

# Innovate2010

The Rational Software Conference

Let's **build** a smarter planet.

Improve your Requirements Definition and Management using Rational Requirements Composer

**Vishy Ramaswamy**

*Chief Architect, IBM*

[vramaswa@ca.ibm.com](mailto:vramaswa@ca.ibm.com)

The premier software and product delivery event.



## Agenda

- Requirements are Everywhere
- Customer Feedback
- Rational's Vision for Requirements Definition and Management (RDM)
- Rational Requirements Composer (RRC)
- Traceability
- Just Enough Management Capabilities
- Rational RequisitePro Integration
- Application Lifecycle Integrations



## The usual disclaimer

- Plans are based on best information available and may change in the future
- Don't make purchase decisions based on features not currently in generally available releases.

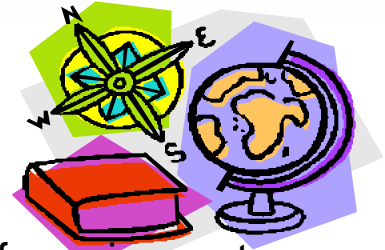


**Information**

© 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.

## Requirements are Everywhere

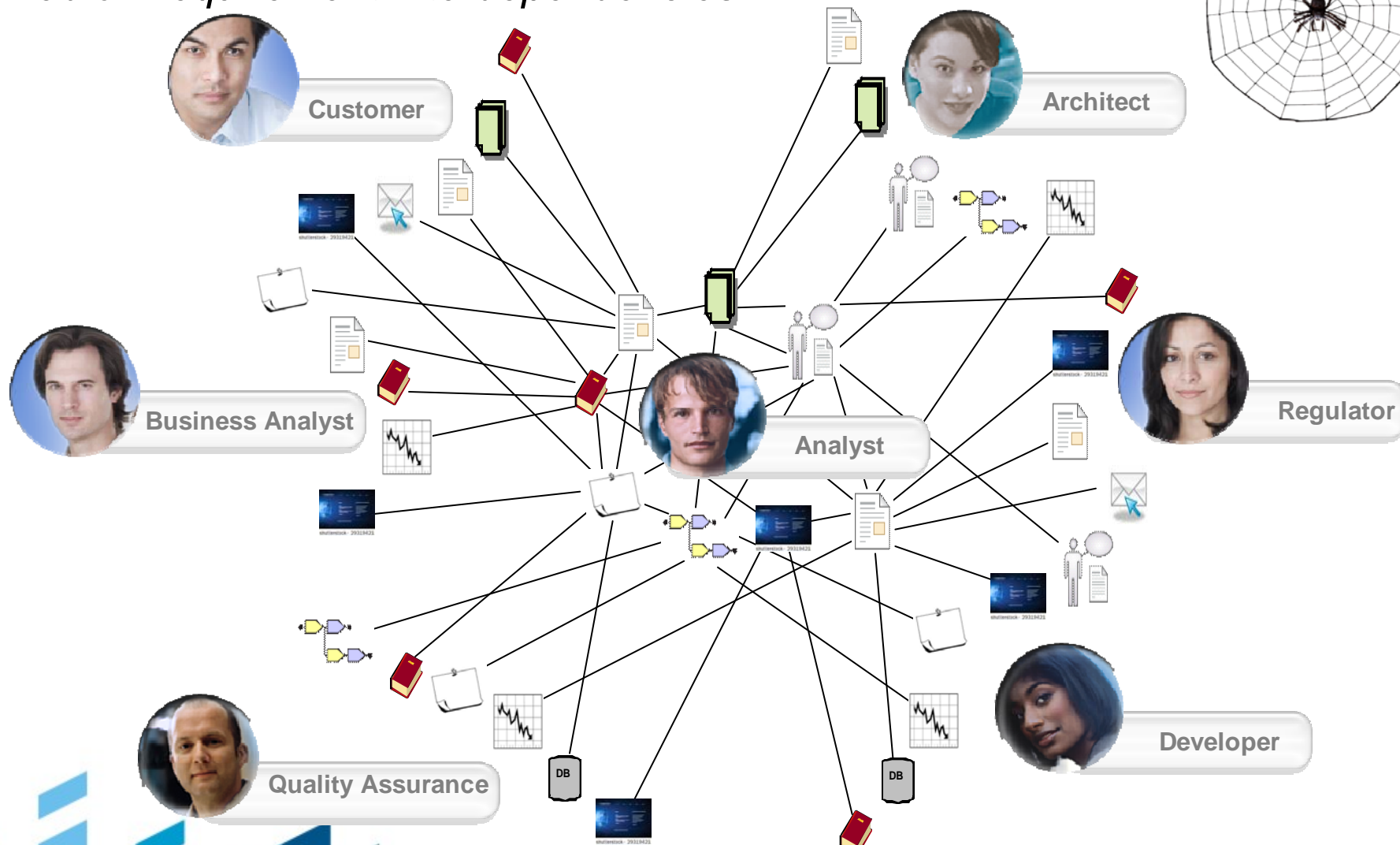
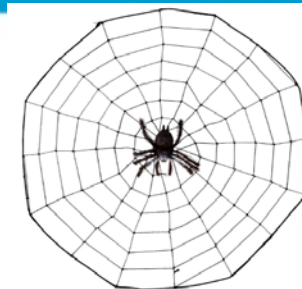
### *Changing Landscape of Requirements*



- **More integrated systems** rather than stand alone systems
  - More interdependence of components hence more interdependence of requirements
- **More stakeholders involved** in projects
  - More requirements sources
    - Business analysts, users, customers, marketing, regulators, architects, domain experts, legacy system experts, development, testing etc.
  - Stakeholders need to validate requirements in a form they can understand
    - Need to leverage diagrams, process flows, screen sketches, use cases etc. as well as textual requirements
    - Need to be able to incorporate these different forms into the requirements process
- **Improved technology** and tools enable us to build more complex systems
  - More complex requirements
  - But users still want solutions to be easy to use!!!

