Empower Software Delivery using Rational Collaborative Lifecycle Management Solutions

Sreerupa Sen
Architect/Developer, Rational Team Concert

Next NOW!

Disclaimer

- © Copyright IBM Corporation 2010. All rights reserved. These materials are intended solely to outline our general product direction and should not be relied on in making a purchasing decision. Information pertaining to new product is for informational purposes only, is not a commitment, promise, or legal obligation to deliver any material, code or functionality, and may not be incorporated into any contract. The development, release, and timing of any features or functionality described for our products remains at our sole discretion. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM products. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

Next NOW!

Outline

- **Application Lifecycle Management**

  – IBM Rational Collaborative Lifecycle Management

- IBM Rational Tools for CLM

- Delivering the CLM Imperatives

What is Application Lifecycle Management?

Application lifecycle management coordinates people, processes, and information in an iterative cycle of software and systems delivery activities that:

✓ **Meets domain-specific needs**

✓ **Improves collaboration across teams and geographies**

✓ **Helps meet compliance requirements**

✓ **Provides the foundation for continuous capability improvement**

✓ **Reduces total cost of ownership**.

The Five Imperatives for
<u>Effective</u> Application Lifecycle Management
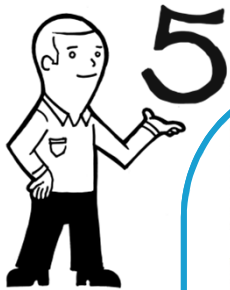
Maximize <u>product value</u> with **In-Context Collaboration**

Accelerate <u>time to delivery</u> with **Real-Time Planning**

Improve <u>quality</u> with **Lifecycle Traceability**

Achieve <u>predictability</u> with **Development Intelligence**
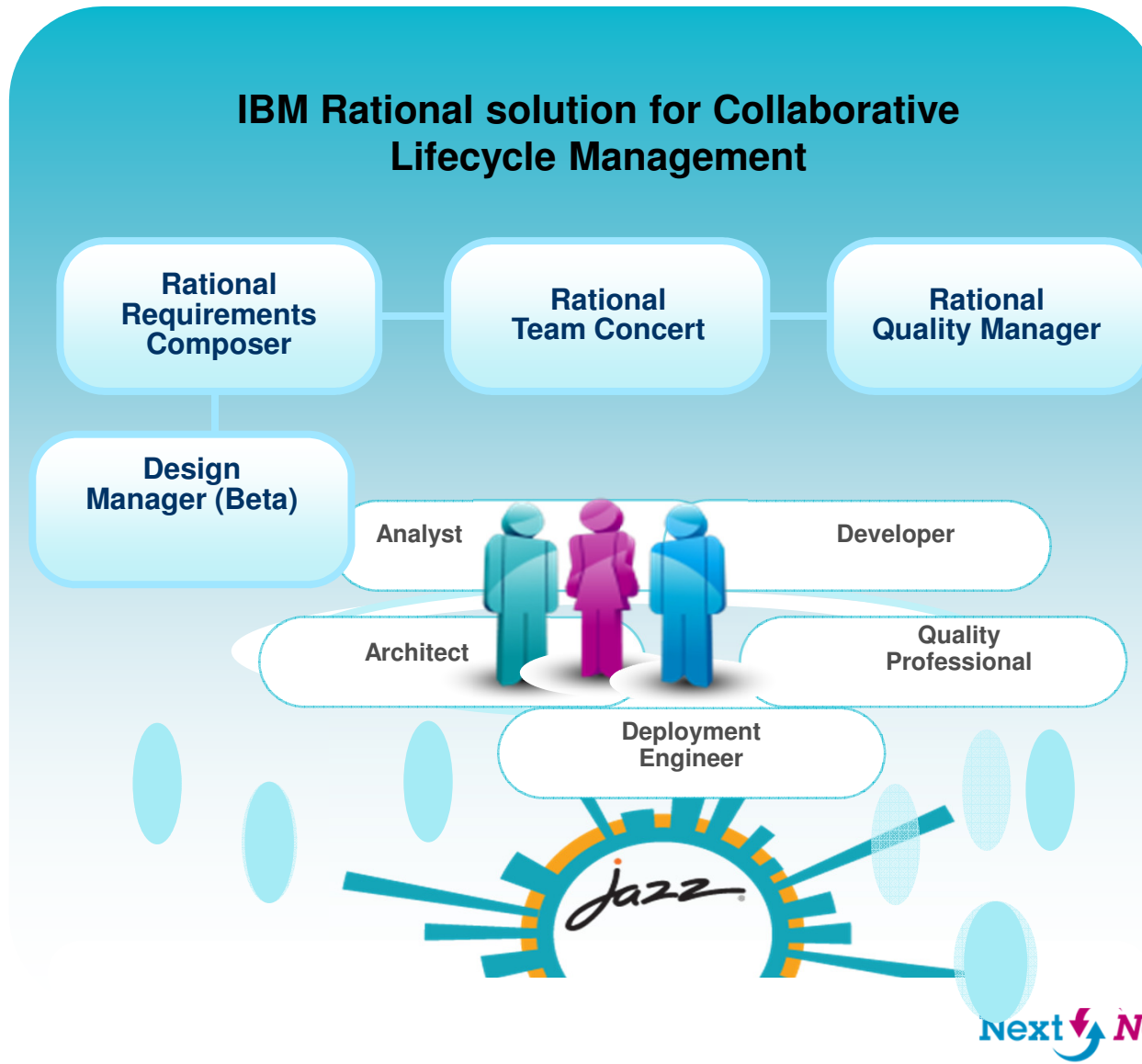
Reduce <u>costs</u> with **Continuous Improvement**

Learn more at ALM Everywhere
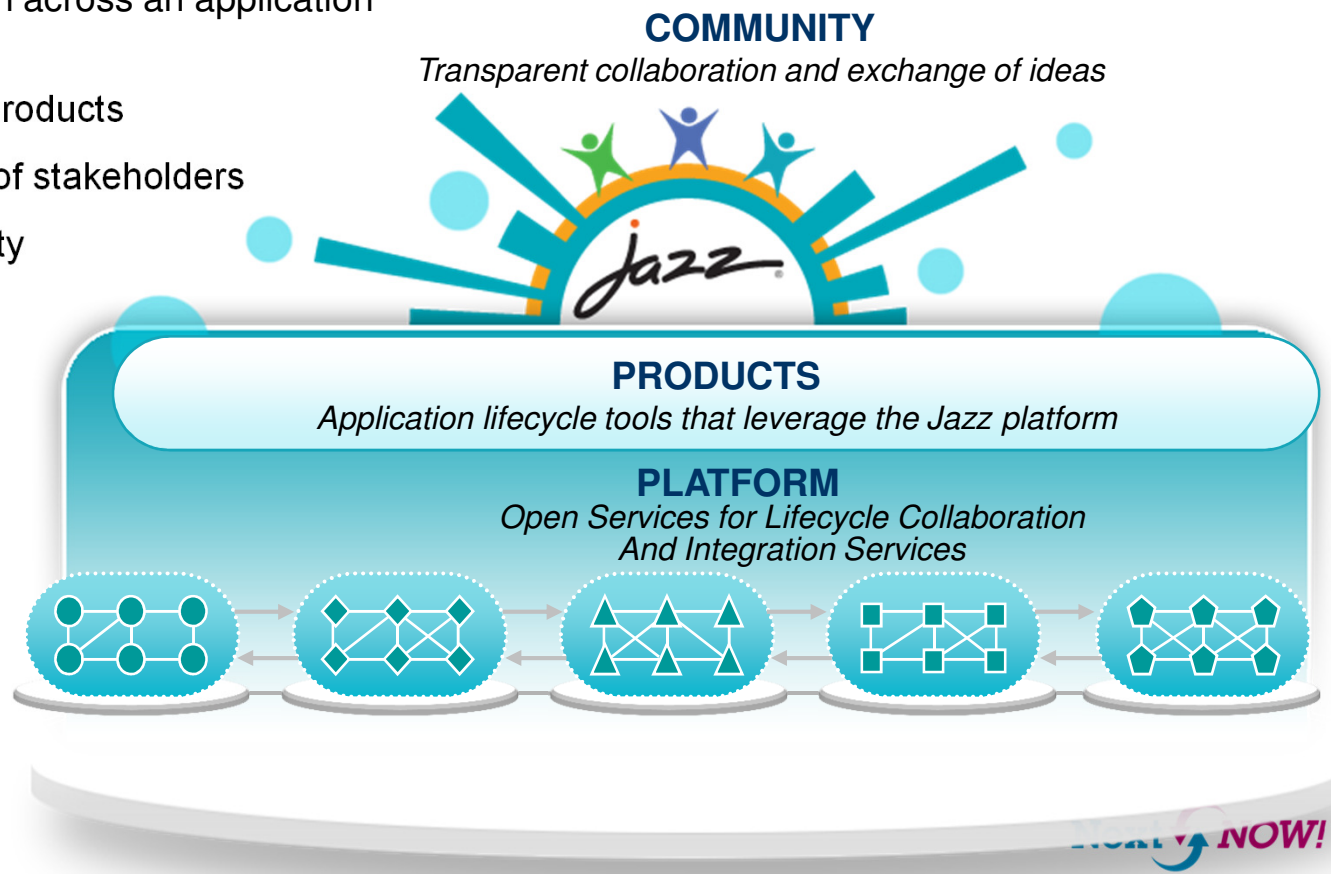
IBM Rational Collaborative Lifecycle Management
Modular, open and extensible

