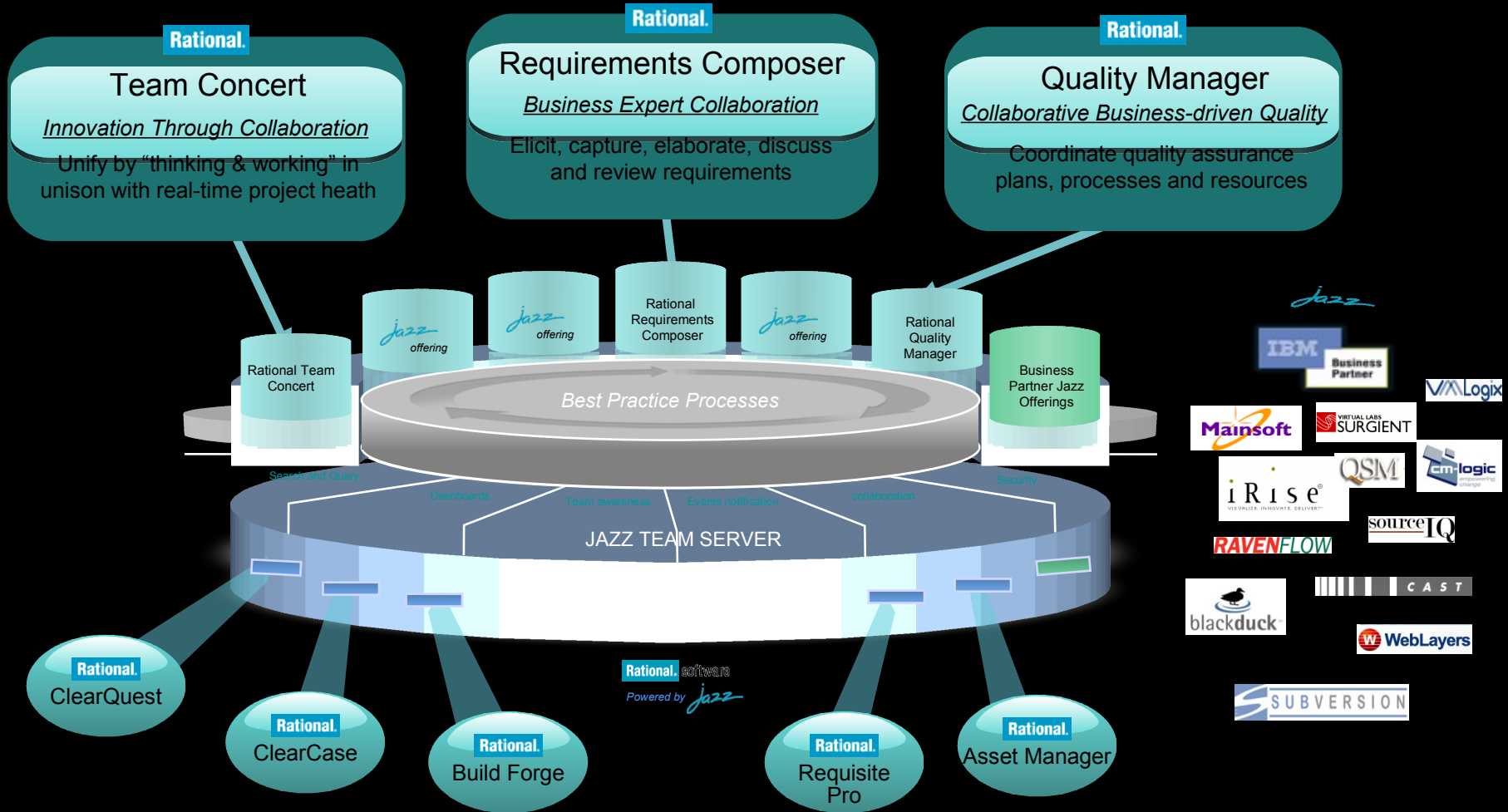
Document Your Communication in a Relevant Way



# Summarizing Jazz Benefits and Vision



# Outlining the Vision





# Jazz „Operational Benefits“ Summary

## ✓ Team Environment Unification

Unifying different tools and teams by providing common team platform

## ✓ Real Time Collaboration

Synchronizing teams by effective collaboration

## ✓ Project Traceability

Brings reliable reporting based on traceability (in context)

Collects the traceability automatically -> reduction of administration effort!!!