## Requirements are Everywhere

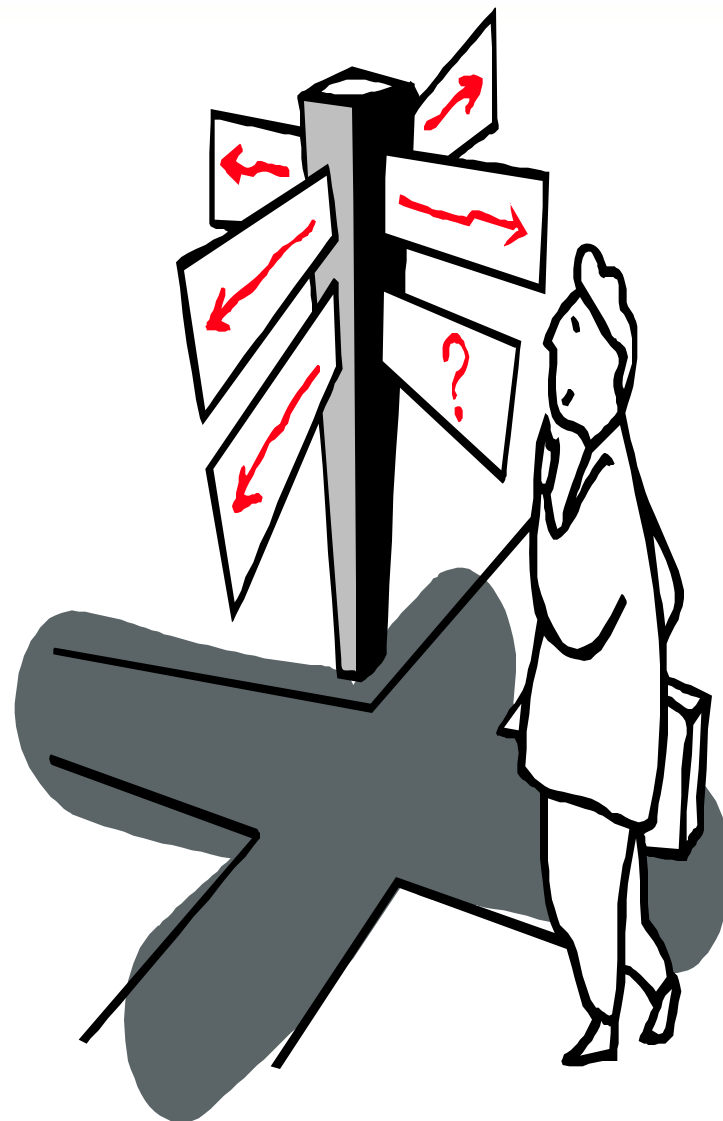
### Web of Requirement Interdependencies



## Requirements are Everywhere

### Motivations for doing more

- Why are requirements important to our customers
  - Gaps in shared understanding between business and IT
  - Uncertainty, driven by perception, knowledge, and communication gaps during the requirements process
- Effects that drive project failure and cost
  - Poorly defining the problem
  - Poorly defining a solution
  - Poorly managing change
- There are opportunities to better address business-driven development beyond what today's Requirements Management (RM) tools can do
  - More focus on discovering requirements in the first place
  - Do more to help validate requirements
  - Support specification techniques beyond text-based



## Requirements are Everywhere

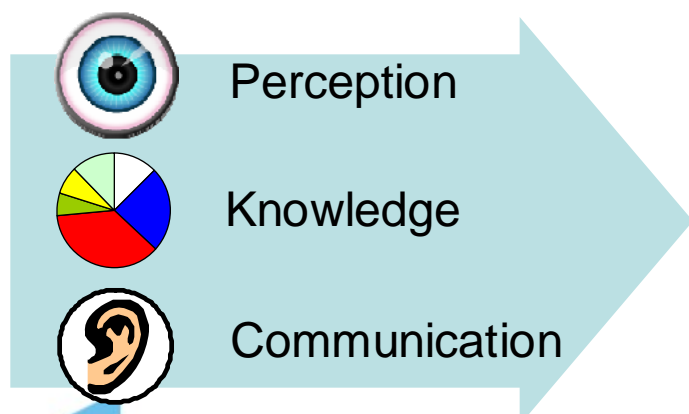
*Also Other Challenges for Project Teams*

- **Shorter market cycle times**
  - Necessitating more project agility and more frequent requirement changes
- Drive to **reduce cost and schedule**
  - Focus on productivity and value add activities
- Teams and stakeholders are **more geographically distributed**
  - Need better communication and collaboration



## Customer Feedback And Market Research

- Requirements are ubiquitous and continue to be labor-intensive. We need to rethink our approach to requirements.
- Business customers are conflicted about requirements. They want more confidence and correctness in requirements, but also more efficiency in the definition process.
- Improving the requirements definition process offers more ROI than incrementally better requirements management for many organizations
- 80-plus percent of software development failures result directly from poor requirements gathering, management, and analysis.
- Need to improve business/IT communication



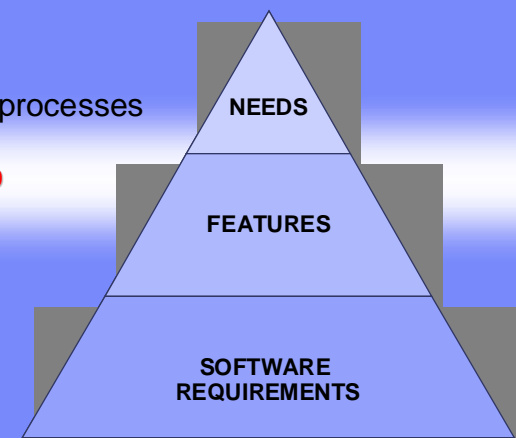
### Problem Space:

- Are we solving the right problem?
- Where we know what we know
- Frequently discontinuous and ad-hoc processes