IBM Rational Collaborative Lifecycle Management Tools

IBM Rational solution for Collaborative Lifecycle Management

Rational Requirements Composer

Rational Team Concert

Rational Quality Manager

Design Manager (Beta)

Analyst

Developer

Architect

Quality Professional

Deployment Engineer

jazz

Next NOW!
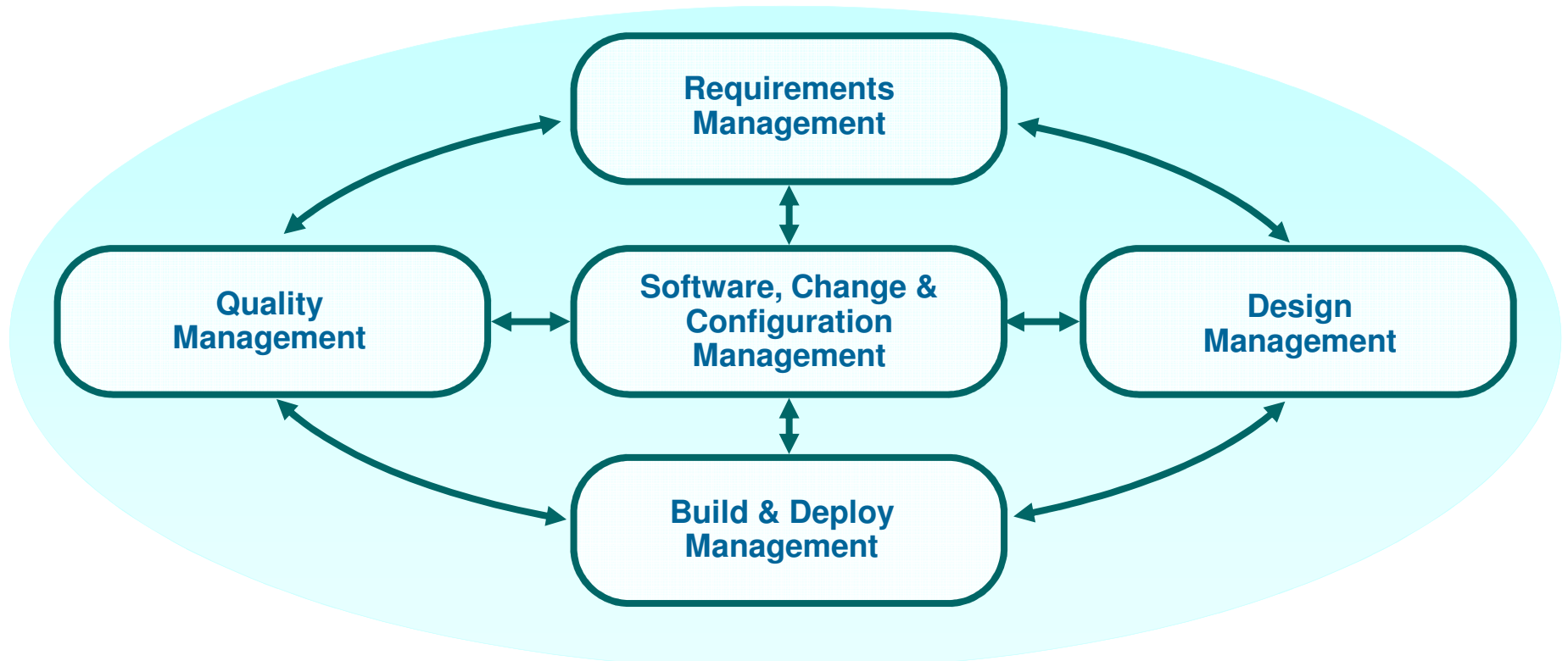
The Vision of Jazz

- Jazz is an architecture, a platform, a set of products that's designed to put teams first

  – Open, flexible standards

  – Transparency

  – Collaboration across an application lifecycle

  – Portfolio of products

  – Community of stakeholders

  – Heterogeneity

A Focus shift
Function First Thinking => Team First Thinking
Individual Productivity => Team Productivity
Desktop Integration => Lifecycle Integration

COMMUNITY
Transparent collaboration and exchange of ideas

PRODUCTS
Application lifecycle tools that leverage the Jazz platform

PLATFORM
*Open Services for Lifecycle Collaboration
And Integration Services*

Open Services for Lifecycle Collaboration (OSLC)

An initiative aimed at simplifying data linking and tool integration across the lifecycle

Open Services for Lifecycle Collaboration

Barriers to sharing resources and assets among tools

- Multiple vendors, open source projects, and in-house tools
- Private vocabularies, formats and stores
- Entanglement of tools with their data

▶ Community Driven – specified at **open-services.net**

▶ Specifications for ALM, PLM and DevOps Interoperability

▶ Inspired by Internet architecture
- Loosely coupled integration with "just enough" standardization
- Common resource formats and services

▶ A different approach to industry-wide proliferation

Learn more at: http://open-services.net/

Next NOW!

