

Rational Team Concert



Introduction to Rational Team Concert

- Rational Team Concert:
 - ▶ Is a product that is based on Jazz for developers, architects, and project managers
 - ▶ Enables team members to collaborate with integrated iteration planning, work-item management, source control, build management, dashboards, and reports
 - ▶ Supports process control and customization
 - ▶ Has an Eclipse-based workbench and a Web interface
 - ▶ Is a key component of the Rational solution for Collaborative Lifecycle Management (CLM)



Project and team areas

- A project administrator or team lead sets up a project area in Rational Team Concert:
 - ▶ Creates a project area and optional team areas
 - ▶ Defines the process for this project
 - ▶ Defines the project iterations and plans
 - ▶ Defines the teams
 - ▶ Creates work items



Work items and Iteration plans

- **Work items** capture planned work for a project:
 - ▶ Describe requirements, defects, and feature improvements
 - ▶ Identify other tasks that are related to project development
- The types of work items available in a project area or team area are defined in the process configuration.
- The project administrator or team lead organizes work into a series of iterations, called an iteration plan.
 - ▶ An iteration is generally bound to a specific time period and has a well-defined scope of work items to completed.
 - ▶ The process can be adjusted, based on iteration plans. Additional approvals might be required to introduce a new feature late in the development cycle.



Rational Team Concert workbench

- The Rational Team Concert workbench refers to the Eclipse-based desktop development environment.
- The Workbench consists of these elements:
 - ▶ **Perspectives**, a group of views and editors
 - ▶ **Views**, the element that you use to navigate a list or hierarchy of information, or display properties for the active editor
 - ▶ **Editors**, the element that you use to edit or browse a resource



Team Artifacts view

Use the Team Artifacts view to access, create, and manage these items:

- Repository connections
- Project areas
- Team areas
- Work-item queries
- Artifacts

The screenshot displays the Rational Team Concert interface for the project "JKE Banking (Change Management)". The left sidebar, titled "All Project and Team Areas (4 of 4 areas selected)", is circled in orange and contains the following items: Repository Connections, JKE Banking (Change Management) [vmuseridxxx], Builds, Enterprise Extensions, Plans, Reports, Source Control, Work Items, Favorites, Feeds, My Repository Workspaces, My Team Areas, and Work Item History. The main area shows details for "Task 43" (Implement - Requests sent in form of email). The details include: Type: Task, Filed Against: JKE/BRM, Team Area: Business Recovery Matters, Creation Date: May 27, 2011 3:56 AM, Created By: Marco, Tags: (empty), Owned By: Deb, Priority: Medium, Planned For: -> Sprint 2, Estimate: 4 d. The bottom pane shows a list of work items with columns for ID, Status, Priority, Severity, Summary, and Owned By. The list contains 10 work items, with the first four visible:

I..	Status	P	S	Summary	Owned By
43	New	Medium	Low	Implement - Requests sent in form of email	Deb
41	New	Medium	Low	Implement - Frequency of dividend transfer	Deb
27	New	Medium	High	Improve link colors	Deb
25	New	Medium	High	Logout is not working anymore	Deb

Find work items by running a query. The Work Items folder contains queries that are private and queries that are shared with your team.

The screenshot shows the IBM Rational software interface. On the left, a tree view displays the 'Work Items' folder, which is expanded to show 'My Queries' and 'Shared Queries'. Under 'Shared Queries', the 'Predefined' folder is expanded, and the 'Open assigned to me' query is selected. An orange callout box points to this query with the text: 'Double-click a query to run it. Query results are displayed in the Work Items view.'

The main area of the interface shows the 'Work Items' view, which displays a table of work items. The table has columns for 'I.', 'Status', 'P', 'S', 'Summary', and 'Owned By'. The table contains 10 rows of work items, all with a status of 'New'. The 'Owned By' column shows 'Deb' for all items. The 'Summary' column contains the following text:

I.	Status	P	S	Summary	Owned By
43	New			Implement - Requests sent in form of email	Deb
41	New			Implement - Frequency of dividend transfer	Deb
27	New			Improve link colors	Deb
25	New			Logout is not working anymore	Deb
24	New			Performance on first startup is bad	Deb

At the bottom of the interface, there is a status bar that reads '<No Current Work>'.

Viewing a work item

The screenshot shows the 'Work Items - JKE Banking (Change Management) - Rational Team Concert' application. The main window displays 'Task 43' with the following details:

- Summary: Implement - Requests sent in form of email
- Type: Task
- Filed Against: JKE/BRM
- Team Area: Business Recovery Matters /
- Creation Date: May 27, 2011 3:56 AM
- Created By: Marco
- Tags:
- Owned By: Deb
- Priority: Medium
- Planned For: -> Sprint 2

The 'Description' section is currently empty. A 'New' button is visible in the top right of the task editor.

At the bottom, a list of work items is shown:

ID	Status	Priority	Summary	Owned By
43	New	Medium	Implement - Requests sent in form of email	Deb
41	New	Medium	Implement - Frequency of dividend transfer	Deb
27	New	High	Improve link colors	Deb
25	New	High	Logout	Deb
24	New	High	Perform	Deb

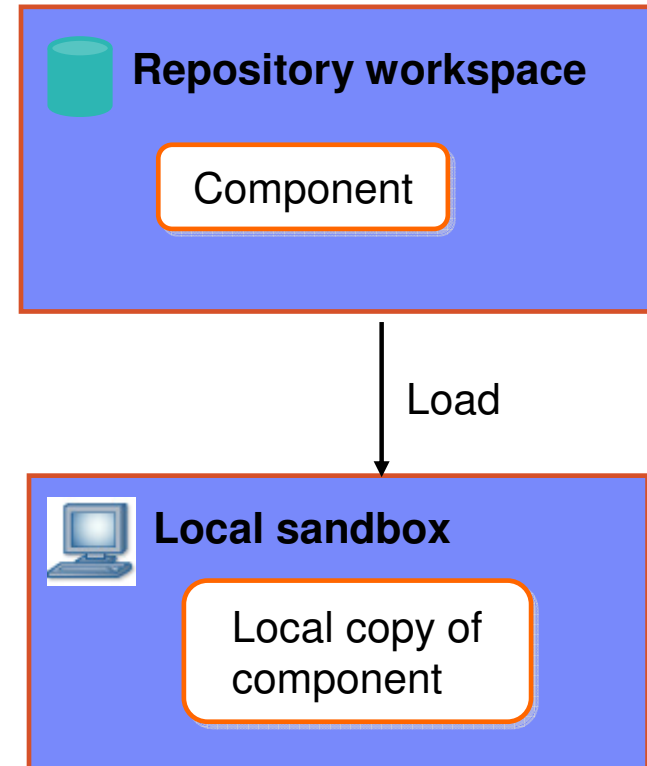
Details about the work item are listed in the Details section.

To begin work on the work item, change the state to **Start Working**.

Double-click a work item to view it in the Work Items editor.

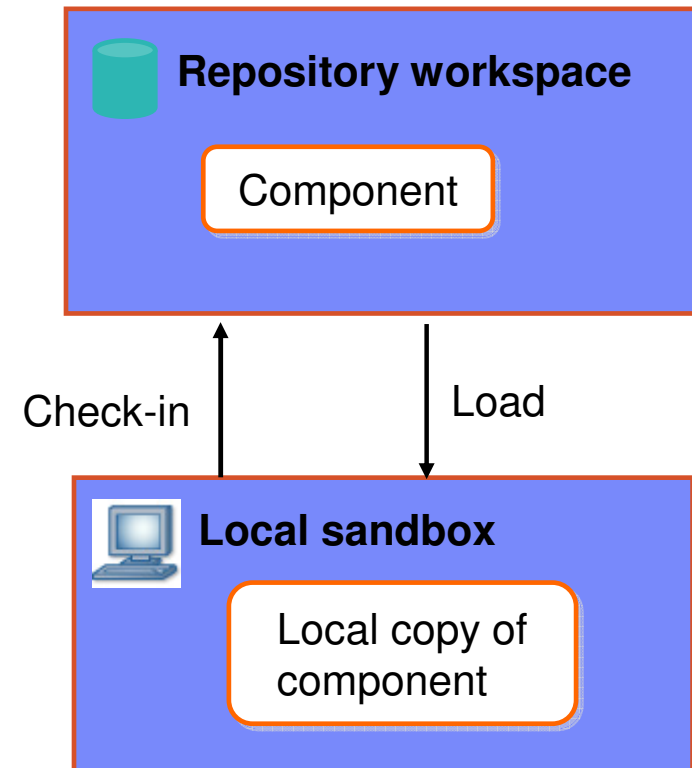
Workspace

- To work on a project, you must have a local **sandbox**, which is a directory in your computer file system where you can work with files and folders under source control. (In an Eclipse environment, this is the Eclipse workspace.)
- You load, or copy, files and folders from a repository workspace on the server to your local sandbox.



Workspace (continued)

- Changes you make in your local sandbox are stored only on your local file system
- When you check in changes, the modified files and folders are copied to your personal repository workspace on the Jazz Team Server



My Repository Workspaces

The screenshot displays the Rational Team Concert interface. On the left, the 'Team Artifacts' view shows a tree structure under 'My Repository Workspaces' containing several project components. An orange circle highlights this tree, and an arrow points from it to a text box. The main area shows the 'Repository Workspace' details for 'Deb BRM Stream Workspace', including repository name, owner, visibility, and a list of components. At the bottom, a table lists work items assigned to the user.

My Repository Workspaces (from Team Artifacts view):

- Banking Logic (4: Sprint 1 (Release 1.0))
- Build (2: Week 1 (Sprint 1, Release 1.0))
- C# UI (4: Sprint 1 (Release 1.0))
- Database (4: Sprint 1 (Release 1.0))
- Java UI (4: Sprint 1 (Release 1.0))
- Prerequisites (2: Week 1 (Sprint 1, Release 1.0))
- Web UI (3: Sprint 1 (Release 1.0))

Repository Workspace Details:

- Name: Deb BRM Stream Workspace
- Repository: deb@vmuseridxxx
- Owned by: Deb
- Visibility: Public
- Description: (empty)

Components:

- Banking Logic (4: Sprint 1 (Release 1.0)) (JKE Banking (Change Management))
- Build (2: Week 1 (Sprint 1, Release 1.0)) (JKE Banking (Change Management))
- C# UI (4: Sprint 1 (Release 1.0)) (JKE Banking (Change Management))

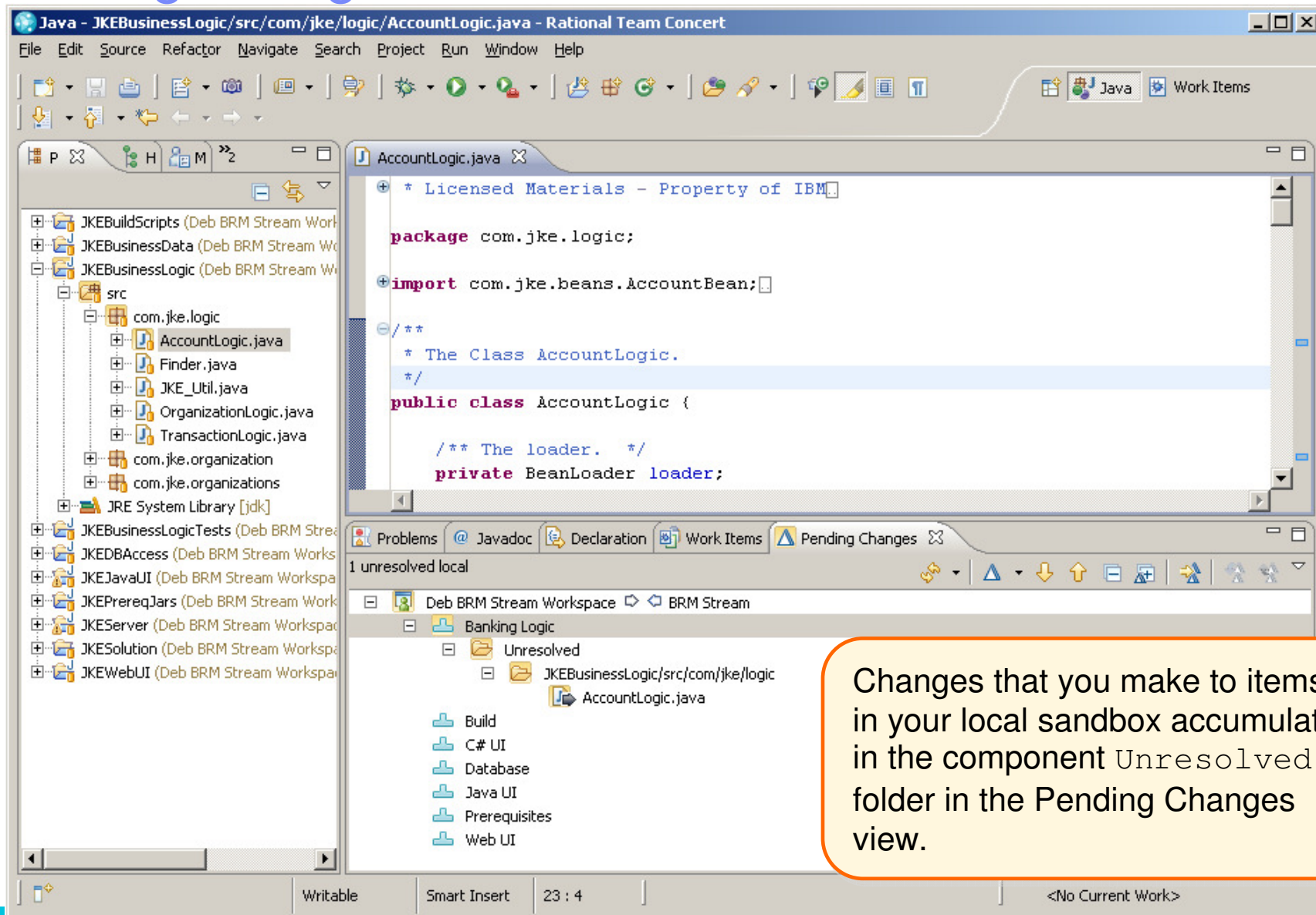
Work Items Table:

Status	P	S	Summary	Owned By
New			Implement - Requests sent in form of email	Deb
New			Implement - Frequency of dividend transfer	Deb
New			Improve link colors	Deb
New			Logout is not working anymore	Deb
New			Performance on first startup is bad	Deb

In the Team Artifacts view, My Repository Workspaces lists your repository workspaces and the project components they are associated with.