### Solution Space

- Are we solving the problem right?
- Results linked to who / what we know
- Structure accelerates issues

**GAP**





## IBM Rational Requirements Definition And Management

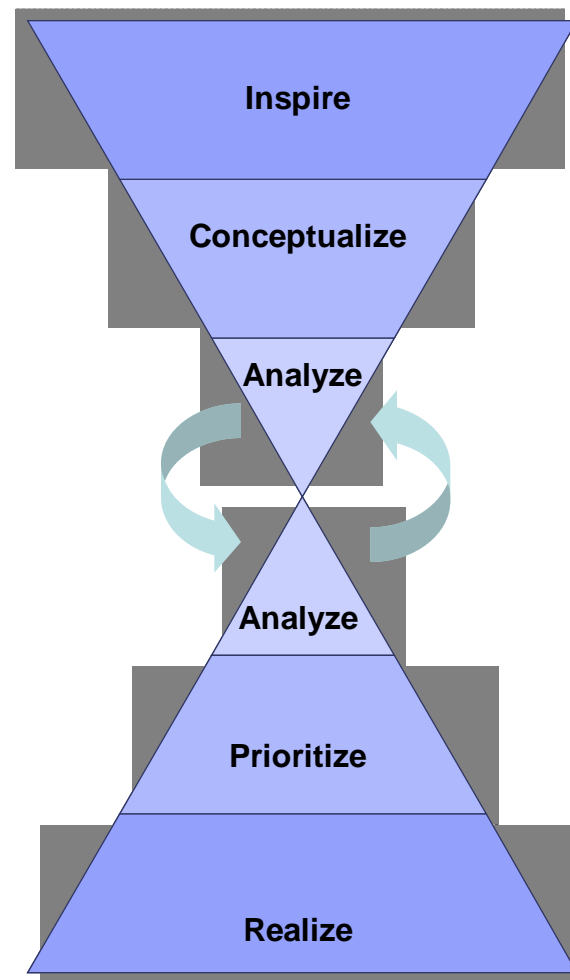
### Our vision for Requirements Definition and Management (RDM)

Foster focused, natural, real-time, contextual **collaboration** using various techniques and artifact types

#### Enable:

- improved **requirements definition, validation and management of requirements** change through the software development lifecycle
- more and clearer **communication among business stakeholders & IT delivery teams** wherever they are located
- **less project rework, faster project execution and lower-cost delivery**

### Requirements Definition



### Requirements Management

## How well do you Write Requirements

*Real World Advertisement Examples*

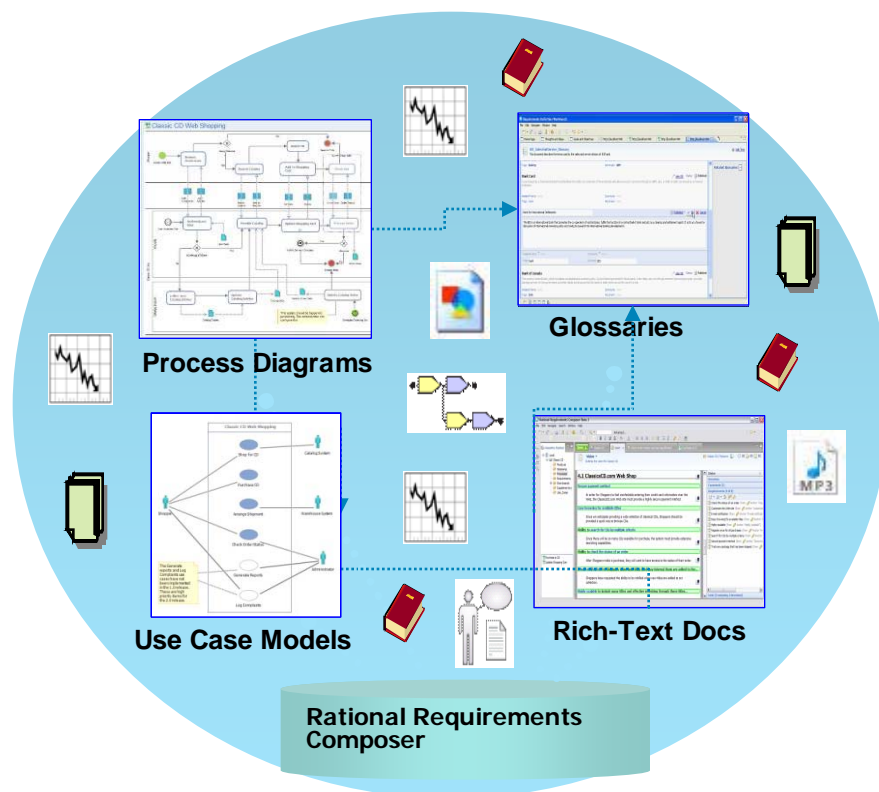
- Dog for sale: eats anything and is fond of children.
- Dinner Special - Turkey \$2.35; Chicken or Beef \$2.25; Children \$2.00.
- We do not tear your clothing with machinery. We do it carefully by hand.
- Wanted. Man to take care of cow that does not smoke or drink.



## Rational Requirements Composer *Requirements Definition and Just Enough Management*



- Rational Requirements Composer **ideally suited to managing different types of requirement information** including:
  - Process flows,
  - User interface sketches
  - Storyboards, Use cases,
  - Rich Text Documents, Glossaries
  - Many other imported artifact types
- Artifacts can be:
  - Categorized/managed
  - Reviewed/commented on
  - Versioned through snapshots/baselines
  - Scoped through collections
  - Linked internally and externally
  - Included in queries/views and reports



## Rational Requirements Composer – Authoring and Reviewing

### Rich Authoring Environment

The screenshot displays the Rational Requirements Composer interface with several key components highlighted:

- Use Cases:** A diagram showing a customer actor interacting with use cases like 'Submit application' and 'Determine eligibility'.
- Rich Text Requirements:** A text editor window titled 'Account Opening' containing sections for 'General' (Thoughts and Ideas, Goals and Objectives, Stakeholder Requirements, Issues and Risks) and 'Use Cases and Storyboards' (Add New Account Use Case Diagram, Update New Account Use Case Diagram, Login Storyboard).
- Glossaries:** A table listing terms like '54', '23', '14', '00', and '00' with corresponding icons.
- UI Sketching and Storyboarding:** A visual design area showing mobile device screens and a storyboard flow.
- Process Sketching:** A flowchart diagram with steps like 'Receive application form', 'Validate Form', 'Update form with reasons for refusal', and 'Update with new ID card'.