Outline

- Application Lifecycle Management
  - IBM Rational Collaborative Lifecycle Management

- **IBM Rational Tools for CLM**

- Delivering the CLM Imperatives

CLM 2012 Architecture
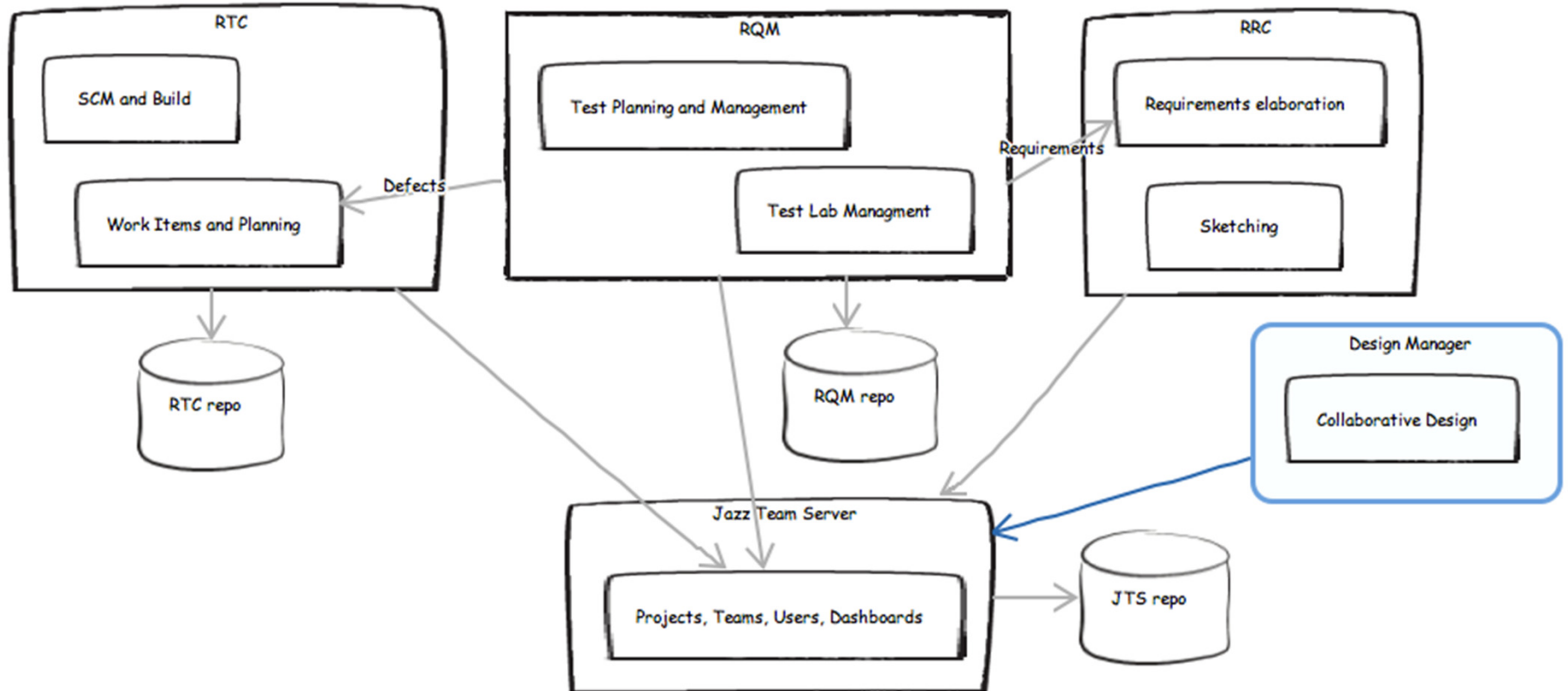
CLM on jazz.net

Our tools help you plan, design, develop, and deliver better software and systems

Agile Development

✓ Task tracking
✓ Source control
✓ Agile planning

🌐 Rational Team Concert
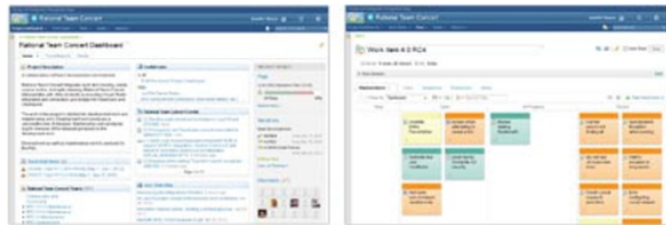
Requirements

✓ Capture business needs
✓ Manage requirements

◆ Rational Requirements Composer

Quality Testing

✓ Get products ship-ready
✓ Plan, execute, and report on testing

◆ Rational Quality Manager

Design

✓ Collaborate on architecture and design
✓ Manage design change

🌐 Rational Software Architect
🌐 Rational Rhapsody

Our products combine seamlessly to improve collaboration across the lifecycle and can integrate with a wide range of industry tools through the Jazz Platform.

Lifecycle management for the whole team

🌐 Rational Collaborative Lifecycle Management
🌐 Rational Systems and Software Engineering

Check out all the Jazz products >

Next NOW!

Rational Requirements Composer – requirements management

- **Capture your requirements**
  - UI mock-ups, story boards, rich text editors, predefined templates

- **Visually define the requirements**
  - Use cases, diagrams…

- **Collaborate and review**
  - Review comments, approvals

13

Rational Requirements Composer

- **Traceability**
  - Connect project requirements, scenarios, test artifacts, development work items
  - Identify gaps
  - Impact changes
  - Release coverage
- **Reports**
  - Project progress, traceability, status
- **Live dashboards**
- **Audit history**
  - What changes were made?
  - Who made the changes?

Next NOW!

© 2012 IBM Corporation

Rational Team Concert – Change and Configuration management

- **Track your tasks**
  - Smartly linked work items
  - Linked to each other, to test cases, to requirements
- **Plans your way**
  - Agile, waterfall, hybrid
  - Impact
  - Prioritization via ranking
- **Version control your source code**
  - Share, update, review, suspend, resume, load.
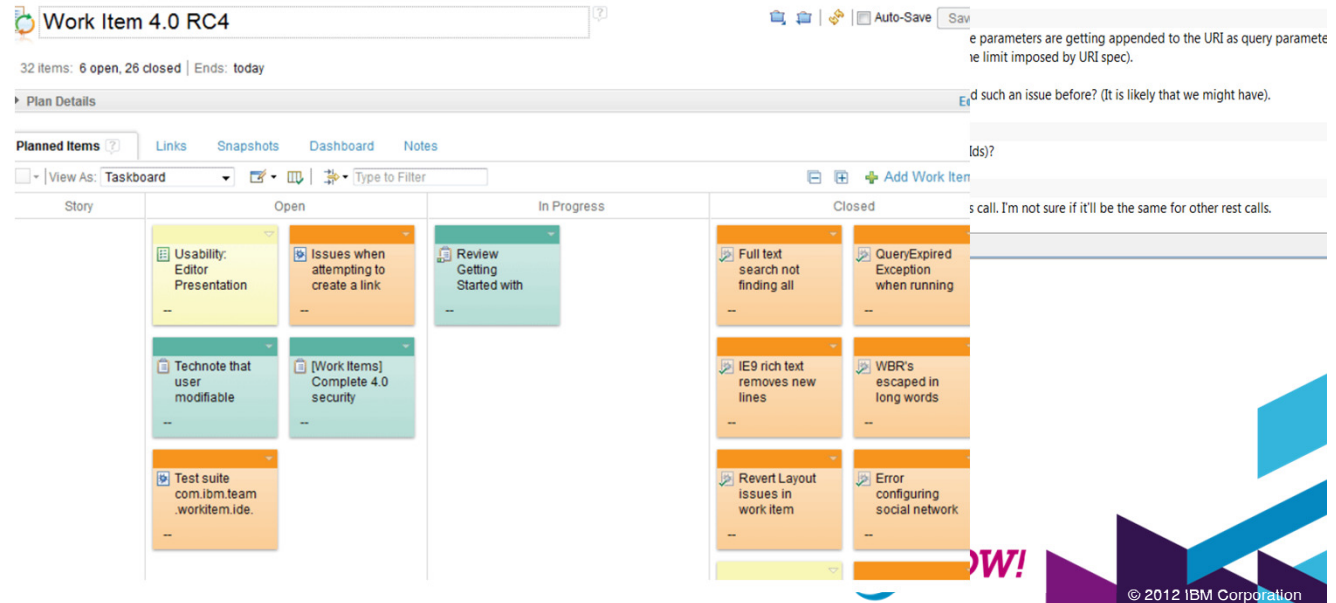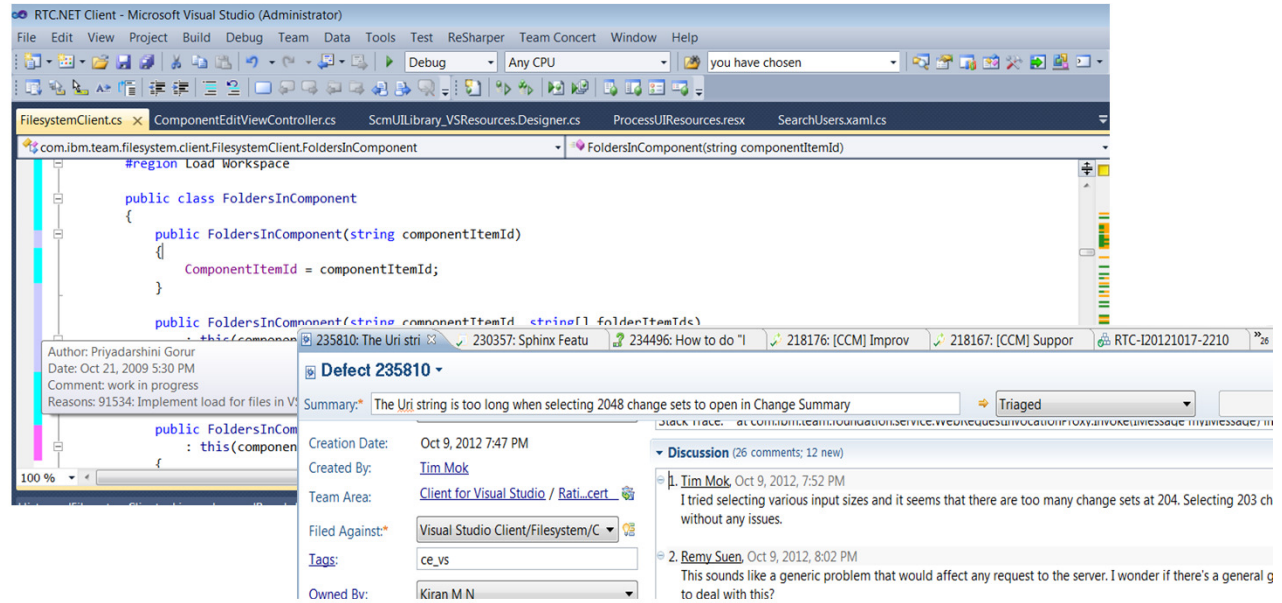
Rational Team Concert – Change and Configuration management