Pending changes



The screenshot displays the IBM Rational Team Concert IDE interface. The main editor window shows the source code for `AccountLogic.java`. The code includes a package declaration, an import statement, a class comment, and the start of a `public class AccountLogic` with a private `BeanLoader loader` attribute. The `Pending Changes` view at the bottom shows a tree structure with an `Unresolved` folder under `Banking Logic`, which contains the `AccountLogic.java` file. This indicates that changes made to this file in the local sandbox have not yet been resolved or committed.

```
package com.jke.logic;

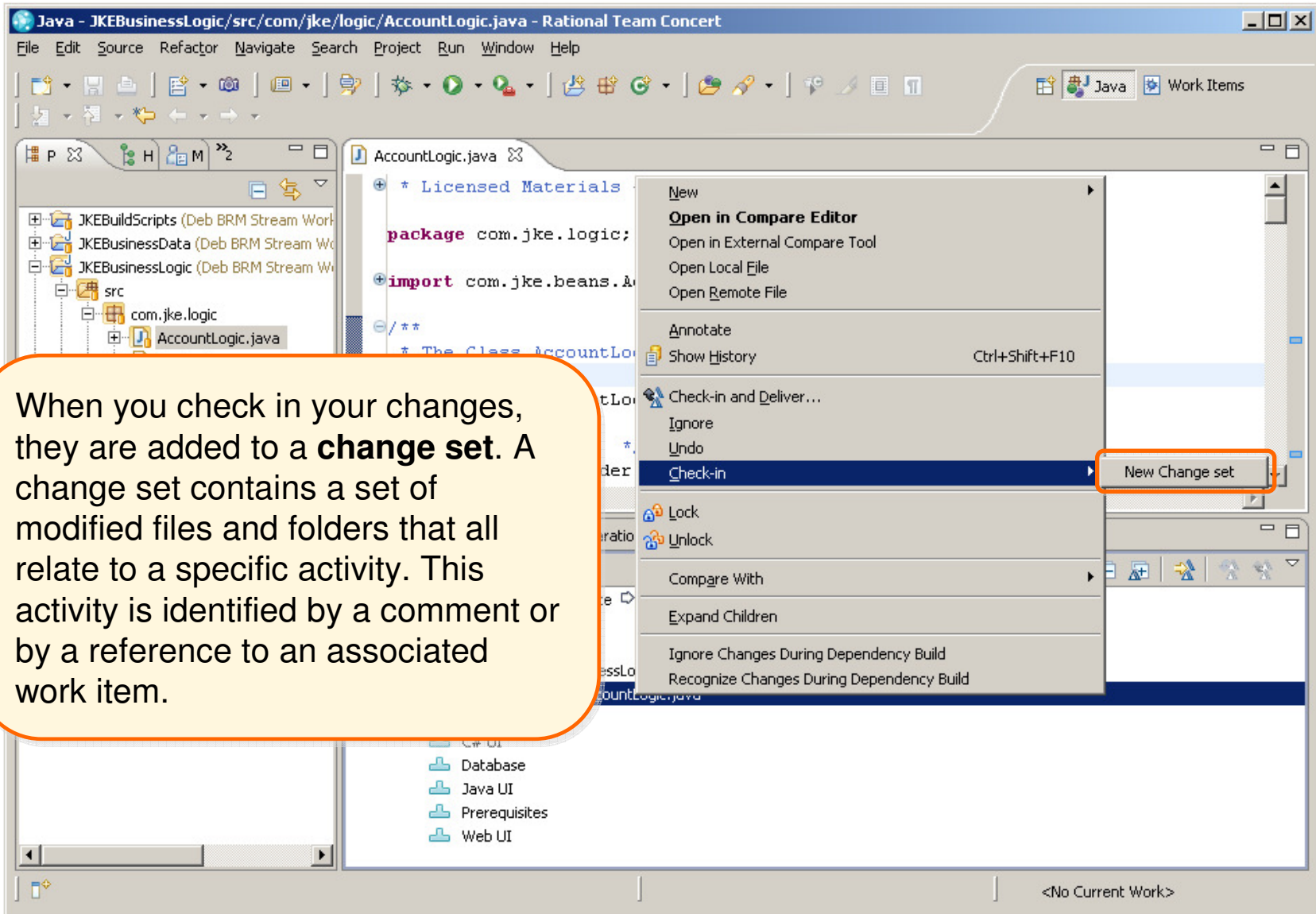
import com.jke.beans.AccountBean;

/**
 * The Class AccountLogic.
 */
public class AccountLogic {

    /** The loader. */
    private BeanLoader loader;
```

Changes that you make to items in your local sandbox accumulate in the component `Unresolved` folder in the `Pending Changes` view.

Checking in changes



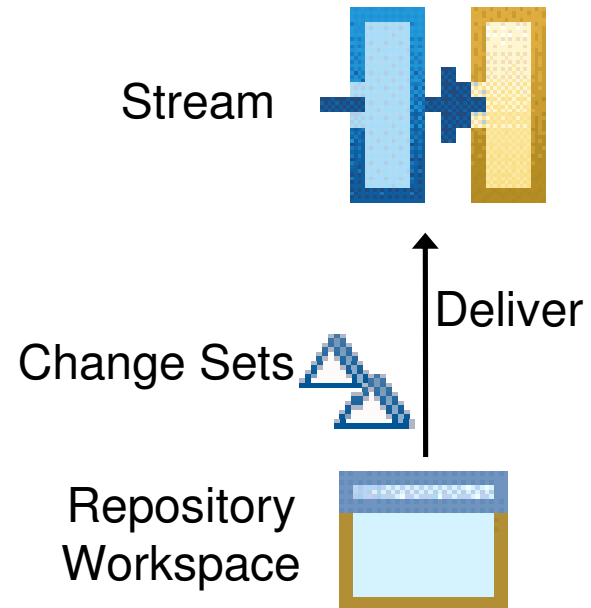
The screenshot shows the IBM Rational Team Concert interface. The main window displays the source code for `AccountLogic.java`. A context menu is open over the file, with the `Check-in` option selected. A sub-menu is visible for `Check-in`, with `New Change set` highlighted. The interface includes a menu bar (File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help), a toolbar, and a project explorer on the left showing the project structure.

When you check in your changes, they are added to a **change set**. A change set contains a set of modified files and folders that all relate to a specific activity. This activity is identified by a comment or by a reference to an associated work item.



Delivering changes to a stream

- Checked-in changes are not shared with the team until you **deliver** the change sets from your repository workspace to the team's stream.
- A **stream** is a repository object that is used to integrate the work done in developer's workspaces



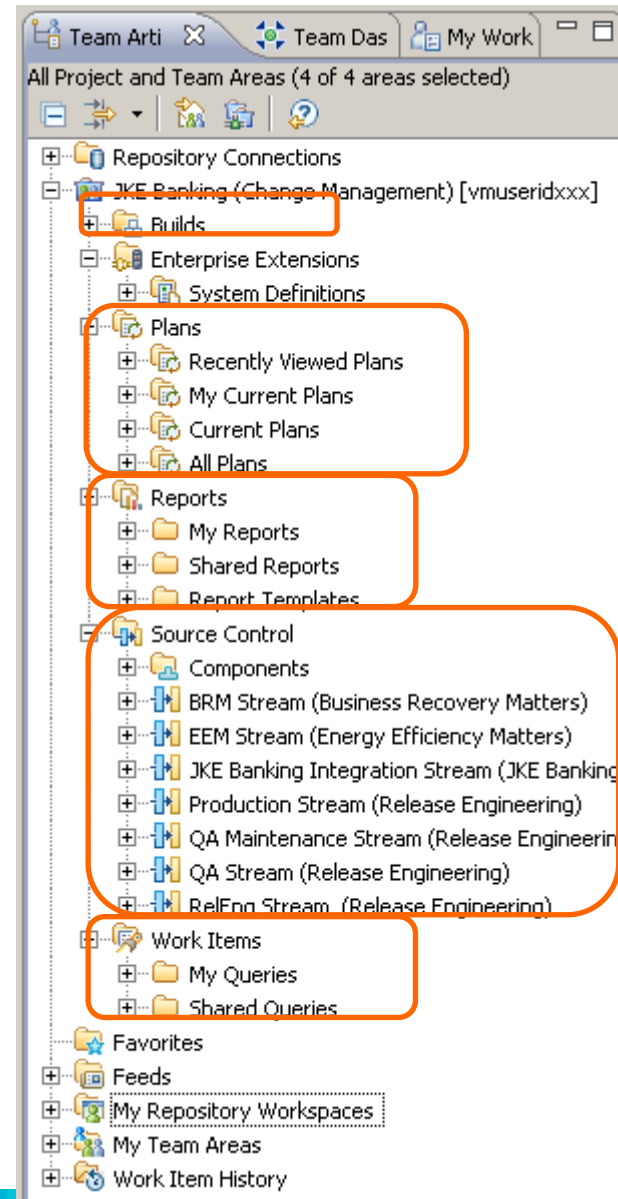
Rational Team Concert user interfaces

- Rational Team Concert client for Eclipse
- Rational Team Concert client for Microsoft Visual Studio
- Rational Team Concert web client
- “scm” command-line interface



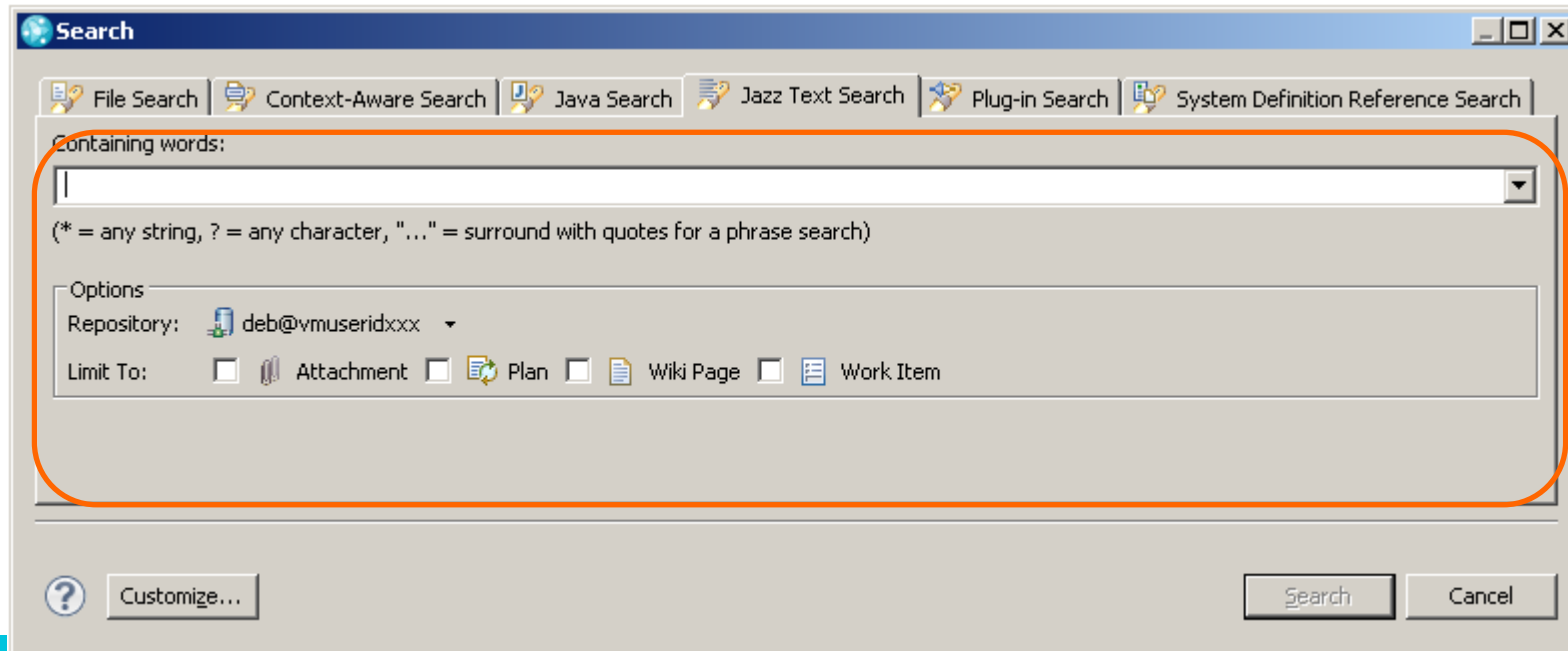
Jazz/RTC capabilities in the Eclipse client

- ▶ Artifact management
 - ▶ Build management
 - ▶ Project planning
 - ▶ Reporting
 - ▶ Source control
 - ▶ Work-item management



Jazz and Rational Team Concert capabilities in the Eclipse client

- ▶ Repository search
 - ▶ Search an entire repository
 - ▶ Limit a search to these elements:
 - ▶ Attachments
 - ▶ Plan items
 - ▶ Wiki pages
 - ▶ Work items



Rational Team Concert web client

- Ideal for project managers, stakeholders, and other contributors who are not using a supported integrated development environment
- Provides access to these capabilities:
 - ▶ Dashboards
 - ▶ Work-item management
 - ▶ Project planning
 - ▶ Source control
 - ▶ Build management
 - ▶ Reporting



Jazz and Rational Team Concert capabilities in the web client

► Dashboards

The screenshot shows the Rational Team Concert web client interface for the 'JKE Banking (Change Management)' project. The dashboard is organized into several sections:

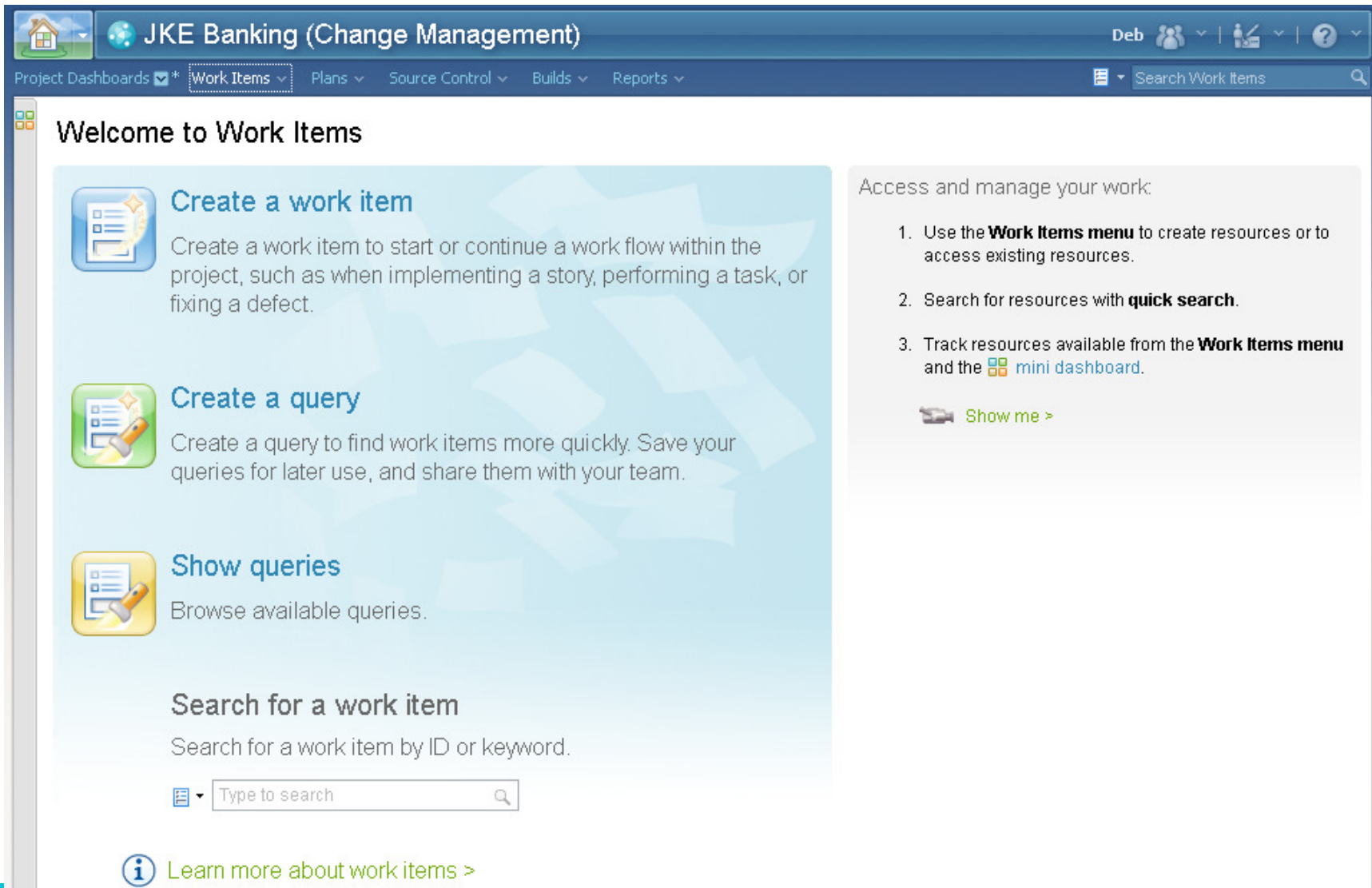
- General:** Includes a 'Welcome to Rational Team Concert' message, 'Current Plans (2)' showing 'Current Iteration: Sprint 2' with 'BRM Sprint 2 (1.0) Plan' and 'Release 1.0 Backlog', 'Project Events' listing membership changes and credit increase requests, and 'Work Items (1)' showing '35: Running out of SWT handles'.
- Release:** Contains 'Work Items (1)' with '34: Meetings are too long', 'Blocked Work Items (1)', 'Recently modified (3)', and 'Recently closed (0)'. It also shows 'Builds' with four successful builds for various production and QA environments.
- Current Sprint:** Features a 'Sprint Burndown' chart. The chart shows 'Remaining Work' (blue line) starting at 227 and decreasing towards 0, 'Planned Work' (grey line) at 0, 'Ideal' (orange line) at 0, and 'Expected Complete' (green line) at 0. A legend at the bottom identifies these lines.

The interface includes navigation menus at the top (Project Dashboards, Work Items, Plans, Source Control, Builds, Reports), a search bar for work items, and an 'Add Widget' button. The IBM logo is visible in the bottom left, and the Jazz logo is in the bottom right.



Jazz and Rational Team Concert capabilities in the web client

► Work-item management



The screenshot displays the IBM Rational Team Concert web client interface for the 'JKE Banking (Change Management)' project. The top navigation bar includes a home icon, the project name, and user information (Deb). Below the navigation bar, there are tabs for 'Project Dashboards', 'Work Items', 'Plans', 'Source Control', 'Builds', and 'Reports'. A search bar for 'Search Work Items' is also present.

The main content area is titled 'Welcome to Work Items' and features three primary actions:

- Create a work item**: Create a work item to start or continue a work flow within the project, such as when implementing a story, performing a task, or fixing a defect.
- Create a query**: Create a query to find work items more quickly. Save your queries for later use, and share them with your team.
- Show queries**: Browse available queries.

Below these actions is a section for **Search for a work item**, which includes the instruction 'Search for a work item by ID or keyword.' and a search input field with the placeholder text 'Type to search'.

On the right side of the interface, there is a section titled 'Access and manage your work:' with a list of instructions:

1. Use the **Work Items menu** to create resources or to access existing resources.
2. Search for resources with **quick search**.
3. Track resources available from the **Work Items menu** and the **mini dashboard**.

A 'Show me >' link is provided below the list.

At the bottom left, there is an information icon and a link: [Learn more about work items >](#)

Jazz and Rational Team Concert capabilities in the web client

► Project planning

The screenshot shows the IBM Jazz web client interface for Project Planning. The top navigation bar includes a home icon, the title "JKE Banking (Change Management)", and user information "Deb". Below the navigation bar, there are tabs for "Project Dashboards", "Work Items", "Plans", "Source Control", "Builds", and "Reports". A search bar labeled "Search Work Items" is also present.

The main content area is titled "Welcome to Plans" and features two primary actions:

- Browse plans**: Browse your project or team area plans, or explore other plans.
- Create a plan**: Create a plan to organize your milestones and track the progress of your project development.

Below these actions is a link: [Learn more about plans](#)

On the right side, there is a section titled "Access and manage your work:" with a list of instructions:

1. Use the **navigation bar drop-down menu** to create resources or to access existing resources.
2. Search for resources with **quick search**.
3. Track resources available from **navigation bar drop-down menu** and **mini dashboard**.

At the bottom of this section is a button: [Show Me >](#)



Jazz and Rational Team Concert capabilities in the web client

► Source control

The screenshot shows the IBM Jazz web client interface for Source Control. The browser title is "JKE Banking (Change Management)". The navigation bar includes "Project Dashboards", "Work Items", "Plans", "Source Control", "Builds", and "Reports". A search bar for "Search Work Items" is visible. The main content area is titled "Welcome to Source Control" and features four main sections:

- Browse streams**: You can view streams and edit resources in them.
- Browse repository workspaces**: You can view your repository workspaces and the resources in them.
- Show locks**: You can view locked resources in streams or repository workspaces.
- Advanced search**: Search for resources in source control.

At the bottom left, there is a link: [Learn more about source control >](#)

On the right side, a box titled "Access and manage your work:" contains a numbered list:

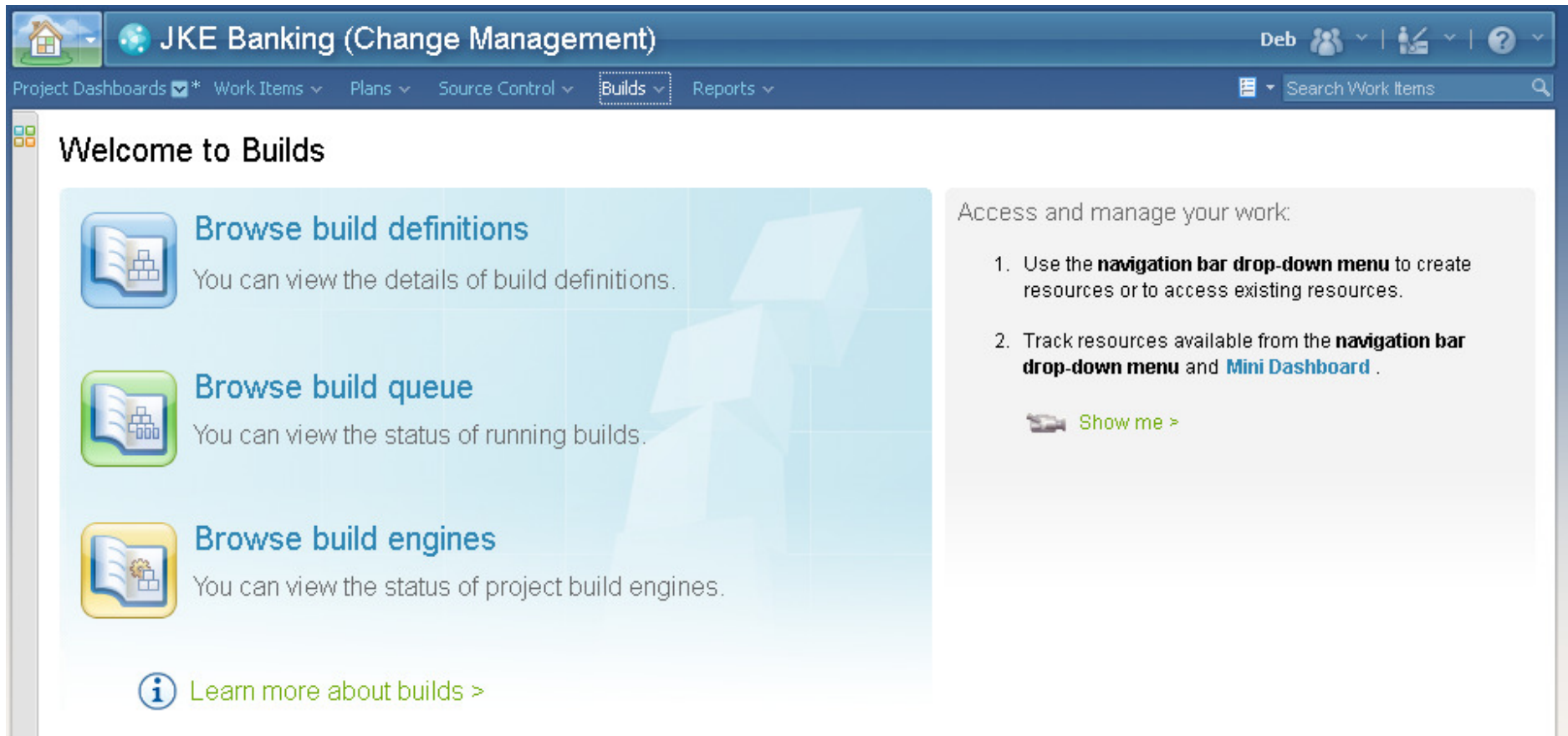
1. Use the **navigation bar drop-down menu** to create resources or to access existing resources.
2. Search for resources with **quick search**.
3. Track resources available from **navigation bar drop-down menu** and **mini dashboard**.

Below the list is a "Show me >" link with a small icon.



Jazz and Rational Team Concert capabilities in the web client

► Build management



The screenshot displays the IBM Rational Team Concert web client interface for the 'JKE Banking (Change Management)' project. The navigation bar includes 'Project Dashboards', 'Work Items', 'Plans', 'Source Control', 'Builds', and 'Reports'. The 'Builds' menu is currently selected. The main content area is titled 'Welcome to Builds' and features three primary actions:

- Browse build definitions**: You can view the details of build definitions.
- Browse build queue**: You can view the status of running builds.
- Browse build engines**: You can view the status of project build engines.

Below these actions is a link: [Learn more about builds >](#)

On the right side, a box titled 'Access and manage your work:' provides instructions:

1. Use the **navigation bar drop-down menu** to create resources or to access existing resources.
2. Track resources available from the **navigation bar drop-down menu** and **Mini Dashboard**.

A [Show me >](#) button is located below the instructions.



Jazz and Rational Team Concert capabilities in the web client

► Reporting

The screenshot shows the 'Reports' section of the IBM Rational Reporting web client. The page title is 'Welcome to Reports'. The navigation bar includes 'Project Dashboards', 'Work Items', 'Plans', 'Source Control', 'Builds', and 'Reports'. The main content area is divided into two columns. The left column contains four task cards: 'View reports', 'Create report from template', 'Organize your reports', and 'Perform additional reporting tasks'. The right column contains a 'Show me' section with a list of three steps for accessing and managing work.

View reports
Browse available reports, to select and view a report.

Create report from template
Create a report from an existing template.

Organize your reports
Organize your reports by creating folders and nesting folders.

Perform additional reporting tasks
Rational Reporting for Development Intelligence is required to perform additional reporting tasks. Contact your Administrator.

[Learn more about reporting >](#)

Access and manage your work:

1. Use the **navigation bar drop-down menu** to create resources or to access existing resources.
2. Search for resources with **quick search**.
3. Track resources available from **navigation bar drop-down menu** and **Mini Dashboard**.