### Web Review and Approval

The screenshot shows the Rational Requirements Composer interface accessed via a web browser (Mozilla Firefox). The main content area displays a 'JKE Account Opening Description' with a list of requirements and associated comments.

- Wiki style interface**
- Categorize / Tag**
- Comment**
- Review / Approve**



### Rational. RequisitePro

- Integrate requirements across the application lifecycle

## Elaborate scenarios to uncover customer needs

Describe flows and capture requirements in **Rich-text Documents**

Define and reuse common terminology in **Glossaries**

### 1. Brief Description

A registered customer has applied to open a new **Account**. Based on existing **Customer** information, the eligibility system will use predefined heuristics to determine whether or not the customer is eligible. In certain circumstances, intervention by the **Branch Manager** may be necessary.

### 2. Basic Flow of Events

2.1 BEGIN

The use case starts when the bank's web portal has validated the contents of a new **Account** application form filled out by the **Customer**.

#### RETRIEVE CUSTOMER RECORDS

The **Eligibility Subsystem** retrieves existing banking records from the bank's files. Depending on the type of account being requested, third parties may also be consulted (eg: the credit bureau, if the account is a line of credit, etc.)

#### APPLY HEURISTICS

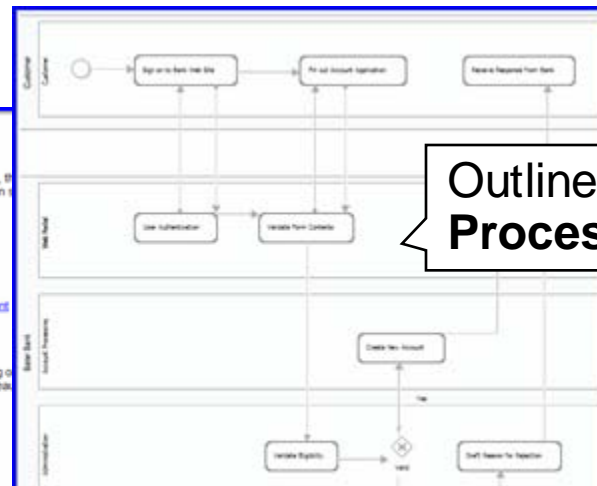
The **Eligibility Subsystem** applies its predefined heuristics to the available data based on account type requested, banking history, **Credit History**, etc. Based on this information the system either decides to automatically approve or automatically reject the application.

The use case ends when the application has been either approved or rejected.

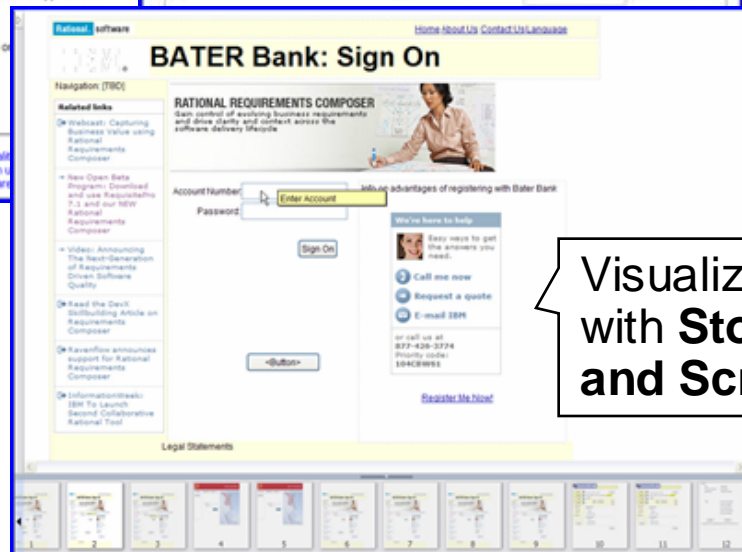
### Alternative Flows

#### Branches

There are multiple alternative flows related to a single area of functionality (e.g. withdrawal facilities, card handling or receipt handling for the Withdraw Cash use case). It improves readability if these conceptually related sets of flows are grouped together.