- **Continuous integration**
  - Continuous builds at multiple levels

- **Process Rules!**
  - Dynamic, continuously improving process

- **Team aware**
  - Day 1: up, running, coding and productive!

- **Reports and dashboards**
  - Configurable and customizable
  - Project health and team information at a glance

- **Heterogeneous teams**
  - Linux to Windows, Mac t Mainframe, Eclipse to Visual Studio

Burndown

- Remaining Work
- Planned Work
- Ideal

Next NOW!
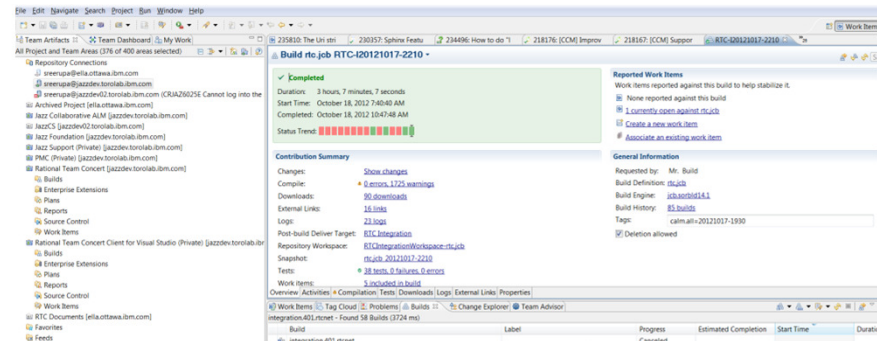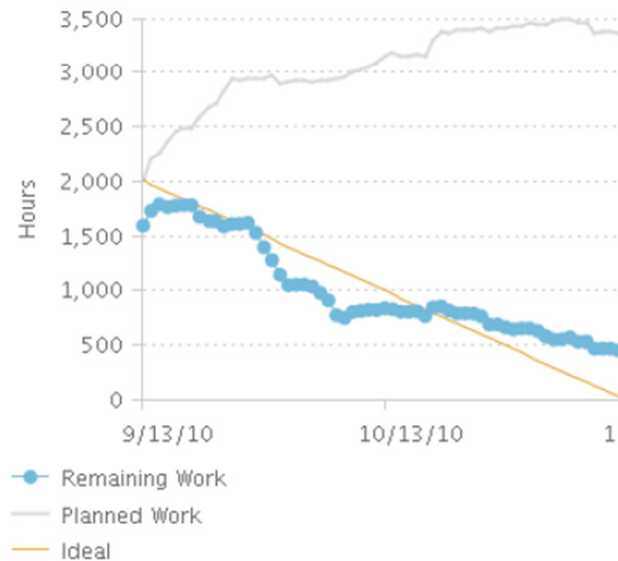
Rational Quality Manager – Test Management

- **Comprehensive test plans**
  - Quicker manual testing

- **Process awareness**
  - Customizable process
  - Automatic detection of process violation

- **Requirements driven testing**
  - Requirements linked to test cases and defects

- **Reporting, dashboards**
  - Test metrics, defect counts, trend charts

© 2012 IBM Corporation

Rational Collaborative Design Manager – Design Management

- **Design in context**
  - Work seamlessly with analysts, developers and testers
  - Manage your design artifacts as a part of your lifecycle project

- **Collaborate**
  - Sketches
  - Comments, design markers, automated design reviews

- **Manage change**

- **Traceability and impact analysis**
  - Link designs to requirements, test artifacts, work items, other designs
  - Visualize relationships
  - Impact of change

- **Dashboards**
  - Recent links, comments, most active design reviews
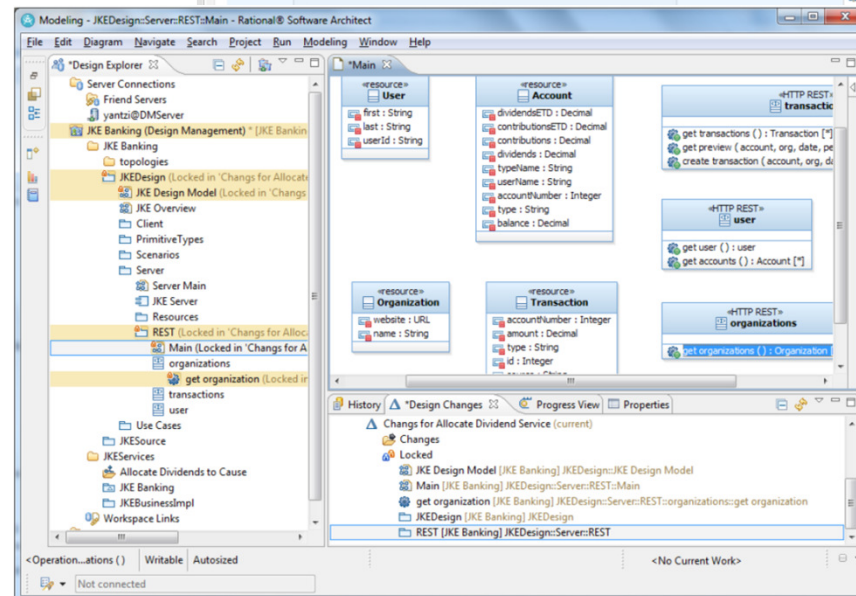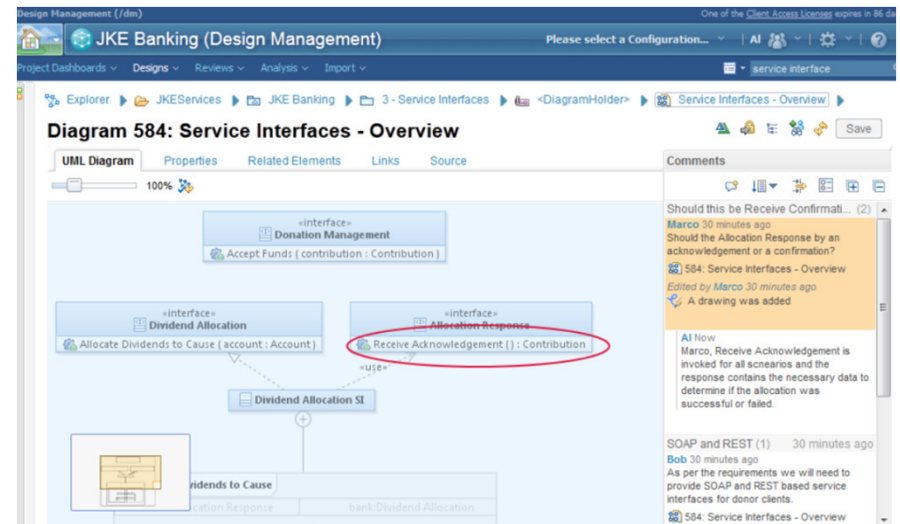
18