## ✓ Quality Management

Enables to systematically plan and control the quality in context of project needs

## ✓ Very Effective Learning Curve

Simplifies adoption of new technologies by using the same concepts

# Jazz „Strategic Benefits“ Summary

## ✓ Getting Visibility into the Projects -> Risk Reduction

Unifying different tools and teams helps to relate important informations -> traceability

Setting up a realiable reporting based on real project data (in context)

## ✓ Rising Team Effectivity -> Costs and Time Savings

Synchronizing teams by effective collaboration

People can collaborate on data stored in a single repository or unified through Jazz

Everybody has the right information in a right time

## ✓ Quality Improvements

Key project decisions as well as daily routine work is based on right informations

Quality problems are identified early

Fixing defects is not invoking new defects

# Meet the Jazz Community at <http://jazz.net>

Home About Jazz Projects Downloads Forums Library Community Register | Log In

**Jazz Forums**

Jazz.net Forum Index >

## Using Team Concert

Users browsing this forum: wainwmr

**New Topic** | Previous | Page 1 of 146 | Next

Topics	Mark all read	Replies	Views	Author	Last Post
RTC versus Jira/SVN/CruiseControl		1	4450	roman.amirak	Fri, Apr 24, 2009 10:20 by peckham
Problem with RTC 2.0 beta 2		5	103	agus	Wed, May 27, 2009 5:22 by agus
Migration from ClearCase to RTC		1	19	David.Caikkel	Wed, May 27, 2009 4:21 by gmlemm
Filepath confusion for junitLogPublisher		3	15	manonfire	Wed, May 27, 2009 1:42 by Chemi
How to share information in wikipedia style?		4	67	David.Caikkel	Tue, May 26, 2009 4:08 by nedgar
Project Area Membership versus Team Area membership		1	25	nerack	Tue, May 26, 2009 2:05 by dmweinan
How to delete build artifacts when a build result is deleted		1	132	james.branigan	Tue, May 26, 2009 1:57 by dmweinan
Removing some of the main headings in RTC Web UI		0	3	dagosti	Tue, May 26, 2009 1:54 by dagosti
Build repository workspace retrieves all stream components		10	525	mluke	Tue, May 26, 2009 1:54 by nedgar
Symbolic links and Source Control.		1	25	amcohen.us.ibm.com	Tue, May 26, 2009 1:49 by echughes
Error when trying to create a report.		4	38	talrab	Tue, May 26, 2009 1:19 by jmoody
Problem with RTC uninstall		1	41	roberto_pozzi	Tue, May 26, 2009 1:19 by jmoody
LDAP configuration		3	49	milan.krivic	Tue, May 26, 2009 1:09 by balagik
Multiple JBE instances using same engine ID		2	428	kvogen	Tue, May 26, 2009 12:23 by nedgar
RTC Build Engine Call to file repository process		1	33	manonfire	Tue, May 26, 2009 12:01 by nedgar
Jazz Team Server Report cant show the data		10	366	disfunction2106	Tue, May 26, 2009 11:27 by jmoody
About jdtCompileLogPublisher		0	3	Chemi	Mon, May 25, 2009 5:14 by Chemi
Work items included in Build results.		0	4	Chemi	Mon, May 25, 2009 2:01 by Chemi
Create a Repository Workspace automatically		0	14	Chemi	Mon, May 25, 2009 10:53 by Chemi
Default work item type in RTC 2.0		0	13	ana.lopez	Mon, May 25, 2009 10:50 by ana.lopez
Tagging automatically Build results		0	3	Chemi	Mon, May 25, 2009 7:00 by Chemi
cannot delete project		5	64	cduswzh	Mon, May 25, 2009 2:48 by gmlemm
How to let customer contribute to a workitem directly		3	55	czhuang	Mon, May 25, 2009 1:11 by rschoon
RTC 2.0 Beta 2 is now available		9	185	acho	Mon, May 25, 2009 1:02 by rschoon
work items personalization		1	41	cargne80	Mon, May 25, 2009 12:56 by rschoon

Previous | Page 1 of 146 | Next

Powered by **phpBB** © 2001, 2005 phpBB Group

Rational software Contact | Privacy Policy | Terms of Use

Built on Eclipse is a trademark of Eclipse Foundation, Inc. Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Microsoft, Visual Studio, and the Visual Studio logo are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.