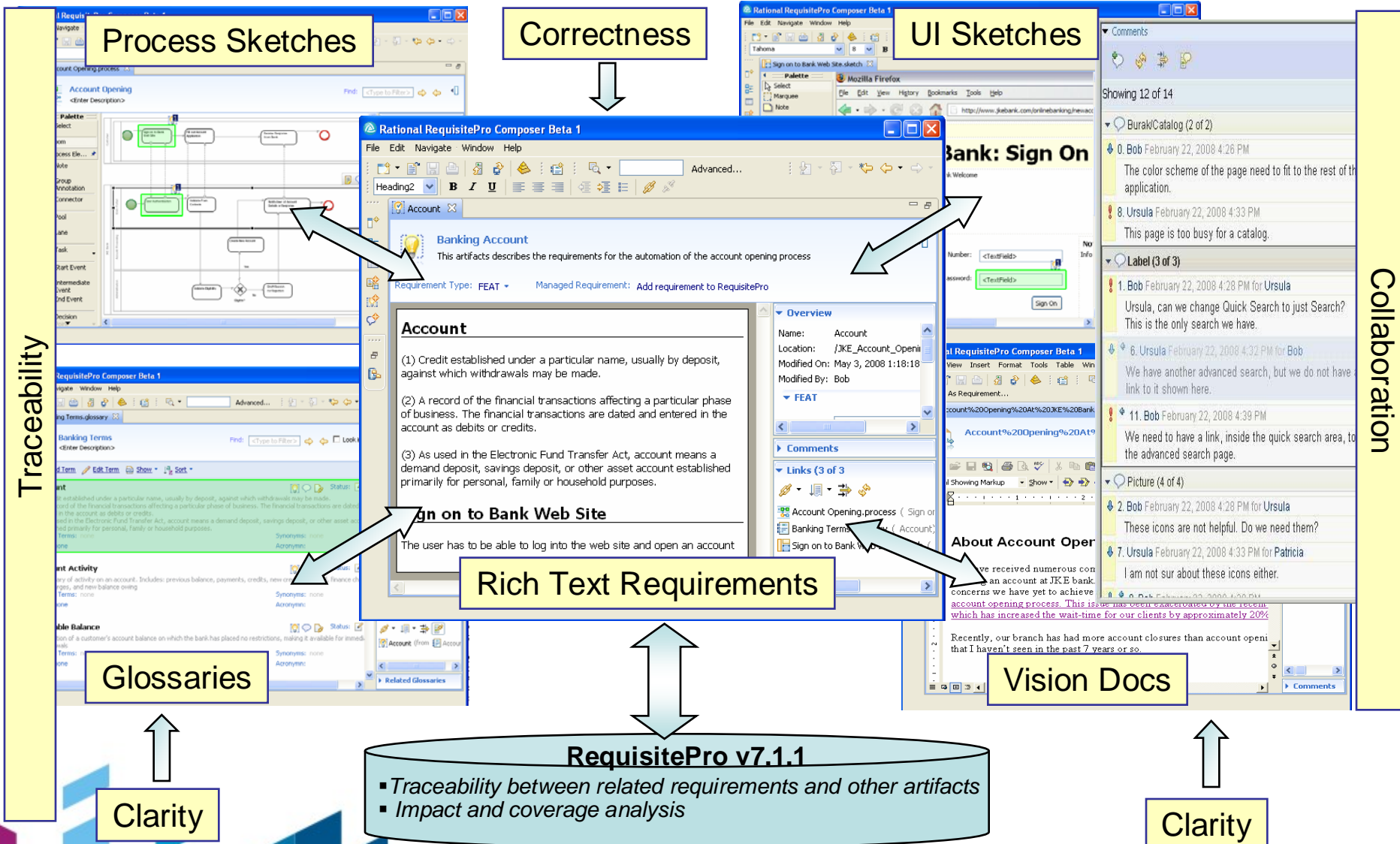


Outline flows with **Process Diagrams**



Visualize scenarios with **Storyboards and Screen Flows**

## Extract Requirements - Clarity, Collaboration, Correctness



## What is Traceability?

### *Requirements Specific Definitions*



A much cited definition of requirements traceability:

"Refers to the ability to **describe and follow** the **life of a requirement**, in both **forwards and backwards** direction."

Another definition of requirements traceability stressing relationships between many kinds of development artifacts:

"Refers to the ability to **define, capture and follow** the **traces left by requirements** on other **elements of the software development environment** and the **trace left by those elements** on **requirements**."

## What is Traceability?

### *Benefits*



- **Process visibility** and auditing
  - Understand where a requirement came from, its importance, how it was implemented, and how it was tested.
- Build the **right system**
  - Verify that all stakeholder needs are implemented and adequately tested or validated.
  - Verify that there are no “extra” system behaviors that cannot be traced to a stakeholder requirement.
- Project **management and maintenance**
  - Understand requirements and project status
  - Understand the impact of changes and manage the implementation of changes
  - Keep the project team in sync

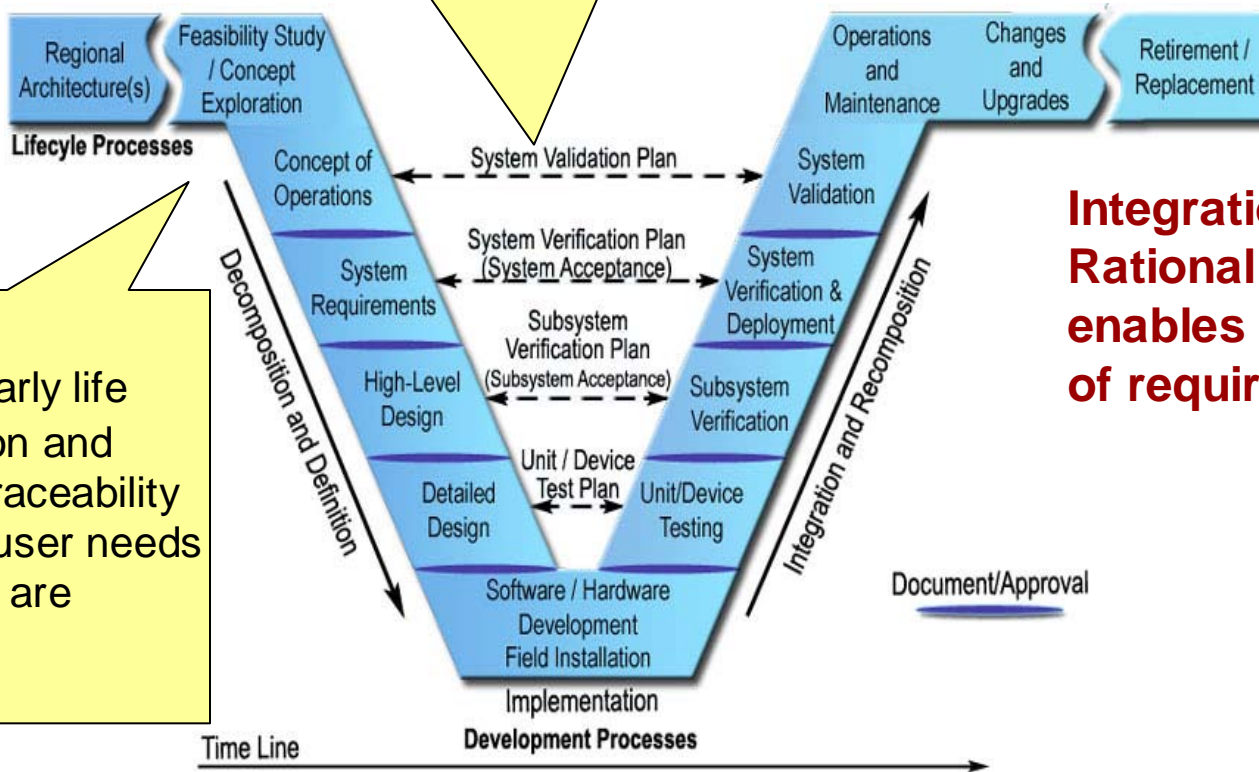
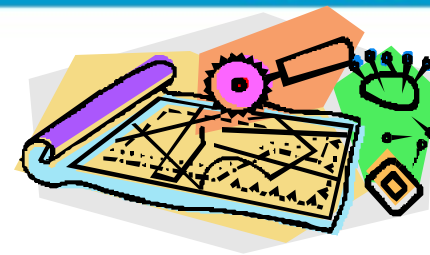