Outline

- Application Lifecycle Management
  - IBM Rational Collaborative Lifecycle Management

- IBM Rational Tools for CLM

- **Delivering the CLM Imperatives**

Delivering on the CLM Imperatives

- **In-context collaboration**
  - Collaborate on designs, tests, requirements, plans, code and tasks
  - Make decisions from a single source of truth

- **Lifecycle traceability**
  - Link your design, requirements, test plans and tasks

- **Continuous improvement**
  - Let the tools adapt to your process, your team

- **Real-time planning**
  - Plans linked to real work

- **Development intelligence**
  - Set measurable goals and track them easily.

What is Real-time Planning?

Real-time Planning accelerates time to delivery by:

– Providing a **single plan** that spans requirements, development, and test, ensuring a team understanding of the overall scope of a project

– **Integrating planning with execution** ensuring the entire team understands the true project status

– Allowing **everyone to participate** in keeping the plan current and accurate

– Helping teams **respond to the unexpected** in a timely manner ensuring the team stays on schedule

One plan - Multiple views facilitate detailed analysis
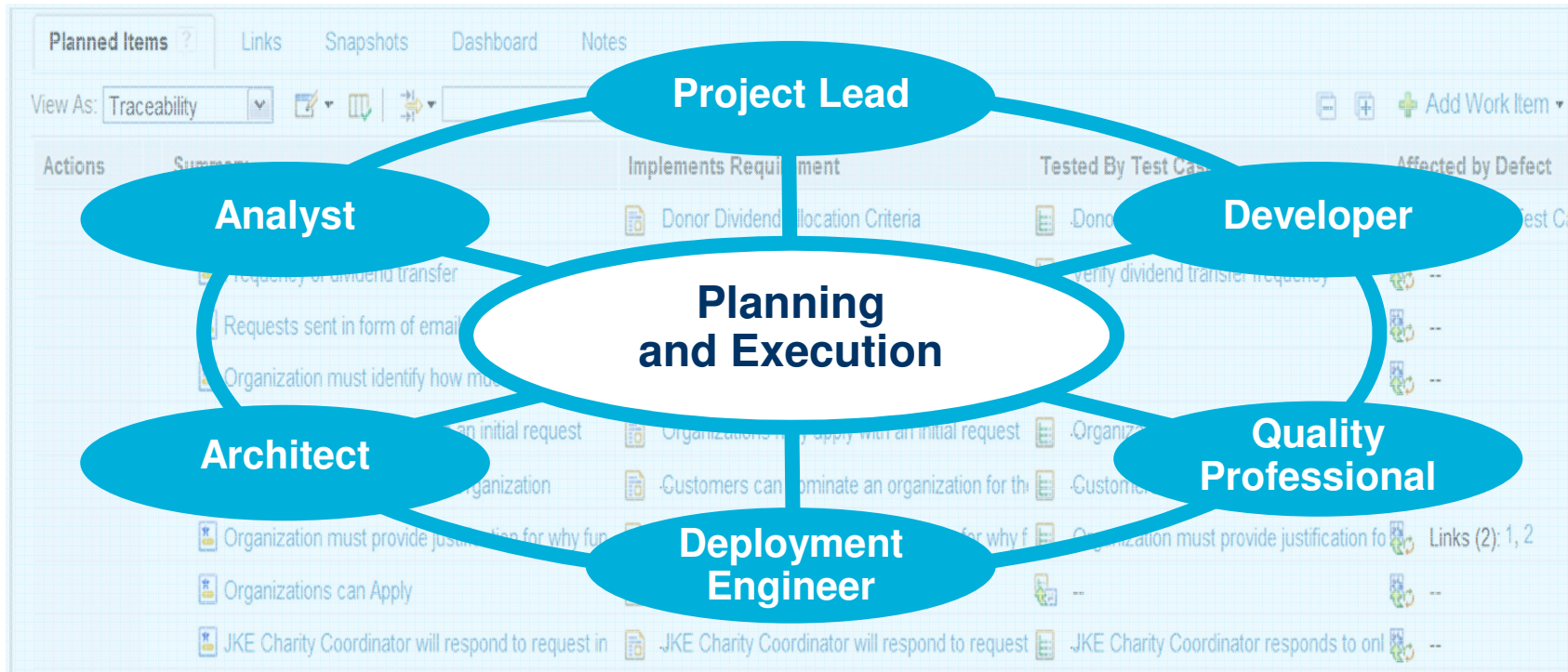
Rankings improve clarity

Team clearly understand the goals

✓ Rankings are better!

✓ Priorities are good.

Plan at the right level of detail

- Everyone participates in planning

✓ Plans include all work!
✓ All team members,
✓ All types of work

✓ Stories decomposed into the right set of tasks

Next NOW!

© 2012 IBM Corporation

Plans tied to execution

- Deliverables and Work Products show results

✓ Progress & Status are good indicators of execution.

✓ Change Sets & Attachments are better !

IBM

What is Lifecycle Traceability?

Lifecycle Traceability improves quality by:

– Establishing **relationships between software artifacts**

– Helping you **identify and close artifact gaps,** ensuring **coverage** across disciplines

– Provides visibility into the **completeness** of planned items by inspecting all related artifacts

– Provides easy **access to related artifacts** ensuring everyone shares the same view

– Delivers transparency which enables everyone to make **fully informed decisions** based business priorities

Instant access to details from any point in the development process

| Customer | Final Product | Build Artifacts | Environment | Supporting Systems | Requirements Management | Initial Idea |