[Show me >](#)



Work items

- Work items capture planned work for a project.
- Different types of work items capture different kinds of work:
 - ▶ **Defect:** Identifies a bug
 - ▶ **Enhancement:** Describes a requested new feature
 - ▶ **Task:** Describes a specific piece of work
 - ▶ **Plan Item:** Provides a high-level description of work that is targeted for a specific iteration
 - ▶ **Story:** Describes part of a use case
- Typical work-item attributes:
 - ▶ Workflows and status
 - ▶ Properties and values
 - ▶ Approvals
 - ▶ Attachments



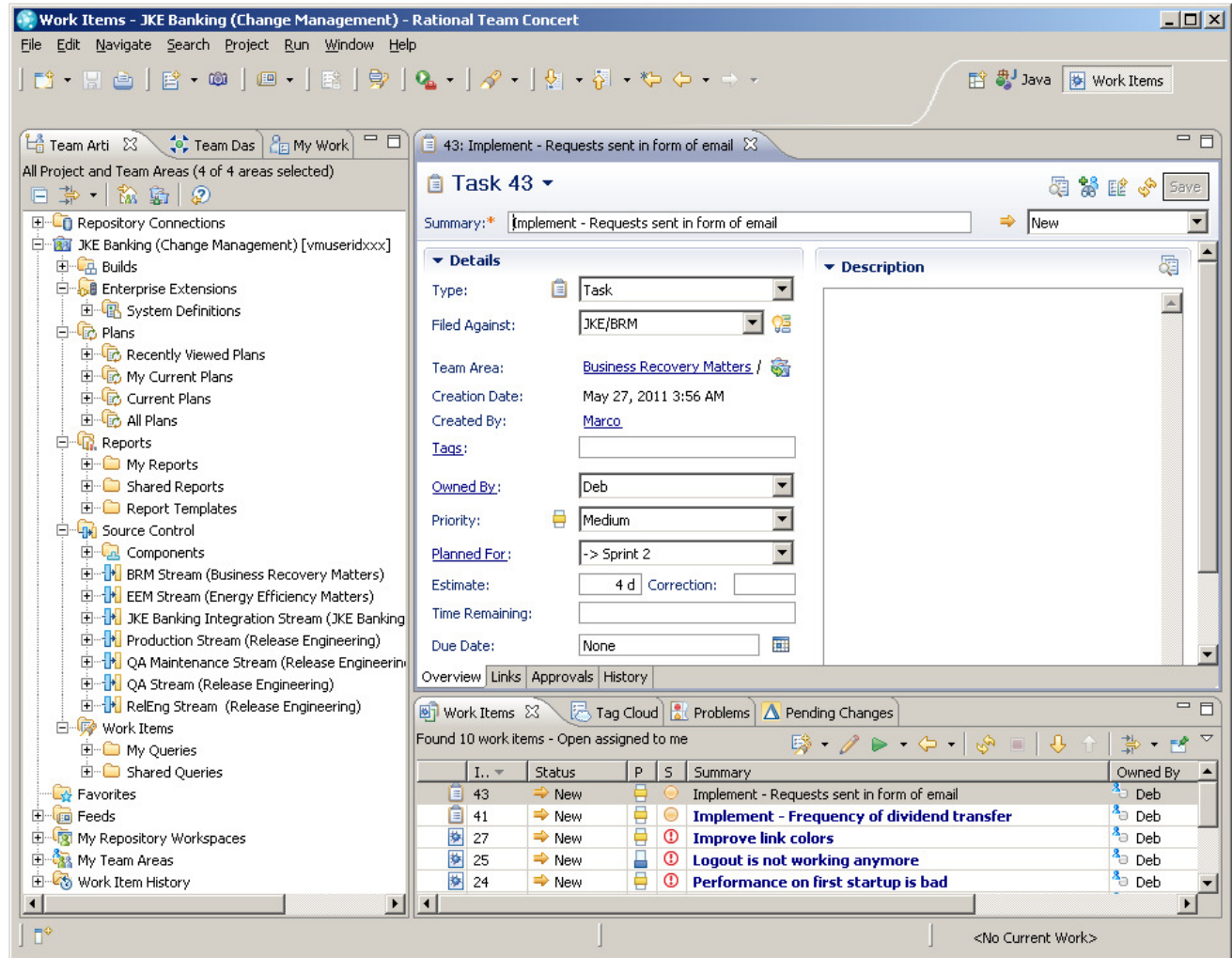
Additional work item capabilities

- Dynamically create relationships to other work items
 - ▶ Establish parent-child relationships
 - ▶ Specify dependencies
 - ▶ Find duplicates
 - ▶ Create references
- Use work items to help you complete these tasks:
 - ▶ Associate work items with change sets.
 - ▶ Allocate work items to plans and iterations.
 - ▶ Estimate the effort that is required to resolve the work item.
 - ▶ Provide metrics regarding project health.



The Work Items

- In the Work Items perspective, you can access and manage your work items. The perspective has these views:
 - Team Artifacts
 - My Work
 - Team Dashboard
 - Work Items



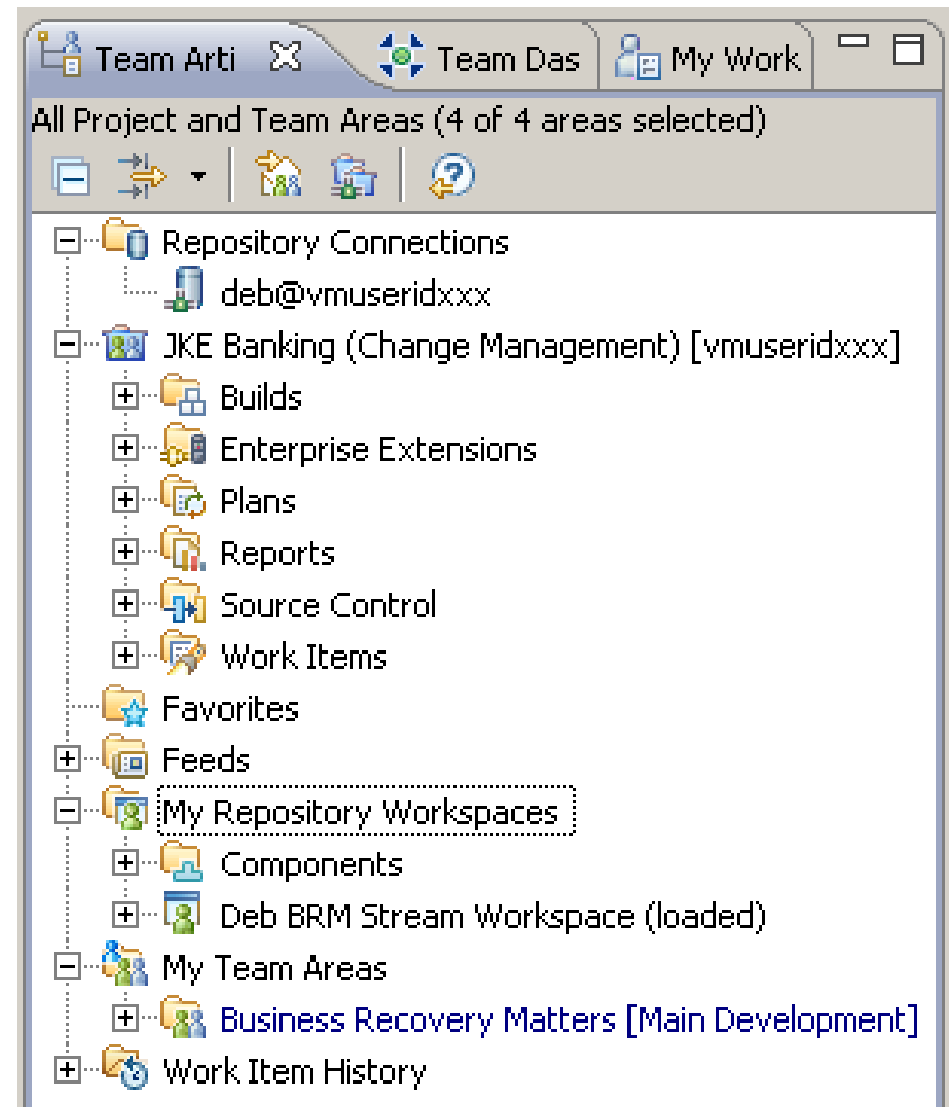
The screenshot shows the Rational Team Concert Work Items perspective. The left-hand tree view displays the project structure, with 'Work Items' selected under the 'JKE Banking (Change Management)' project. The main area is divided into two panes. The top pane shows the details for 'Task 43', including its type, filed against, team area, creation date, and priority. The bottom pane shows a table of work items, with 10 items found and 1 open assigned to the user.

I..	Status	P	S	Summary	Owned By
43	New			Implement - Requests sent in form of email	Deb
41	New			Implement - Frequency of dividend transfer	Deb
27	New			Improve link colors	Deb
25	New			Logout is not working anymore	Deb
24	New			Performance on first startup is bad	Deb



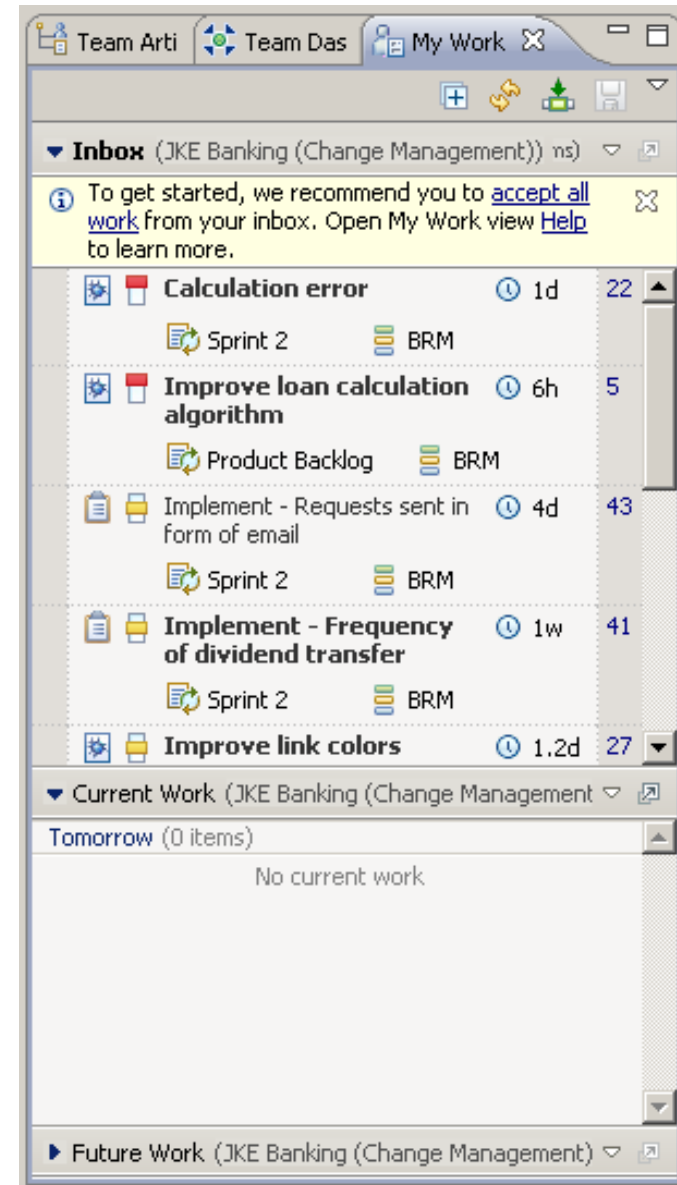
Team Artifacts view

- This view is the central access point for project data.
- The view is organized by these folders:
 - ▶ Your repository connections
 - ▶ Project and folder hierarchies
 - ▶ Your personal folders
 - Feeds
 - Repository workspaces
 - Teams areas that you are assigned to
 - Work Item history

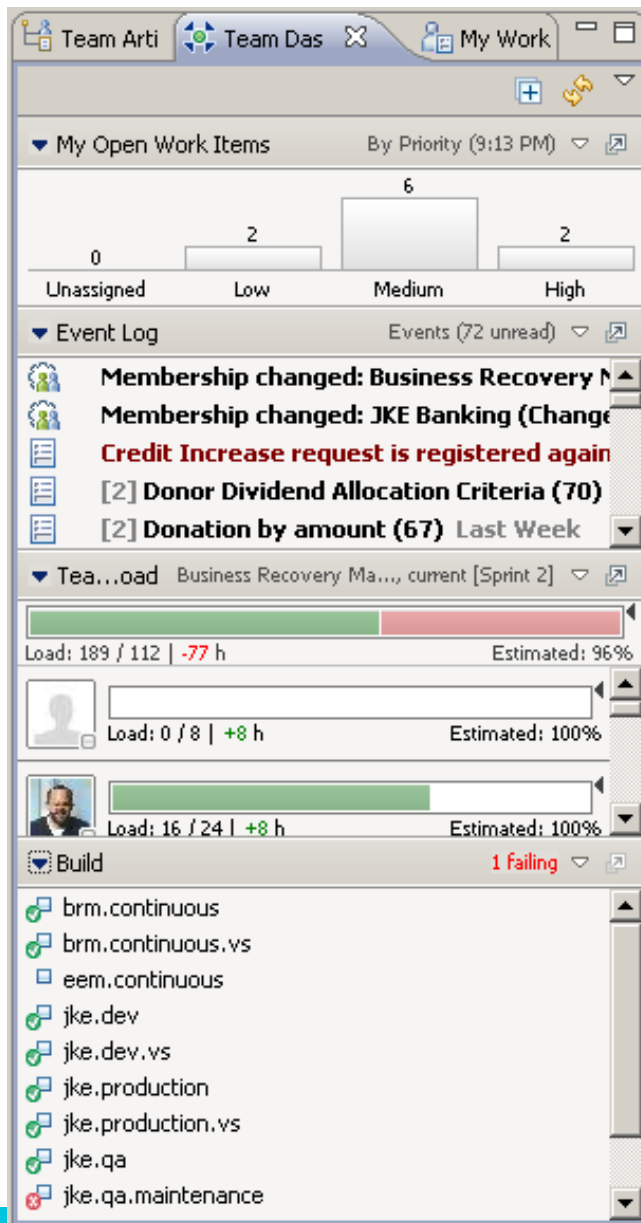


My Work view

- Use this view to manage your work items.
- This view is organized by sections:
 - ▶ **Inbox** lists newly assigned work.
 - ▶ **Current Work** lists work items for the current iteration.
 - ▶ **Future Work** lists work for future iterations.
- Changes that you make in the My Work view can directly impact work items and your teams plans.
- You can customize this view by clicking actions in the **view** menu: Click the **View Menu** icon.



Team Dashboard view



- This view is a central access point for team communications and news.
- The view is continuously updated.
- The view includes these default sections:
 - ▶ **My Open Work Items** lists open work items by Priority
 - ▶ **Event Log** displays information from news feeds.
 - ▶ **Team Load** shows workloads by member.
 - ▶ **Build** lists events for builds.
- You can customize this view by using **view** menu.

Work Items view

- This view displays work- item query results in a table.
- Double-click a work item to open it in the Work Item editor.

The screenshot displays the IBM Rational software interface. On the left, a tree view shows project areas, with 'Work Items' selected. The main pane shows the details for 'Task 43: Implement - Requests sent in form of email'. Below the details is a table of work items.