## Rational Requirements Composer

### Traceability Focus

**RRC focuses on early life cycle informal traceability**

**Validation** – Application Life Cycle Traceability ensures that system and subsystem requirements are fully verified.



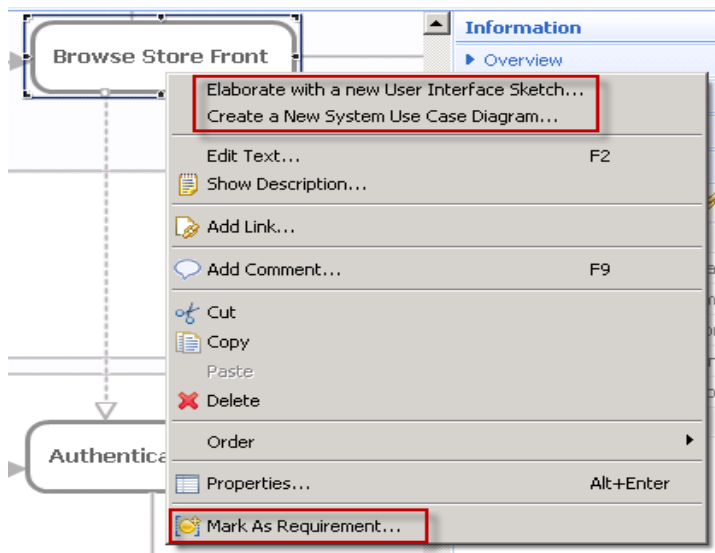
**Origin** – Early life cycle elicitation and stakeholder traceability ensures that user needs and concepts are addressed

**Integration with other Rational products enables validation of requirements**

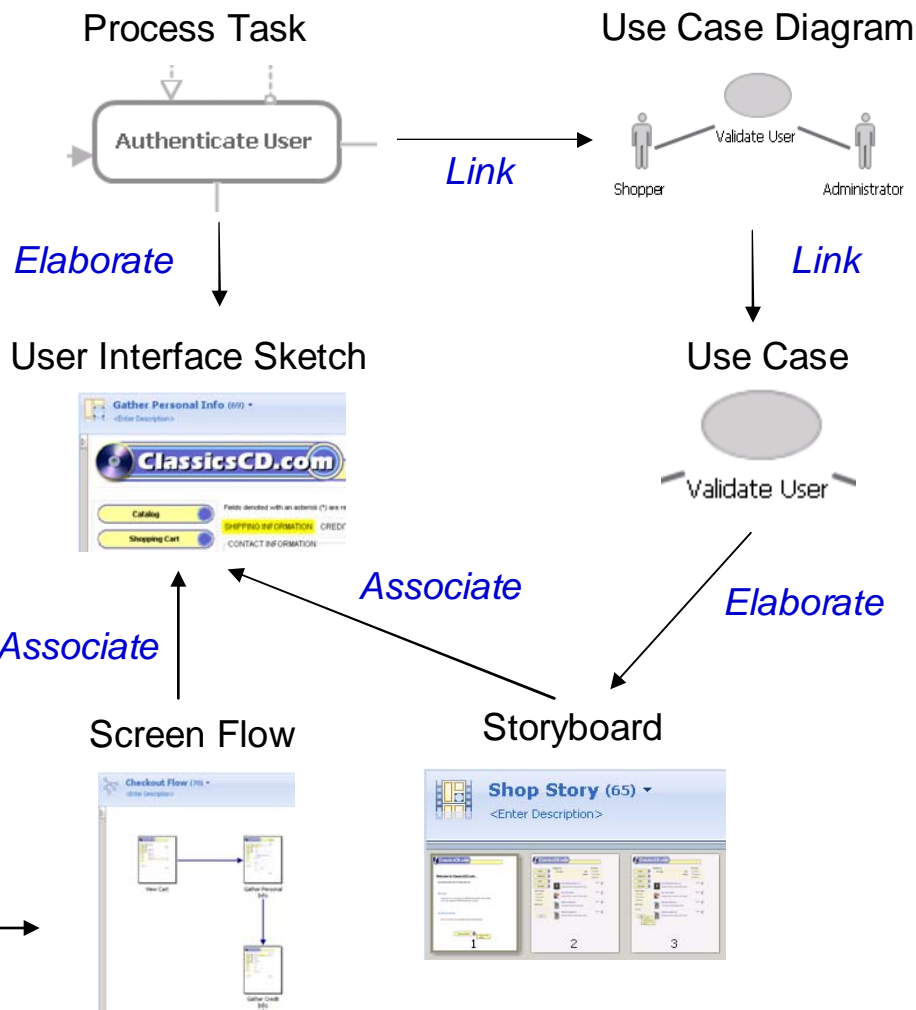
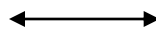