Next NOW!
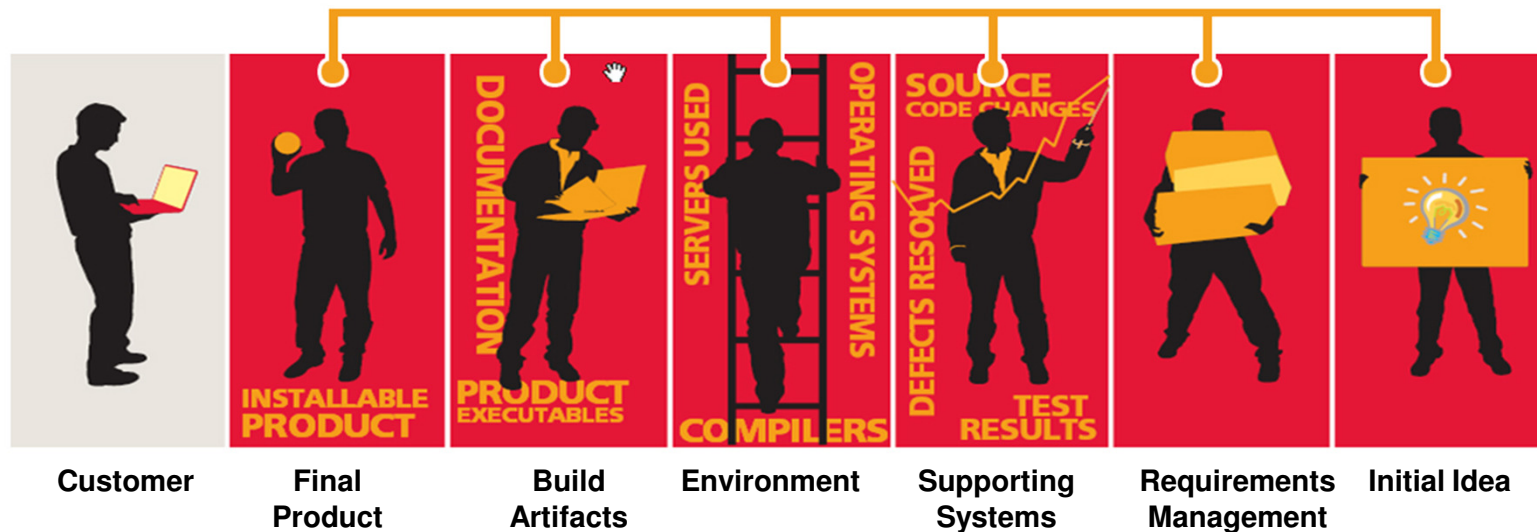
CLM Link Types

Learn more at: https://jazz.net/wiki/bin/view/Main/CALM2010LinkTypes
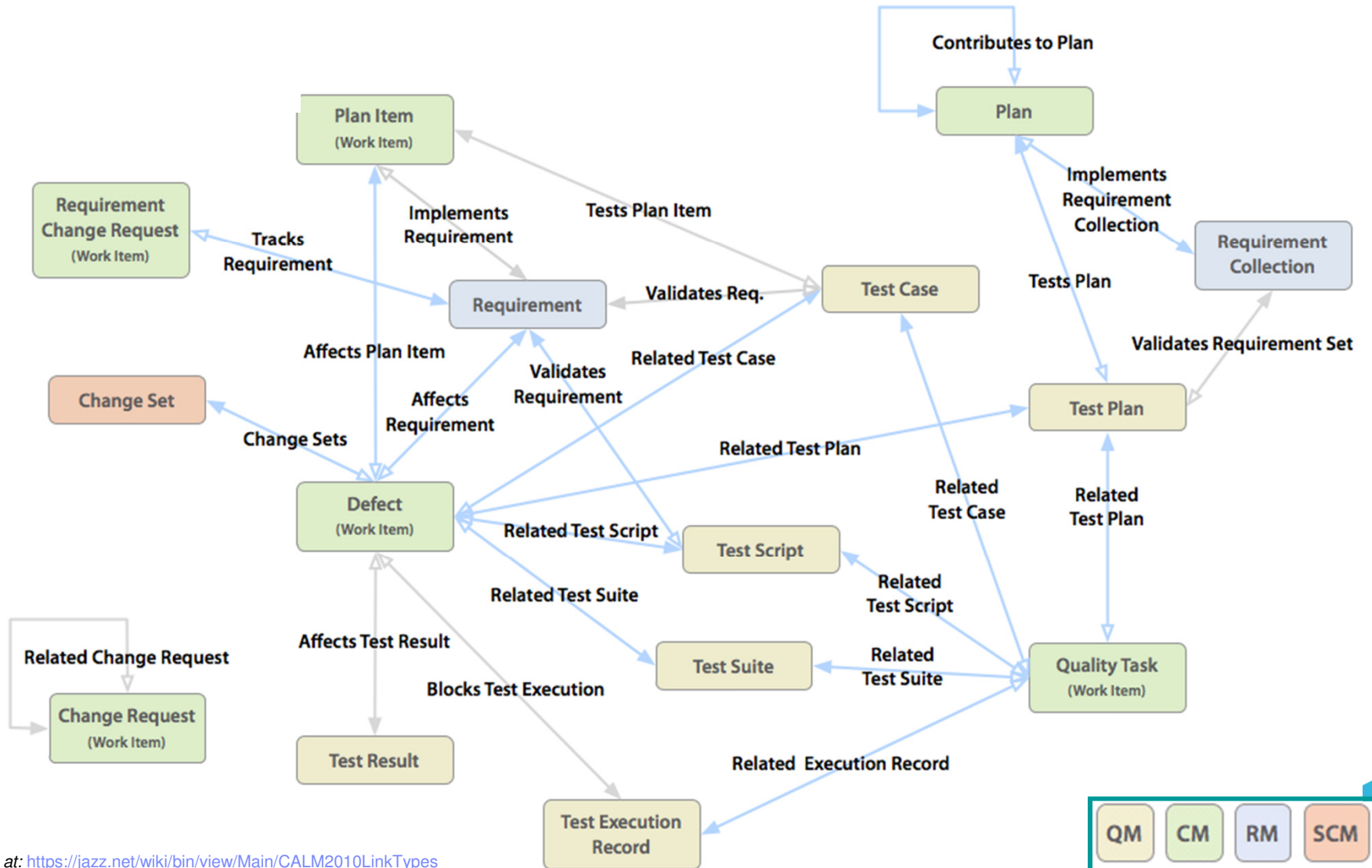
Team Leads have visibility into coverage & completeness

- Proactively respond to gaps as they surface through out the project

- Issues quickly highlighted and resolved

What is In-Context Collaboration?

In-Context Collaboration improves product value by:

- Making **information <u>immediately</u> accessible** to all team members in the context of their work

- Empowering teams to collaborate on and review software development artifacts so they can **incorporate feedback early and often**

- Providing **single source of truth** hosted in a shared repository so that team members can collaborate effectively around the globe
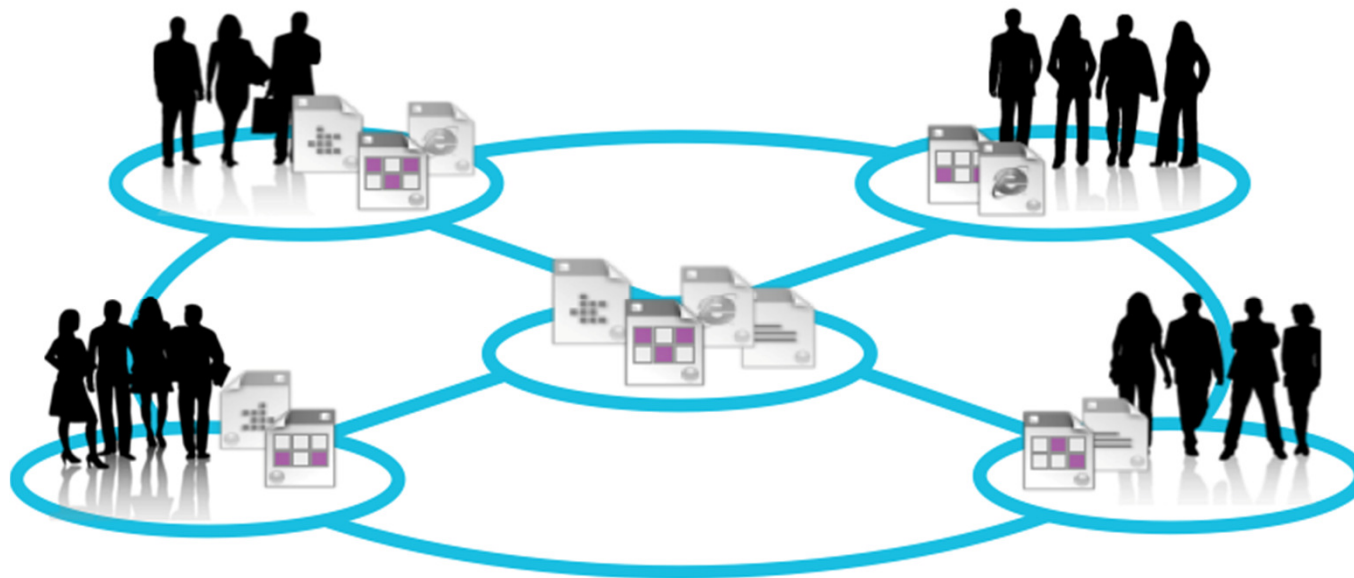
Rich hovers automate information access without having to click

✓ Rich hovers provide quick access to information to determine if additional details are required!