I..	Status	P	S	Summary	Owned By
43	New			Implement - Requests sent in form of email	Deb
41	New			Implement - Frequency of dividend transfer	Deb
27	New			Improve link colors	Deb
25	New			Logout is not working anymore	Deb
24	New			Performance on first startup is bad	Deb



Ways to create a work item

- You create work items to submit a defect, task, enhancement, or other type of work request for a project.
- There are several ways to create a work item:
 - ▶ Click **Work Items > New > Work Item**.
 - ▶ Duplicate an existing work item.
 - ▶ Create a work item whose **Summary** field contains text that you select from text in the **Summary**, **Description**, or **Discussion** fields in the Work Item editor
 - ▶ Take a screen capture, and attach it to a new work item.
 - ▶ Create a work item in the Build editor.



Creating a work item

1. Right-click the **Work Items** folder, and then click **New > Work Item**.

2. Select the type of work item to create, and then click **Finish**.



Creating a work item (continued)

Defect <09:20:05>

Summary:*

3. Set the properties for the new work item on the Overview page.

Save

Details

Type:

Defect

Filed Against:*

Unassigned

Severity:

Normal

Found In:

Unassigned

Project Area:

JKE Banking...Management)

Tags:

Owned By:

Unassigned

Priority:

Unassigned

Planned For:

Unassigned

Estimate:

Correction:

Time Remaining:

Due Date:

None

Description

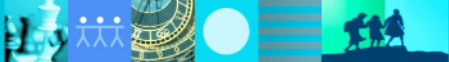
Large empty text area for description.

Quick Information

<No information>

Discussion

Add Comment



Creating a work item (continued)

Defect <09:20:05>



Summary:*

Uninitialized

Attachments

Id	Name	Created	Created
----	------	---------	---------

Add File...

Add Screenshot...

Remove

Rename

Save As...

5. Click **Add File**, select the file that to add, and then click **Open**.

Subscribers

6. Click **Save**.

Remove

Links

Add

Open

Remove

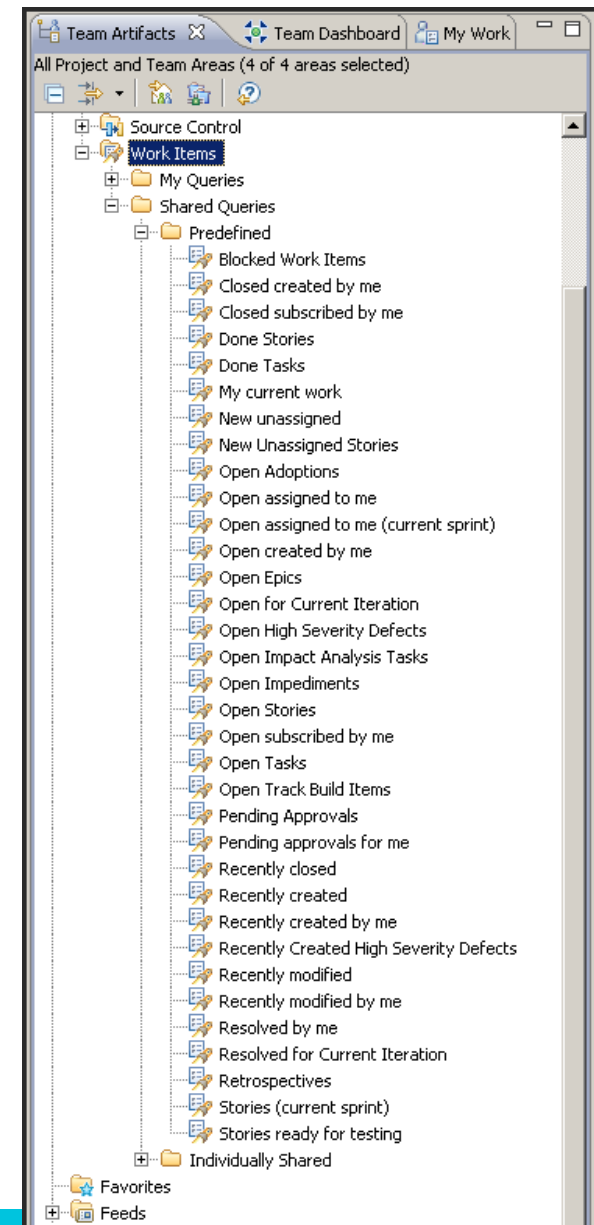
4. To add an attachment to the work item, click **Links**.

Overview **Links** Approvals History



Use queries to find work items

- Queries are the primary mechanism for finding work items.
- Queries are stored in the `Work Items` folder of the `Team Artifacts` view:
 - ▶ The `My Queries` folder stores your private queries.
 - ▶ The `Shared Queries` folder stores predefined and individually shared queries.



Finding work items that are assigned to you

The screenshot shows the IBM Rational software interface. The left sidebar contains a tree view of queries, with 'Open assigned to me' selected. The main view displays a table of 10 work items. The title bar of the 'Work Items' view shows 'Found 10 work items - Open assigned to me'. The table lists work items with their IDs and titles.

ID	Status	Title	Owned By
43			Deb
41			Deb
27			Deb
25			Deb
24			Deb
22			Deb
7	New	Offer more services related to loans	Deb
5	New	Improve loan calculation algorithm	Deb
3	New	Allow a user to create a dashboard of information	Deb
2	New	Allow to edit user details	Deb

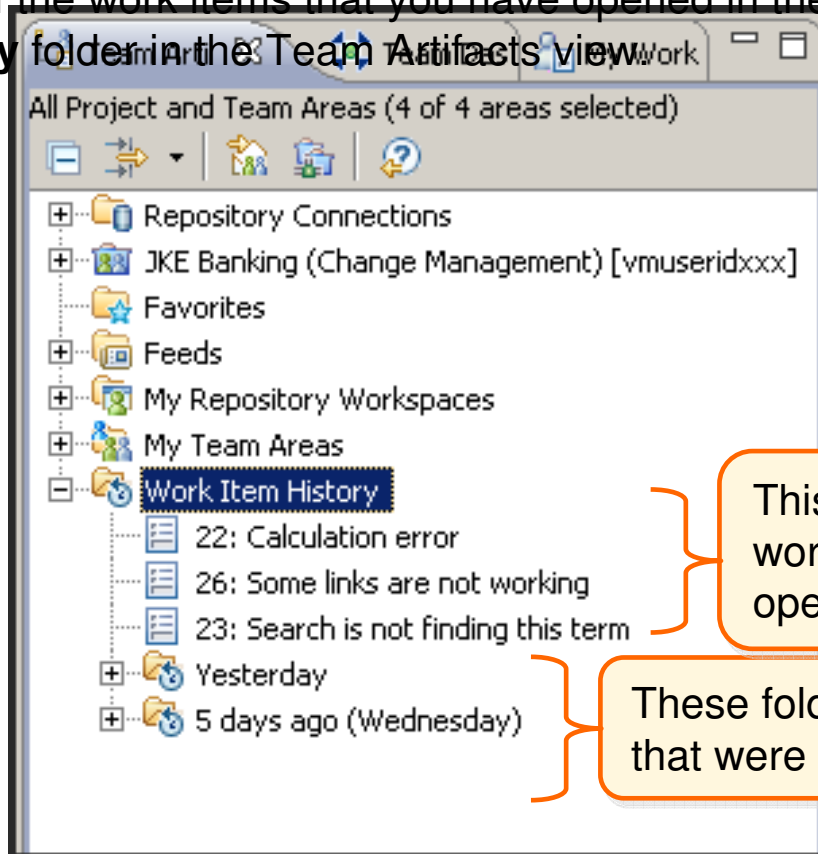
The number of items that are found and the query name are displayed on the title bar.

Double-click the **Open assigned to me** query to find all work items that are assigned to you. The results are displayed in the Work Items view.



Your work item History

- To see a history of the work items that you have opened in the Work Item editor, expand the **Work Item History** folder in the Team Artifacts view.



This list includes the work items that you opened today.

These folders contain work items that were previously opened.



Work items must reflect real-time status

- State the overall status of the work item.
- Describe your work and progress.
- Typical status changes include the following information in these areas:
 - ▶ Overview tab
 - State
 - Estimated time to resolve the work item
 - Discussions
 - ▶ Links tab
 - Add an attachment or subscriber
 - Add relationships to other work items
 - ▶ Approvals tab: Identify who can approve the work item
 - ▶ History tab: View the change history of the work item



Resolving a work item

The screenshot displays a web-based interface for managing a defect. The main window is titled "75: Change Username to Customer Name". The defect summary is "Change Username to Customer Name". The status is currently "In Progress". A dropdown menu is open, showing "In Progress" selected and "Resolve" as an option, which is highlighted with an orange border. The "Details" section on the left includes fields for Type (Defect), Filed Against (JKE/BRM), Severity (Normal), Found In (Sprint 2 Development), Team Area (Business Recovery Matters), Creation Date (Jun 30, 2011 1:06 AM), Created By (Deb), Tags (ui), Owned By (Deb), Priority (Medium), and Planned For (-> Sprint 2). The "Description" section is currently empty. At the bottom, there are tabs for Overview, Links, Approvals, and History.



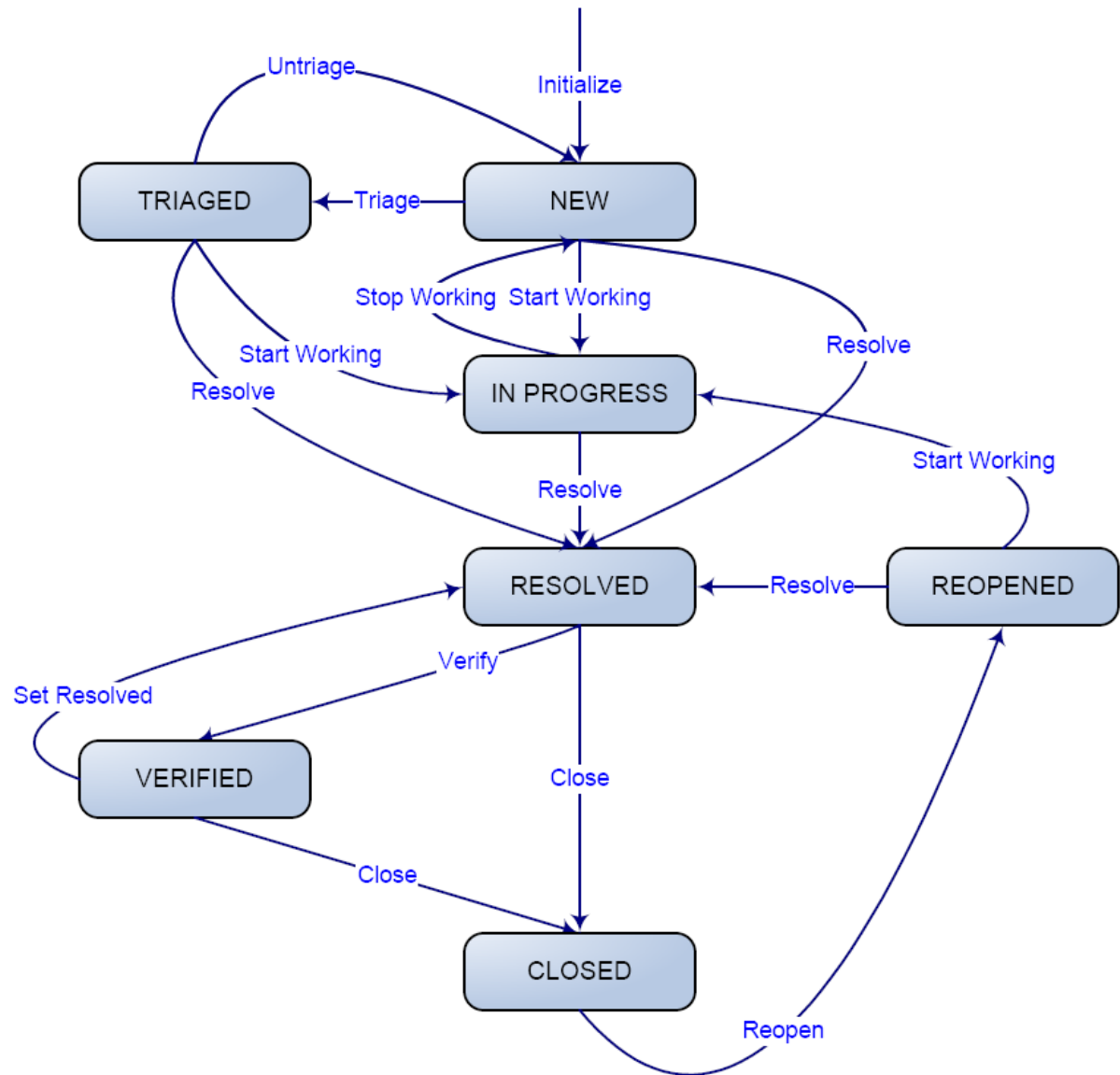
Work item states

Work items evolve from an initial state (New) to a final state (Closed).

Actions move the work items between the states.

This is a typical **workflow** for task-level work items.

Workflows are customizable.