## Rational Requirements Composer Inherent Relationships

- Rational Requirements Composer inherent right click linking support



Requirements,  
Internal Artifacts,  
External Artifacts



## Just enough management - Organization and collaboration creates higher quality requirements

- Customize and share queries and views
- Customize and share dashboards that instantly show progress or problems
- View linkages between requirements and analyze impact
- Quickly search and find

The screenshot displays the IBM Rational Requirements Manager interface. The main window shows a project named 'Test Project' with 18 artifacts. The artifacts are organized into several categories:

- Screen Flow (2):** Checkout Flow, Shop
- Storyboard (1):** Shop Story
- User Interface Part (7):** Cart List Item, Catalog List Item, Mini Cart, Quick Search, rrc6, Side Bar, Side Navigation
- User Interface Sketch (8):** Catalog, CD Details, Checkout Confirmation

The interface includes several filter panels on the left:

- Selected Filters:** Folder: Storyboards
- Saved Filters:** Customer requirements; M1 Requirements;
- Filter Display by Tag:** non-functional;
- Filter Display by Attribute:** Filter by filename, Filter by user, Filter by date, Filter by artifact type
- Filter Display by Folder:** Test Project, Features, Glossaries, Processes, Requirements, Storyboards, Supplementary, Use\_Cases

The right-hand side of the interface shows an 'Information' panel with project details, reviews, and recent comments. Below this, there are sections for 'Page needs work' and 'What about the searching by album titl...', each with a list of requirements and a 'Quick Search' button.

## Just enough management - Review and approval ensures stakeholder buy-in and increases requirement quality

- Effectively engage stakeholders and guide them to consensus with web based review and approval
- Create permanent, auditable record of key reviewers' approval and comments
- Informally and efficiently review requirements and other artifacts with teammates

The screenshot displays the 'NewReview' interface. At the top, there's a title bar with a 'NewReview' dropdown and a description field. Below this is a workflow bar with buttons for 'Overall Review', 'Draft', 'Paused', 'Continue Review', 'Reviewed', and 'Final'. A text box explains the workflow: 'Modify the review as necessary and click "Continue Review" to continue. If artifacts have been modified, click "Get latest version of artifacts" to update them in the review. Participants can not review artifacts while the review is paused.'

The 'Due Date' is set to 8/19/09. There is a text area for 'Instructions for reviewers:'. Below this is the 'Reviewers' section with a table:

Name	Type	Status
Dave	Approver	⏸ ⏸
Donald	Approver	⏸ ⏸

Buttons for 'Add Participants', 'Remove Participants', and 'Change Participants' are present. Below is the 'Artifacts' section with a table:

Artifact	Version	Status
sketch1	Aug 12, 2009 12:56:45 PM	Not started
Sketch With Links	Aug 12, 2009 4:26:35 PM	Not started

Buttons for 'Add Artifacts', 'Remove Artifacts', 'Edit Artifacts', 'Set Status', 'Move Up', and 'Move Down' are present. On the right side, there is an 'Information' sidebar with 'Overview' and 'Comments (0 of 0)' sections.

## Just enough management - Effectively scope and plan

- Create collections of requirements to scope and plan
  - E.g. Requirements for Release 1
- Collaborate around plans
- Version and lock plans to manage change
- Snapshot plans to create permanent history
- Simplify linking with test plans and work items

Name	Modified By	Last Modified Date
Browser Compliance	JazzAdmin	Aug 17, 2009 8:53:45 PM
Check Order Status	TestJazzAdmin	Aug 15, 2009 12:39:45 PM
Catalog size	TestJazzAdmin	Aug 15, 2009 12:39:46 PM
Check the status of an order	TestJazzAdmin	Aug 15, 2009 12:39:53 PM
Common User Access (CUA)	TestJazzAdmin	Aug 15, 2009 12:39:46 PM

## Just enough management - Baselines ensure project integrity over time

- Create snapshots of projects or subsets of projects to permanently record state
- View snapshots online and in complete context
- Compare snapshots to quickly understand what has changed
- Report against baselines to get a better understanding of the project at a point in time
- Do parallel requirements development

The screenshot displays the 'Compare Collection Versions' window, which compares 'My Collection' and 'My Collection version2'. It shows a table of changes between versions, including artifacts that were deleted, added, or changed, along with the user and date of each change.

Artifact	User	Date
Deleted between versions (0)		
Added between versions (2)		
Artifact10	Bob	02/06/2009 09:15:05 PM
Item 8	Abe	02/06/2009 09:15:05 PM
Changed between versions		
Artifact20 (2)		
Artifact20	Bob	02/06/2009 09:15:05 PM
Artifact20	Abe	01/26/2009 10:21:10 AM
Artifact20	Bob	02/06/2009 09:15:05 PM

Overlaid on this are two dialog boxes: 'New Collection' and 'Associate Project Snapshot'. The 'New Collection' dialog shows a collection named 'MyCollection'. The 'Associate Project Snapshot' dialog shows a list of project snapshots, including 'SVD Snap L-1 (Aug 15, 2009 1:03:32 PM)'. A 'Snapshot' dialog is also visible, showing a snapshot named 'Aug 17, 2009 8:44:20 PM' with a description 'This is my initial baseline'.

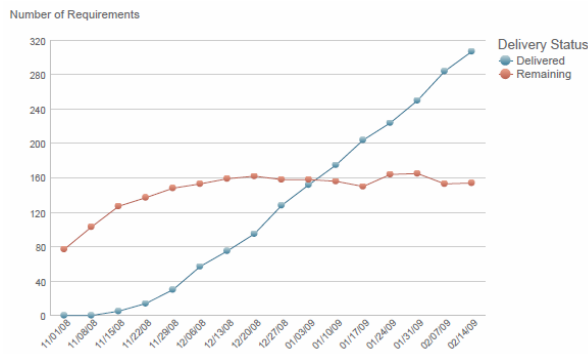
## Just enough management - Reporting and document generation give limitless flexibility on viewing requirement data

- Generate pre-configured and custom reports against all requirements data
  - Requirements, use cases, UI sketches....