Jazz TEAM WIKI

You are here: TWiki > | Main Web > Welcome

## Welcome to the Jazz.net Development Wiki

This wiki is used by the Jazz Development Teams to document and discuss technical designs and operational procedures for how we run the projects at Jazz.net. Only the development teams can update the Wiki, but everyone is welcome and encouraged to browse the information, ask questions, and give feedback in the [Eclipse](#)!

**General**

- [Guidelines for this wiki](#)
- [Current Development Details](#) - FAQ on obstacles you may find when developing with the latest deployed build
- [Release Engineering](#) - How we coordinate project-wide builds
- [Developer Setup](#) - How to set things up to develop for the Jazz Platform
- [Eclipse Documenters](#) - General documents that are of wide scope
  - [Hot Topics](#) - [Installation](#) - [Business](#) - [Training](#) - [Connections](#) - [Component Version Numbering](#) - [Writing Adapters and Tech Notes](#) - [Glossary](#) - [UI Check List](#) - [Resolving Migration](#) - [Performance and Scalability](#)
- [Planning Home](#) - Home page for both RTC and Foundation planning
- [Jazzdev status](#) - Detailed status of unexpected outages, migrations and deployments
- [Tips for creating TTY text cases](#)
- [Workflows](#) - What time is it ... ?

**Jazz Foundation**

**Planning**

- [Outlines for Foundation 1.0 Enigmas](#) - [SCL](#) - [GA](#)
- [Jazz Foundation 1.0 current milestones](#) - [1.0 RC2 and 1.0.1.M2](#) - includes Rules of the Road
- [Weekly status](#)

**Process**

- [Communicating foundation changes requiring adoption](#)
- [Legal approval guidelines and dates](#)

**Components and teams**

- [Jazz Foundation](#)
- [Repository](#)
- [Eclipse](#)
- [Visual Studio Foundation](#)
- [Web Dashboards](#)
- [Jazz REST Services](#)
- [Release Engineering](#)

**Rational Team Concert**

**Planning**

- [Rational Team Concert 2.0 Plan Items](#) (view with the settings View As -> Tree, Group By -> Folders, Sort By -> User Defined Order)
- [Rational Team Concert 2.0 current milestones](#) - [RC2](#)
- [Other planning topics including EMC mandates](#)
- [Weekly status](#)
- [How to setup group chat](#)

**How RTC adopts Jazz Foundation**

**How you can help keep the build farm healthy**

**Components and teams**

- [Block Items](#)
- [Source Control](#)
- [Build](#)
- [Eclipse UI Foundation](#)
- [System/Network Collaboration](#)
- [Agent Planning](#)
- [OS/OS Connectors](#)
- [Reports](#)
- [Data Access](#)
- [Code Coverage](#)
- [User Assistance](#)
- [UI Design](#)
- [System Verification Test \(SVT\) Team](#)
- [Performance and Reliability Team \(PRT\)](#)
- [Visual Studio Client](#)
- [Integrating Jazz and Social Software](#)

**Extending Rational Team Concert**

- [2.0](#)
  - [Rational Team Concert 2.0 SDK](#)
  - [1.0 - 1.0.1.X](#)
    - [Rational Team Concert 1.0 SDK](#)
    - [Jazz Labs](#)

**Collaborative ALM**

- [CALM with JAZZ, RSC, RTC, RDM](#)

**Jazz Integration Architecture (JIA)**

- [Jazz integration Architecture home page](#)

**CALM Green threads**

- [CALM Configurations](#)
- [CALM Script scenario](#)
- [Enterprise ALM scenario](#)
  - [Enterprise ALM scenario configuration with ClearQuest and CALM with Jazz](#)
  - [Enterprise ALM scenario configuration with ClearCase and CALM with Jazz](#)
  - [Enterprise ALM scenario configuration with Asset Management](#)

— Page owned by RTC and JF PMCs (maintained by [Jim Dea](#), [Bryan](#); last updated 27 Feb 2009)

Please see our [Terms of Use](#).

[Edit](#) | [View Source](#) | [View](#) | [Backlinks](#) | [Web](#) | [All Views](#) | [History](#) | [131](#) < [130](#) < [129](#) < [128](#) < [127](#) | [More topic actions](#)

# Questions



# Jazz & Rational Team Concert:

Development Solution for Collaboration, automation and extensibility