Next NOW!

In-context Collaboration always shows the latest

Threaded discussions on requirements

Recent Threaded discussions highlighted on requirements

Unread work Items bolded for developers

Viewlets focus the team on recent comments and changes

Requirements comment updates

Feeds focused on Requirement comments focus stakeholders on changes and strip out the noise of attributes changes etc..

Any changes to a Requirement

Feeds on all changes give project leads visibility into all requirements changes so that know exactly what is changing by whom and can access immediately assess the impact for the team.
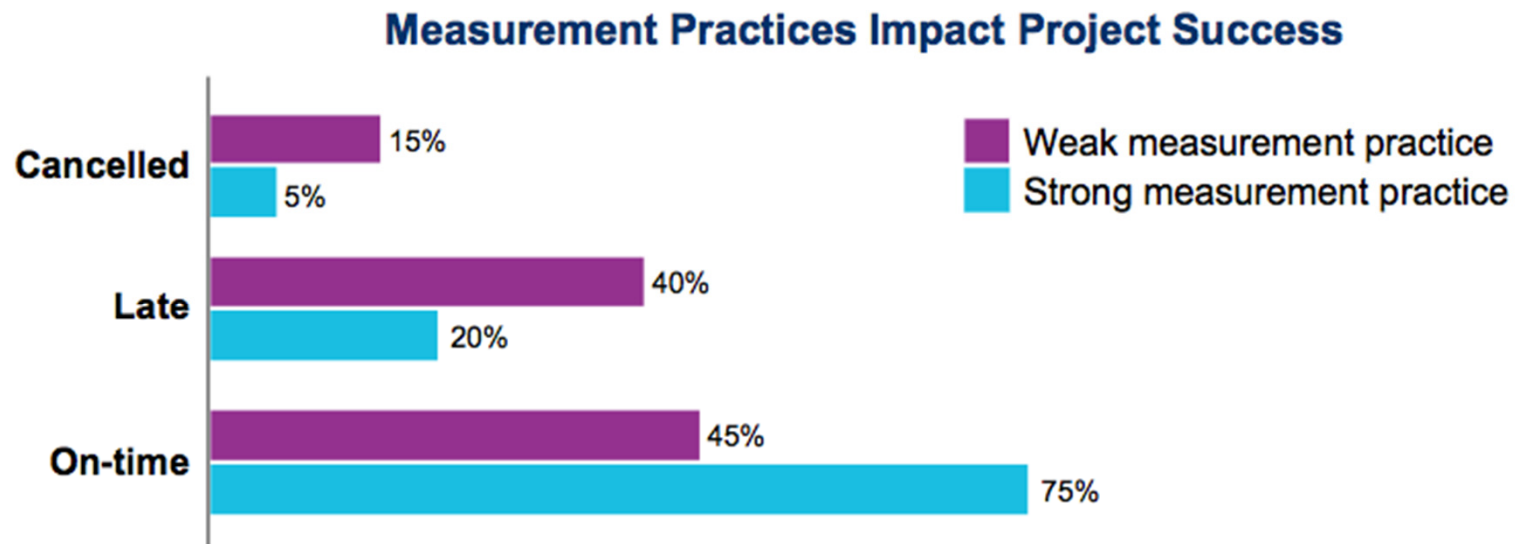
Next NOW!

What is Development Intelligence?

Development Intelligence improves predictability by:

– Applying **Business Intelligence techniques** to software and systems development

– Enabling **fact-based decision making** (to communicate status, monitor progress, diagnose problems, identify corrective actions)

– **Steering projects and programs** to deliver on-time

Measurement Practices Impact Project Success

	Weak measurement practice	Strong measurement practice
Cancelled	15%	5%
Late	40%	20%
On-time	45%	75%

Source: Capers Jones, Measurement, Metrics and Industry Leadership, 2009 and Software Engineering Best Practices, McGraw Hill, 2010.

Next NOW!

Measuring Time-to-Value (Schedule)

Iteration Burndown

Showing how much work is left to do in an iteration. It enables the team to adjust scope or resources to finish the iteration successfully.
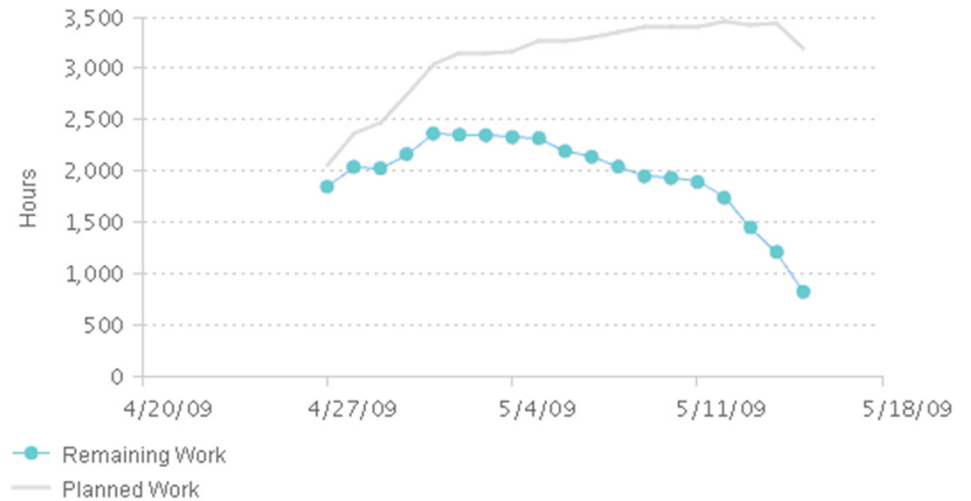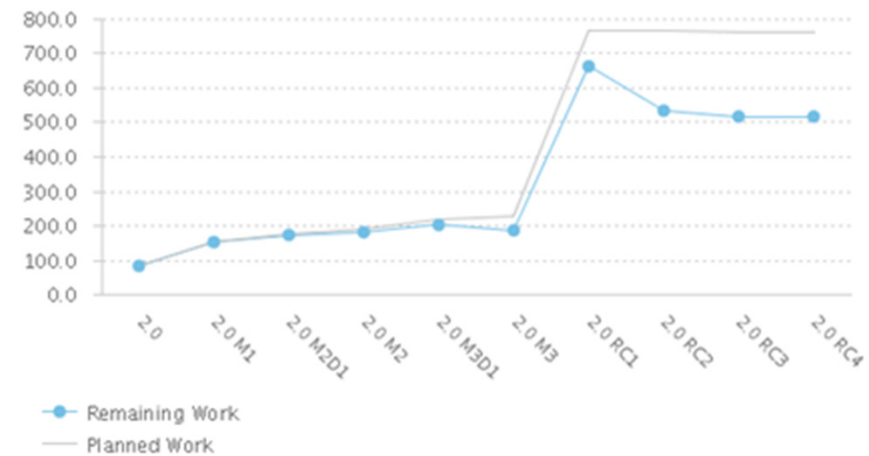
Release Burndown