Jazz source control

▪ The Jazz Team Server

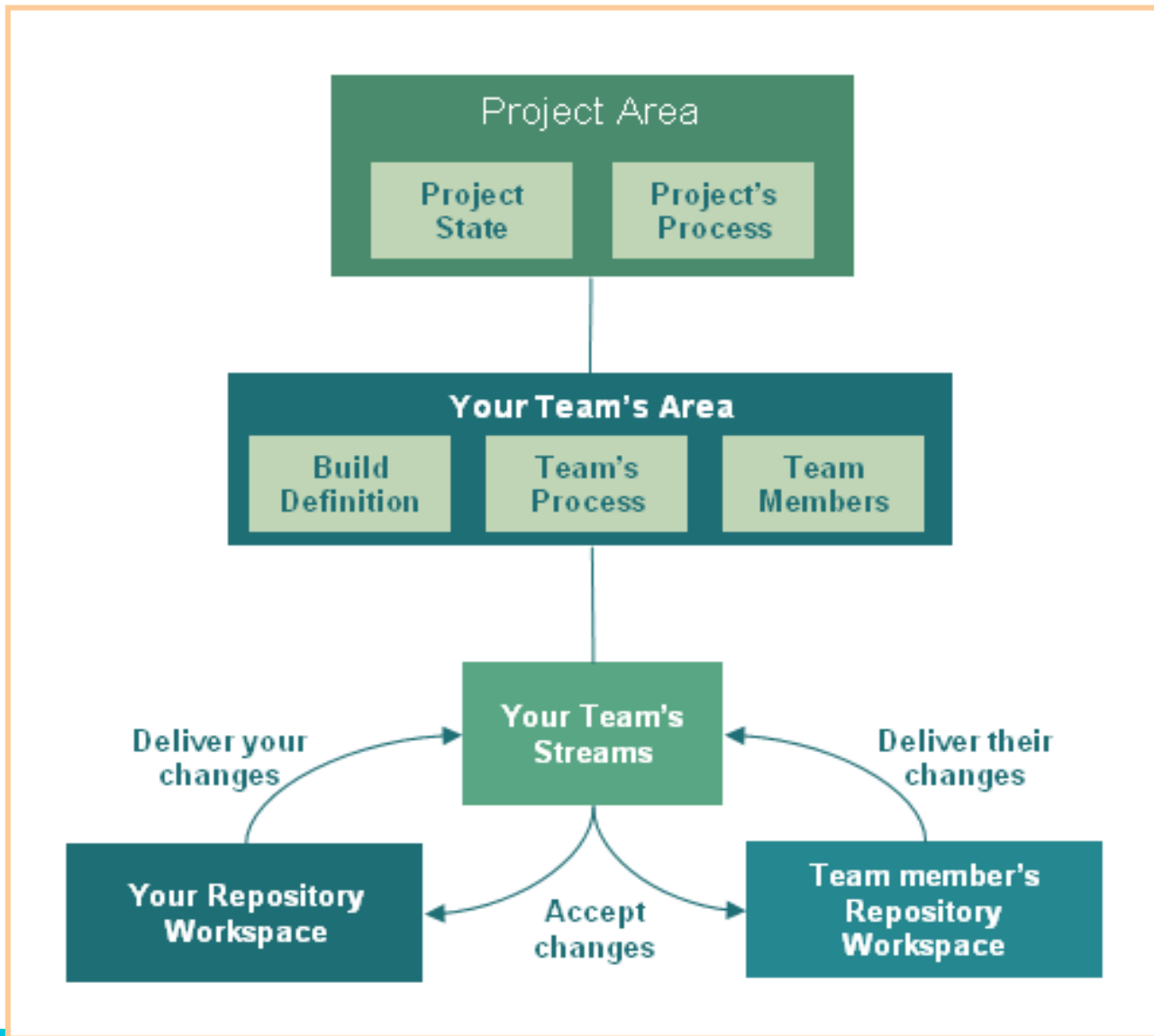
- ▶ Uses a relational database in a central location for all artifacts that are based on Jazz:
 - Stores source code, documents, binary files, images, and so on
 - Stores all Rational Team Concert artifacts
 - Supports geographically distributed teams
- ▶ Provides strong support for parallel development
- ▶ Supports process-centric automation, both agile and traditional processes
- ▶ Provides the infrastructure to link work items, plans, source-control artifacts, and builds

▪ The Jazz source-control repository

- ▶ Is entirely change-set based
 - Requires that all changes to artifacts are related to a change set
 - Ensures atomic changes to sets of files are together
- ▶ Provides secure mechanisms for creating, retrieving, updating, and deleting artifacts
- ▶ Maintains a complete audit trail of all changes, including these events:
 - A record of past states of the item and past field values
 - The user who saved the item
 - The time of the change

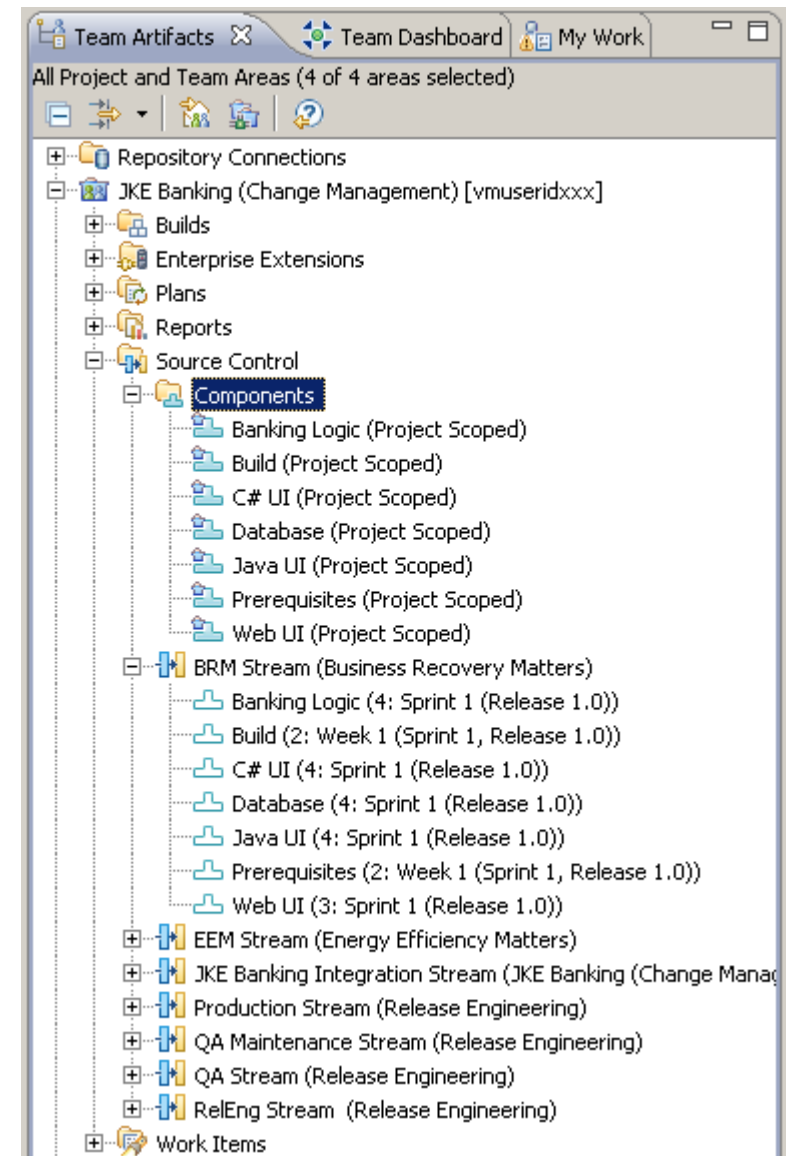


Project area hierarchy



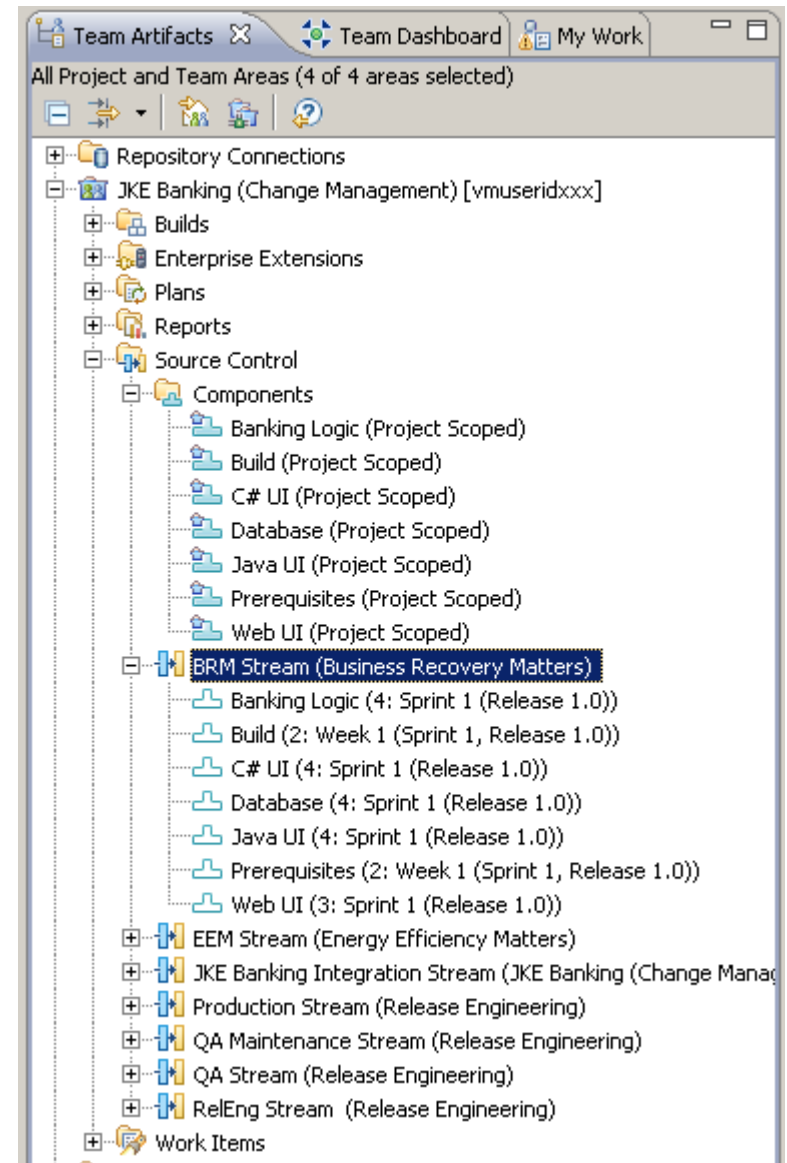
Components

- Fundamental organizational unit for artifacts under source control
- A collection of related artifacts
 - ▶ For example, an Eclipse plug-in or a group of documents
 - ▶ Any group of files and folders that share a common root can be a component
- A **Baseline** represents the “version” of a component



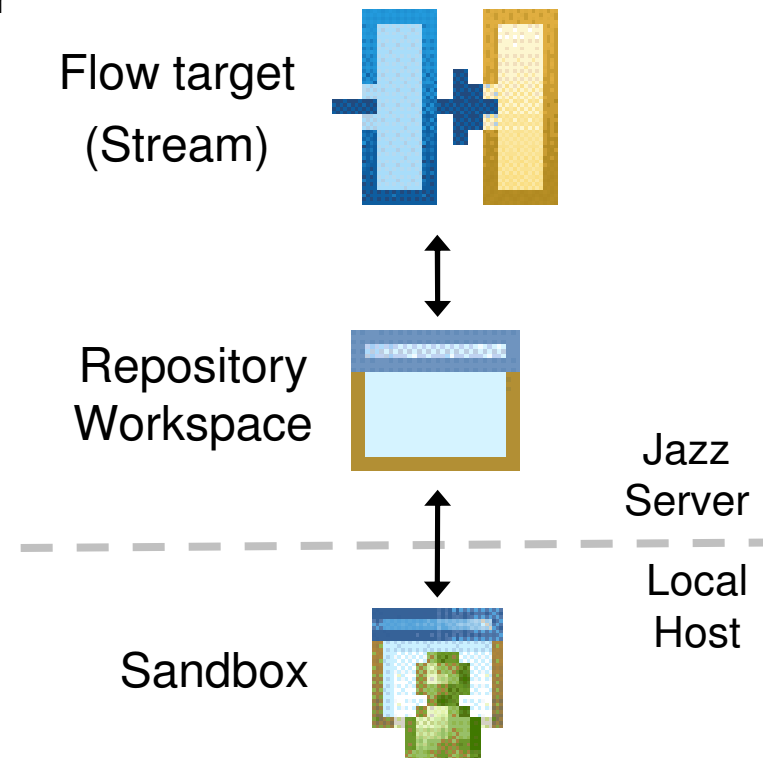
Streams

- A collection of one or more components
- Typical uses for streams:
 - ▶ Provide a team integration area
 - Controlled sharing of changes sets between teams and team members
 - ▶ Recreate important configurations
 - Team integration builds
 - Previous releases
 - Stable versions of third-party or open source software packages
 - ▶ Provide a mechanism for a phased promotion model
 - Development stream or streams
 - Integration stream
 - Production stream



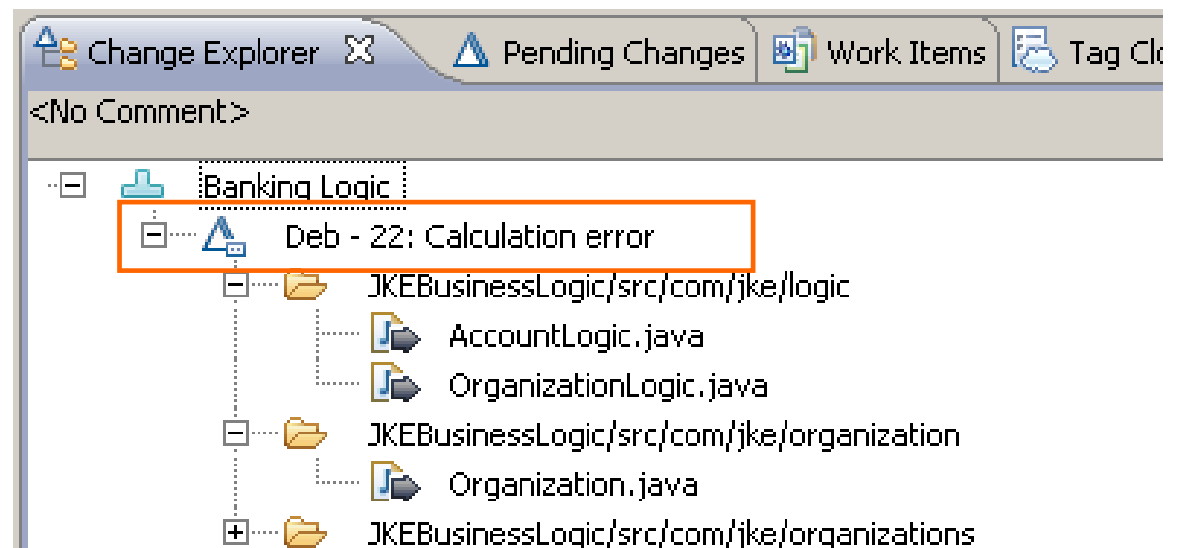
Repository workspaces

- A server-side storage area that a developer uses for current work
 - ▶ **Flow target**
 - Specifies the initial configuration of components
 - Identifies the location where change sets are shared
 - ▶ **Sandbox**
 - provides access to files and folders for desktop integrated development environments (IDEs), such as Eclipse and Visual Studio, and other development tools



Change sets

- A **change set** is a repository object that collects a related group of changes in a component:
 - ▶ The system tracks the changes that you make in the context of your current work as a change set.
 - ▶ Change sets are applied to a flow target in a single atomic operation



indicates change set



Pending changes

- Use the **Pending Changes view** to manage the flow of change sets into and out of your repository workspace

- ▶ **Unresolved** ◀

Files in this section have been modified in the local sandbox, but have not been checked in to the repository workspace.

- ▶ **Outgoing** ◀

Checked-in files are organized into change sets.

Outgoing change sets have not been delivered from the repository workspace to the flow target.

- ▶ **Incoming** ◀

Incoming change sets represent changes that have been applied to the flow target, but have not been accepted into this repository workspace.

Work Items Tag Cloud Problems Team Advisor Pending C

1 unresolved local, 1 incoming change set, 1 outgoing change set

Rebecca's BRM Stream Workspace BRM Stream

Banking Logic

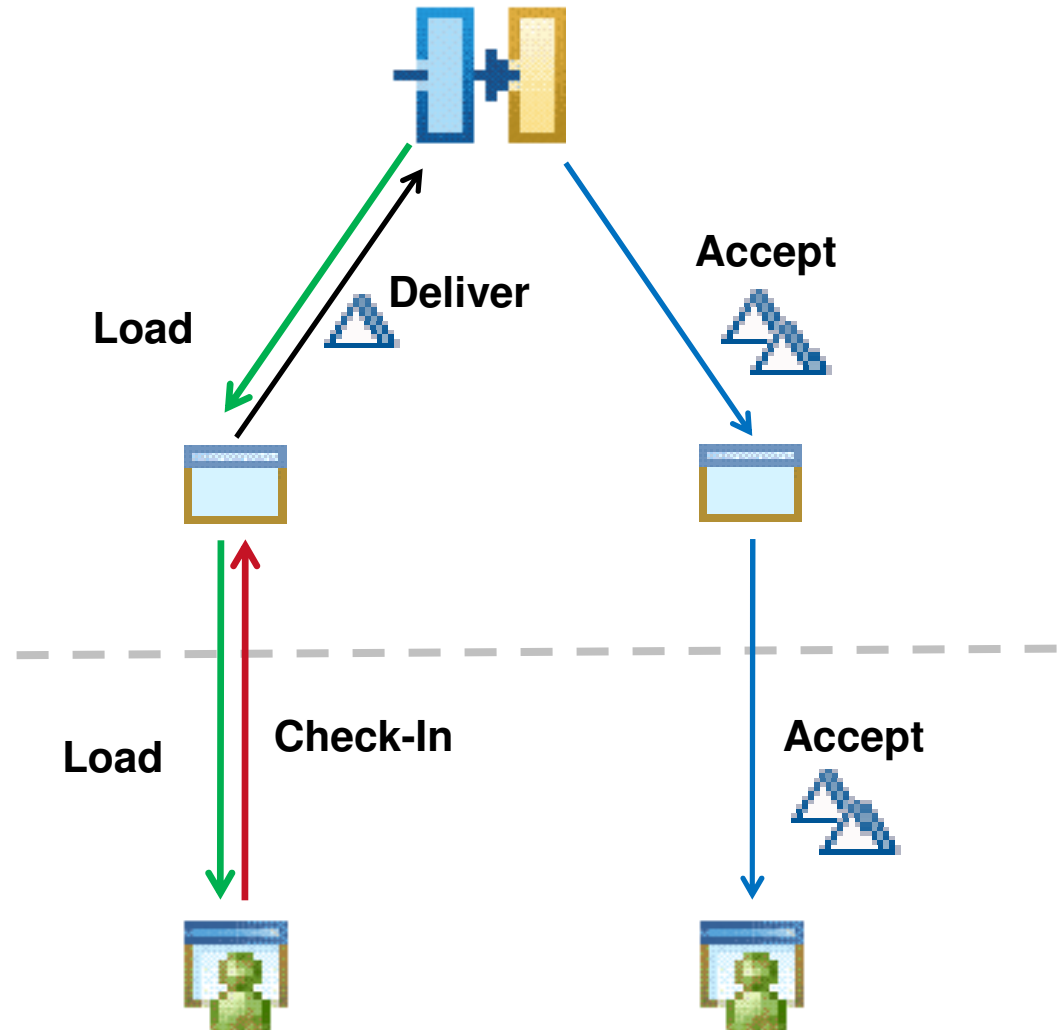
- Unresolved
 - JKEBusinessLogic/src/com/jke/organizations
 - SalvationArmy.java
- Outgoing
 - 21: Some messages are not externalized
 - JKEBusinessLogic/src/com/jke/organizations
 - AmericanCancerSociety.java
 - 21: Some messages are not externalized
- Incoming
 - 22: Calculation error
 - JKEBusinessLogic/src/com/jke/logic
 - AccountLogic.java
 - OrganizationLogic.java
 - JKEBusinessLogic/src/com/jke/organization
 - Organization.java
 - JKEBusinessLogic/src/com/jke/organizations
 - Care.java
 - 22: Calculation error

Build



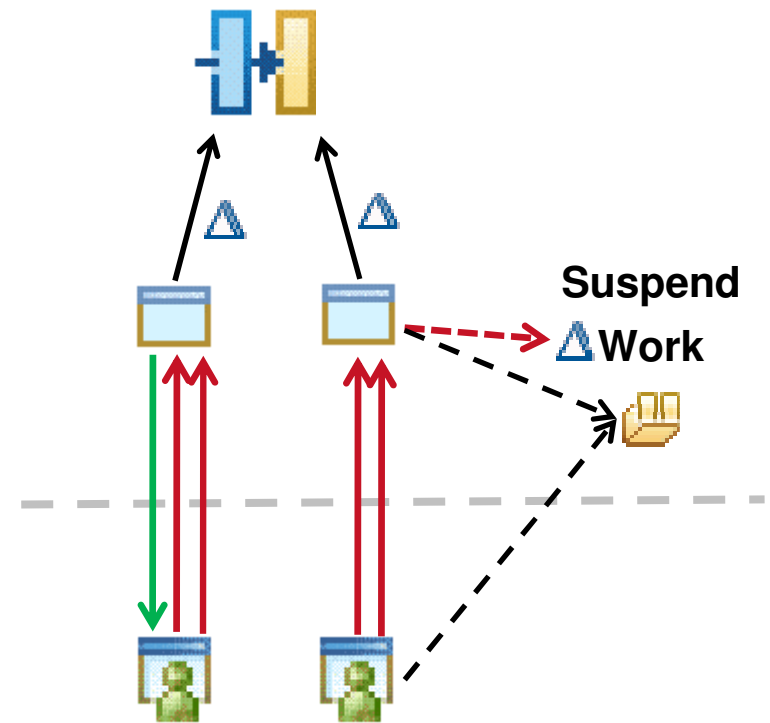
Using workspaces in a team environment

- **Load**
 - ▶ Copies a specified configuration of files from the stream to your repository workspace and local sandbox
- **Check-In**
 - ▶ Copies changed files from the local sandbox to the repository workspace
- **Deliver**
 - ▶ Copies the change set or sets from the repository workspace to the flow target
- **Accept**
 - ▶ Copies a change set or sets in the flow target into the repository workspace and the local sandbox
- **Deliver and Accept operations will identify potential conflicts if the same file is modified in more than one change set.**



Creating and managing change sets

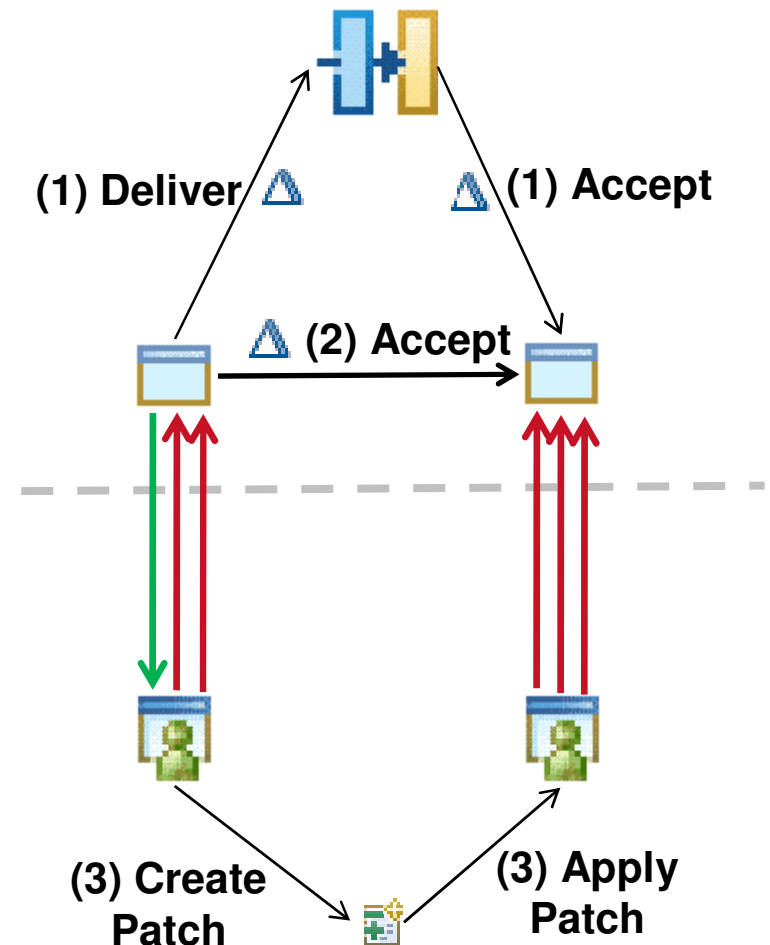
- **Best practices:**
 - ▶ Set your “Current Work Item.”
 - ▶ Work on one task at a time.
- If you have to multitask:
 - ▶ Use multiple repository workspaces if you need to work on multiple tasks at the same time.
 - ▶ **Suspend** work on a change set, which puts that work off to the side temporarily. You can resume the work later.
- Worst case scenario:
 - ▶ Adjust change set contents in your repository workspace before you deliver.



Sharing work between workspaces

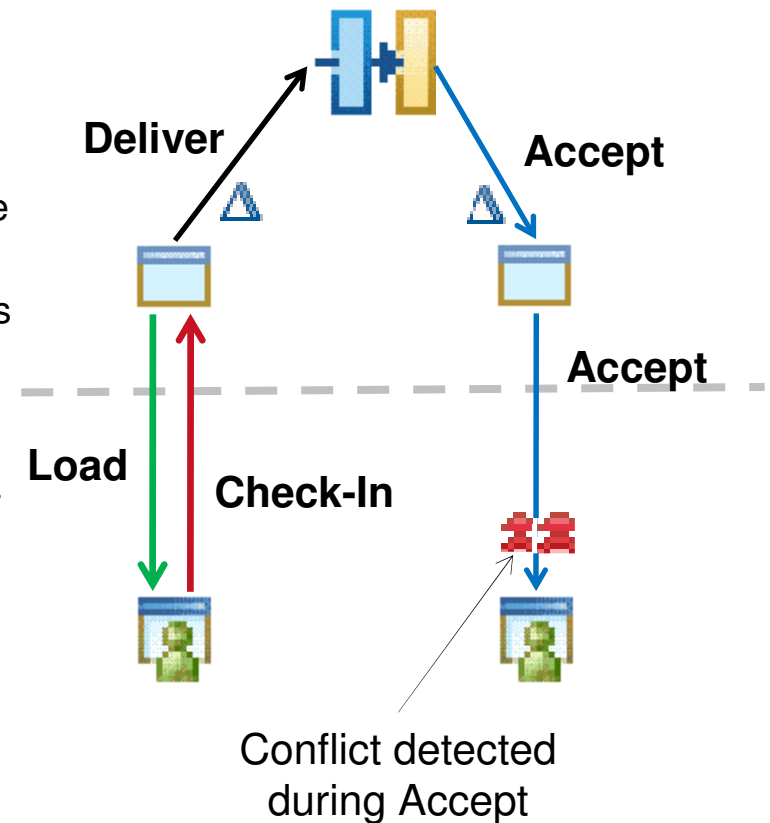
- Multiple developers who collaborate on a task must be able to share their work:

- Use the normal **deliver-accept** model to propagate a change set from one workspace to the other through a parent stream or repository workspace.
- Change your flow target to point to the other workspace and **accept** changes directly from that workspace.
- Create a patch file, which other users can apply to their workspaces.



Finding conflicts

- Conflicts are detected at the file level, so any changes to the same file by two different change sets can potentially result in a conflict
 - Conflicts can be detected during either the **delivery** or the **accept** process
 - Rational Team Concert will offer to “auto-resolve” conflicts without user intervention
 - If the changes cannot be resolved automatically (for instance, the same line of code was modified in both workspaces), Rational Team Concert will prompt the user to perform a manual merge

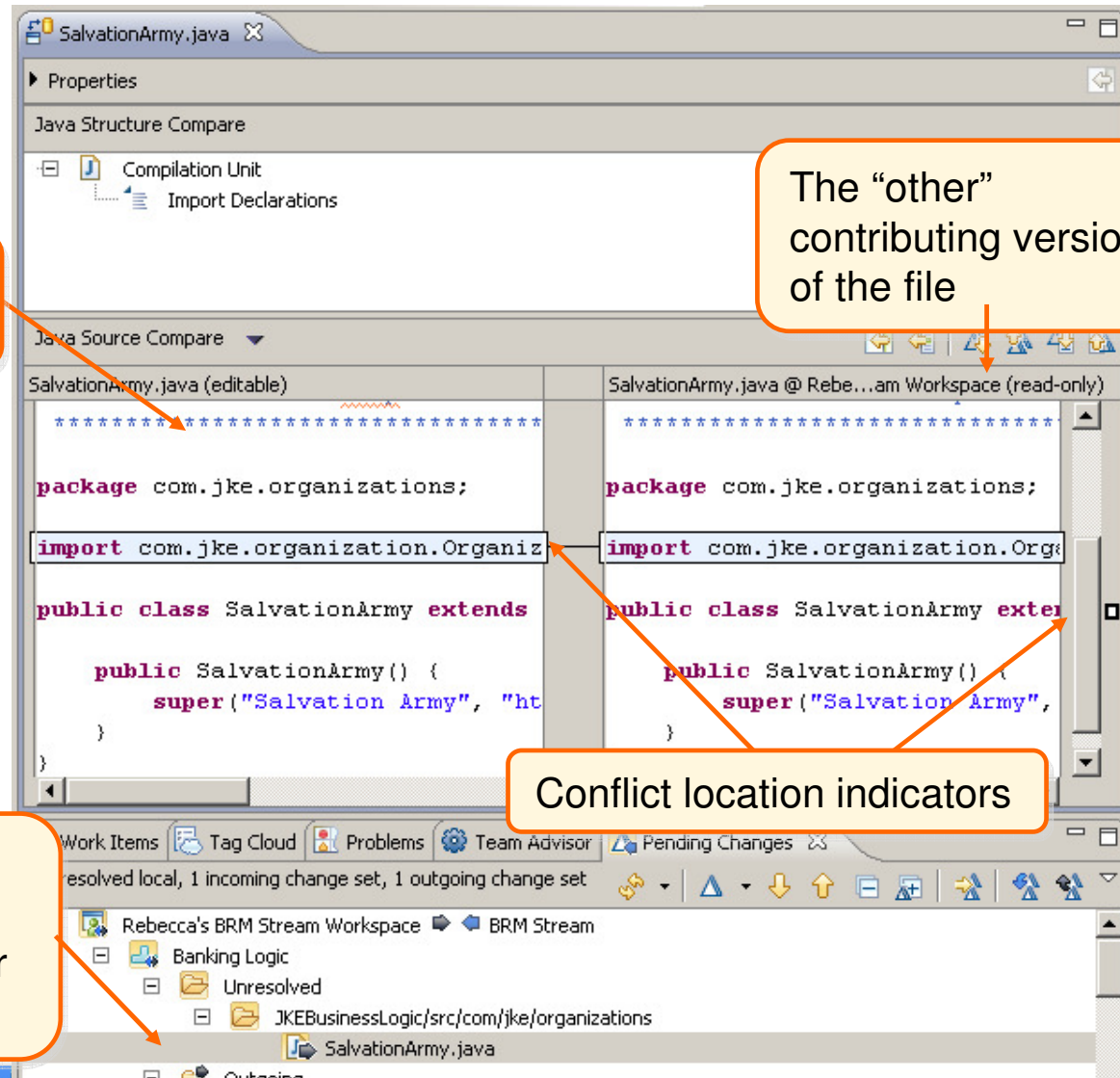


Resolving conflicts

- Use the **Compare editor** to manually resolve conflicts.