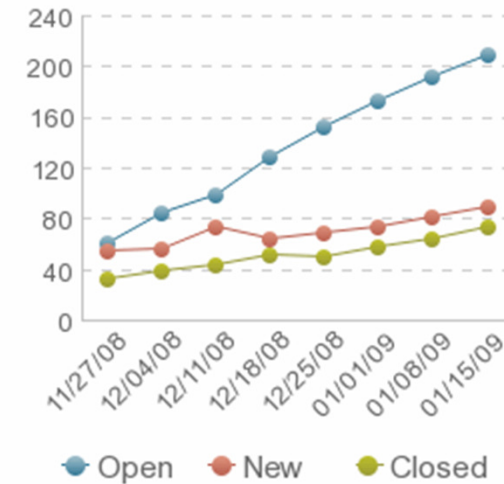Shows the estimated functionality remaining to complete the current release.

Measuring Product Quality

Defect Trends

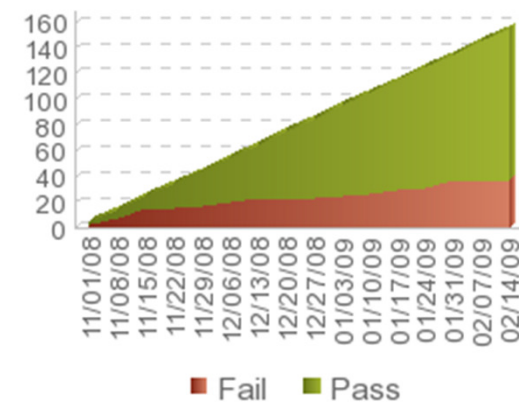Shows defect arrival and closure rates, determines the remaining defect backlog, projects the future defect arrival/close rate up to and post-ship

Number of Defects

Test Execution Status

Monitors test completion and success

Number of Executed Tests

Use dashboards to provide that 1 view of project health

What is Continuous Improvement?

Continuous Improvement reduces cost by:

– Improving software delivery through the ongoing **adoption of best practices and automation** to reduce manual, non-creative and error prone tasks

– Promoting **incremental improvement of a project** when needed

– Enabling breakthrough improvement by **capturing best practices and reusing** across teams

– Allowing **everyone to participate** with easy to adopt best practices at your fingertips.

Assess

ROADMAP

Act

Steer

jazz

"Successful analytics requires taking it beyond software and reporting, and into the realm of management practices and operations improvement."

Information Management Online, February 23, 2011

Next NOW!

Adopt - Process Sharing

Leverage the Best practices from Rational and other teams

Process Templates on Server: JKE.tools.com:9443/JTS

1 **Import/export Templates**
Provides the ability to leverage best practices from Rational and other teams

Scrum Iterative Traditional

2 **Servers support multiple templates**
Allowing administrators to support different processes.
Teams improve their process independently.

Web Banking

Team : **Core**

Team : **UI**

Team : **Benefits**

avings

etail

3 **Process updates can be shared**
Allows teams to control when they want to accept improvements.

Next NOW!
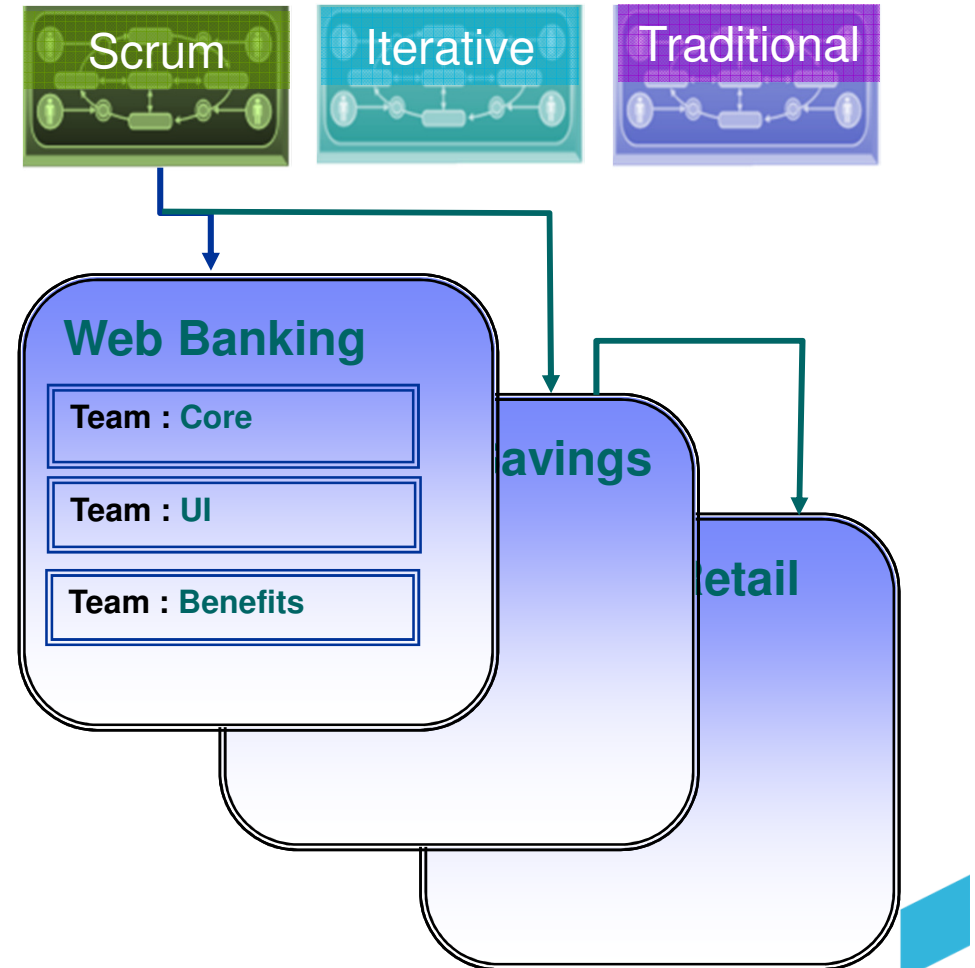
Team Retrospectives help identify areas of improvement

- **Tracking Retrospectives help team members to collaborate on improvements**

  – Ranking of process improvements help to focus on where the team feels the biggest pain.

  – Metrics where available help support process improvements and show results.

Implement : Require Unit test before deliver

Implement: Work Item Template for a set of repetitive tasks

Work Items ✔ *   Plans ⌄   Source Control ⌄   Builds ⌄

📋 Welcome to Work Items

Queries
- 📇 My Queries
- 📇 Shared Queries

Create Query
- 📇 Create Query

Create Work Item
- Defect
- Task
- Story
- Epic
- Track Build Item
- Impediment
- Adoption Item
- Retrospective

Create Work Item Set
- 📇 Create From Template...

Story
(work item)

Children

Task
(work item)

Blocks

Depends on

Task
(work item)

Blocks

Depends on

Task
(work item)

Next NOW!

The Five Imperatives of Application Lifecycle Management
Improve organizational productivity

- Accelerate time to delivery with **Real-time Planning**

- Improve quality with **Lifecycle Traceability**

- Maximize product value with **In-Context Collaboration**

- Refine predictability with **Development Intelligence**

- Reduce costs with **Continuous Improvement**

Learn more at: jazz.net/library/article/637

Next NOW!

www.ibm.com/software/rational

www.ibm.com/software/rational

Next NOW!

© 2012 IBM Corporation