Your editable version of the file

The "other" contributing version of the file



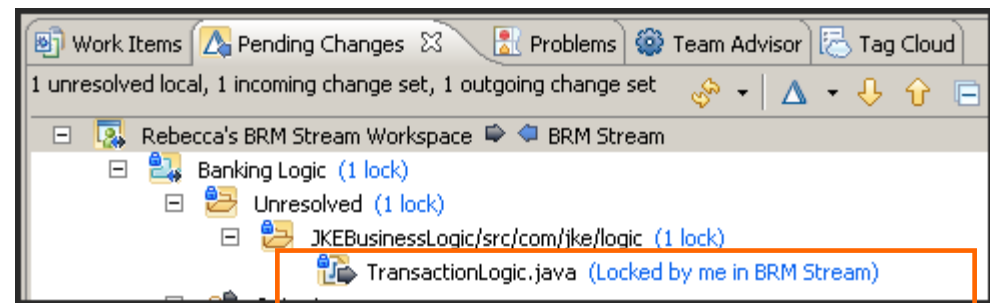
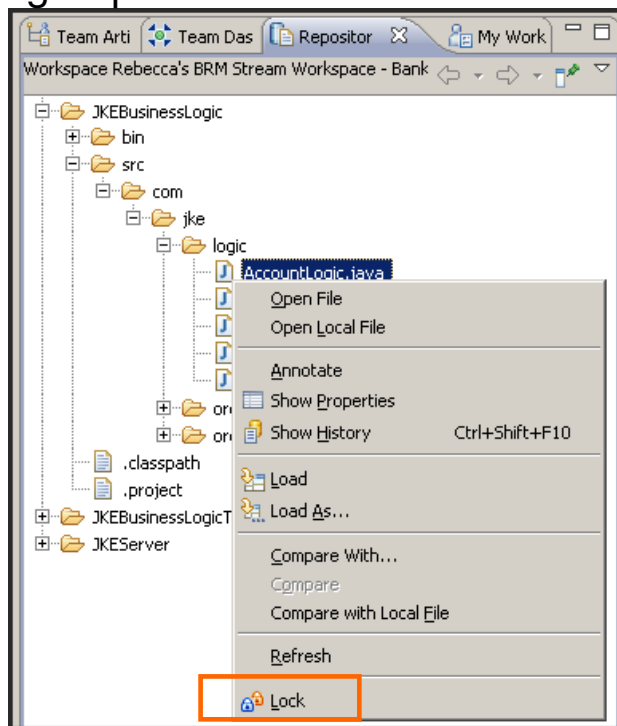
Conflict location indicators

The conflict is called out in the Change Explorer view



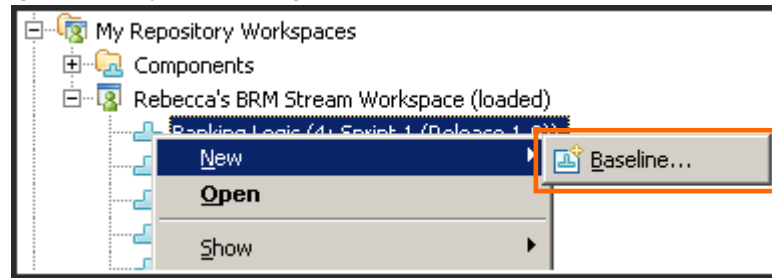
Avoiding conflicts

- File locking
 - ▶ You can lock a file on a "per-stream" basis.
 - ▶ While you have a file locked, nobody else can deliver changes to that file in that stream.
 - ▶ A file can be unlocked by the user who owns the lock or by a member of the JazzAdmins group

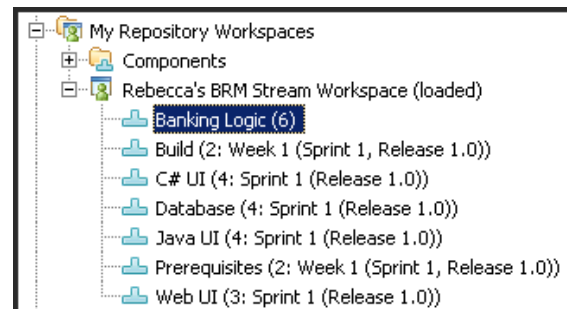


Baselines

- A **baseline** is a version of a component
 - ▶ Components configurations can be saved at any point in time through creation of a **baseline**.
 - ▶ Baselines are created in a repository workspace and delivered to a stream as a special type of change set.



- ▶ The Team Artifacts view indicates which component baselines are in a repository workspace or stream.



Restoring configurations

- Replace the current component configuration in a stream or repository workspace with a different baseline.

1. Open the stream object under the project folder to edit the stream properties.

2. Click **Replace With** to change baselines.

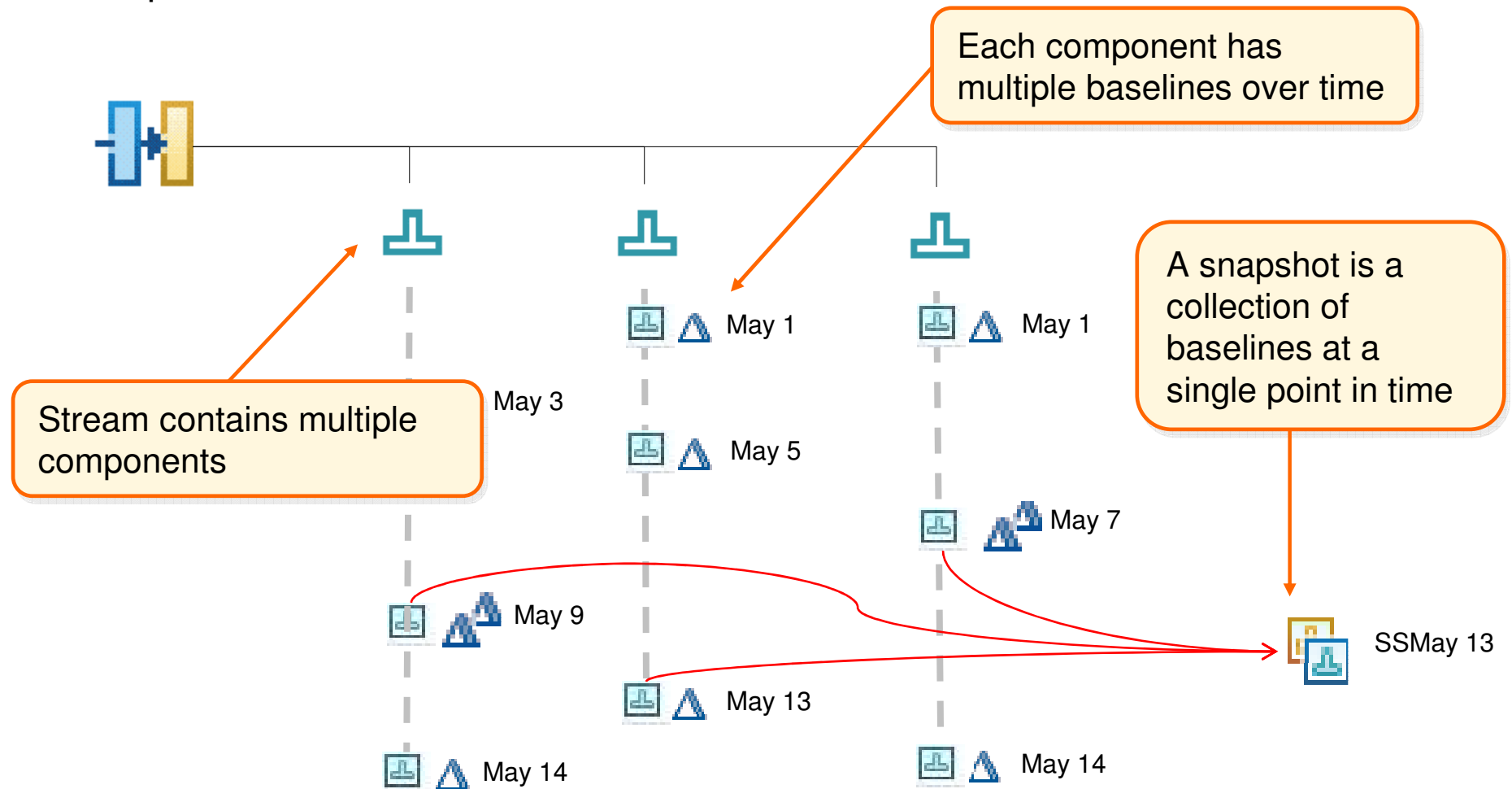
3. Choose which baselines to restore.

Baseline ID	Baseline Name	Date
6	(Jun 15, 2011 12:12 AM)	
4	Sprint 1 (Release 1.0)	(Jun 6, 2011 3:58 AM)
3	Week 2 (Sprint 1, Release 1.0)	(Jun 6, 2011 3:58 AM)
2	Week 1 (Sprint 1, Release 1.0)	(Jun 6, 2011 3:58 AM)
1	Initial Baseline	(Jun 6, 2011 3:56 AM)



What is a snapshot?

- A **snapshot** is a collection of baselines across all of the components in a repository workspace or stream



Project Management - Dashboards

Sprint Backlog

Team Area: Havannah Team | Iteration: Sprint 1 (1.0) (6/1/09 - 6/12/09) | 0 Closed | 33 Open

Engine can choose appropriately between placing a piece that leads to making a bridge and one that leads to making a ring. Unassigned -- 39

Delaney
 Closed items: 0 | Open items: 8
 Load: 50 / 80 | +30 h Estimated: 100%

- As a player I'd like to be able to use the system to play against another human on my computer High 0/32 h 20
- As a player I can play against a weak engine that recognizes rings High 0/84 h 9

Prasad
 Closed items: 0 | Open items: 3
 Load: 36 / 80 | +44 h Estimated: 100%

- As a player I can play against a weak engine that recognizes rings High 0/84 h 9

Rose
 Closed items: 0 | Open items: 2
 Load: 18 / 54 | +36 h Estimated: 100%

Sasha
 Closed items: 0 | Open items: 14
 Load: 60 / 80 | +20 h Estimated: 100%

Unassigned
 Closed items: 0 | Open items: 5

- As a player I can play against a weak engine that recognizes bridges High 0/82 h 27
- As a player I'd like to be able to use the system to play against another human on my computer High 0/32 h 20
- As a player I can play against a weak engine that recognizes rings High 0/84 h 9
- Automate test cases for blocking a human player making a ring Unassigned -- 19
- Identify test cases for blocking a human player making a ring Unassigned -- 18



Sprint Backlog [Sprint 1 (1.0)]

Team Area: Havannah Team | Iteration: Sprint 1 (1.0) (7/27/09 - 8/7/09) | 11 Closed | 23 Open

Progress: 84/198 | +17.75 h | Estimated: 100%

Prasad
Closed items: 5 | Open items: 7

Progress: 10/64 | +10.25 h | Estimated: 100%

To Do	In Progress	Done
As a player I can play against a weak engine that recognizes bridges 11	Automate tests for choosing between making a bridge or a ring 57	Identify tests for forming a bridge around obstacles 47
	Automate tests for blocking a player from making a bridge 54	Automate tests for giving up on a bridge 64
	Automate tests for forming a bridge around obstacles 48	Identify tests for giving up on a bridge 50
As a player I'd like to be able to use the system to play against another human on my computer 10		Identify and automate tests for simple bridge design 45
As a player I can play against a weak engine that recognizes rings 9	Automate test cases for trying to make a blocked ring 34	Automate Tests 43
	Write automated tests for unblocked rings 31	Identify test cases for trying to make a blocked ring 33
		Write automated tests for state management classes 29

Rose
Closed items: 2 | Open items: 1

Progress: 10/14 | -0.5 h | Estimated: 100%

To Do	In Progress	Done
As a player I'd like to be able to use the system to play against another human on my computer 10		Design Tests 42
		Very simple board and graphics 38
As a player I can play against a weak engine that recognizes rings 9	Identify test cases for blocking a human player making a ring 36	

View As

- Backlog
- Developer's Dashboard
- Manned Time
- Team Folders
- Work Breakdown

Edit | Copy

Actions

Re-sort

Exclude

- Assigned Items
- Empty Groups
- Estimated Items
- Execution Items
- Resolved Items
- Unchanged Items

Related Work Items

Show Backlog

Unplanned Closed Items

Next Plans