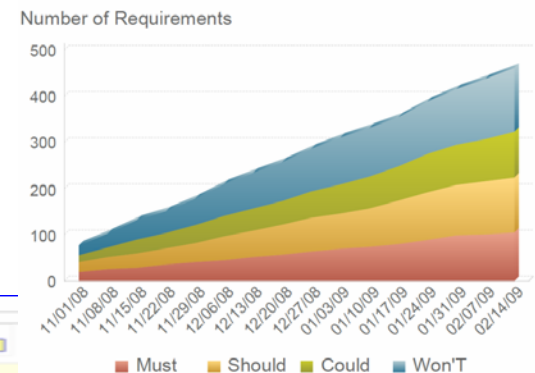
- Automatically generate specs for handoffs, contracts, and traditional reviews
- Traceability, Requirements Survey, Audit History, UI Specification, Review Summary, Use-case Survey

Requirements Traceability Matrix

ID	Tag	Name	Status	Priority	High Level Requirement	Related Defects	Related Activities	Related Test Cases
220	UC1779.1	The system shall respond to inquiries within two seconds.	Approved	Should	<a href="#">FEAT1699</a>			
465	UC1779.2	The user assistance will contain instructions on database configuration and extensibility	Validated	Won'T	<a href="#">FEAT32348</a>			<a href="#">Platform support is sufficient so customers are not forced to deploy new operating systems</a>
783	UC1779.4	Fault Tolerance	Validated	Should	<a href="#">FEAT44356</a>			<a href="#">Verify session timeout</a>
142	UC1781	<b>Generate reports</b>	Validated	Could				<a href="#">Edit a data set</a>



Smarter Planet Online Auction



Widget Properties					
ID	Widget Type	Label	Description	Selected	Enabled
24	Part			false	true
2	Picture		Access the catalog	true	true
3	Picture		Access the shopping cart	true	true
4	Picture		Access the payment information	true	true
5	Picture		Check the order status	true	true
1	Part			true	true
6	Group			true	true

## Managing requirements with Rational RequisitePro

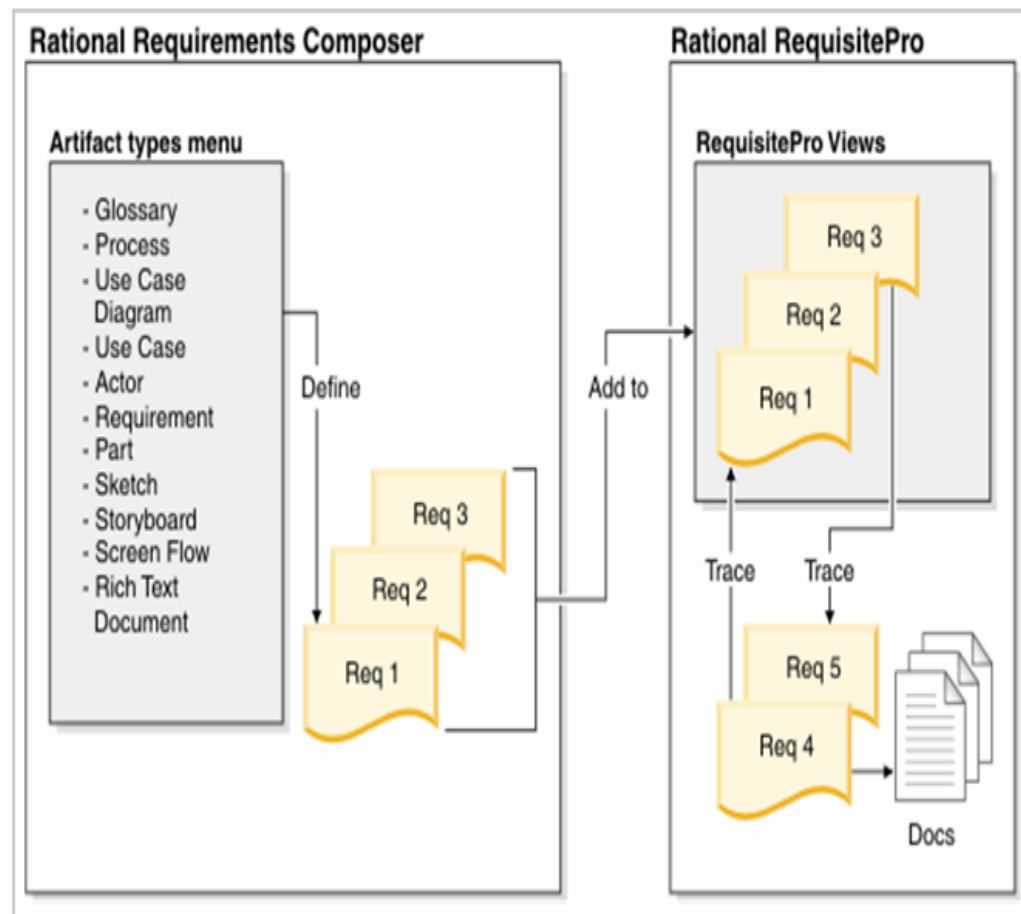
- Create requirements in either Rational® RequisitePro® or Rational Requirements Composer and then add the requirements from one tool to another.
- Rational RequisitePro provides requirements management capabilities including traceability, views, and queries.
- Complement Rational RequisitePro capabilities with a variety of requirements definition techniques in Rational Requirements Composer.
- Users can:
  - Configure a connection to a project in RequisiteWeb
  - Synchronize attribute groups and requirement types
  - Add requirements to Rational RequisitePro
  - Import requirements from Rational RequisitePro
  - Elaborate and Collaborate in Rational Requirements Composer
  - Navigate between both tools



## Rational RequisitePro Integration Scenarios – Scenario 1

Define requirements in RRC and add the requirements to a Rational RequisitePro® project.

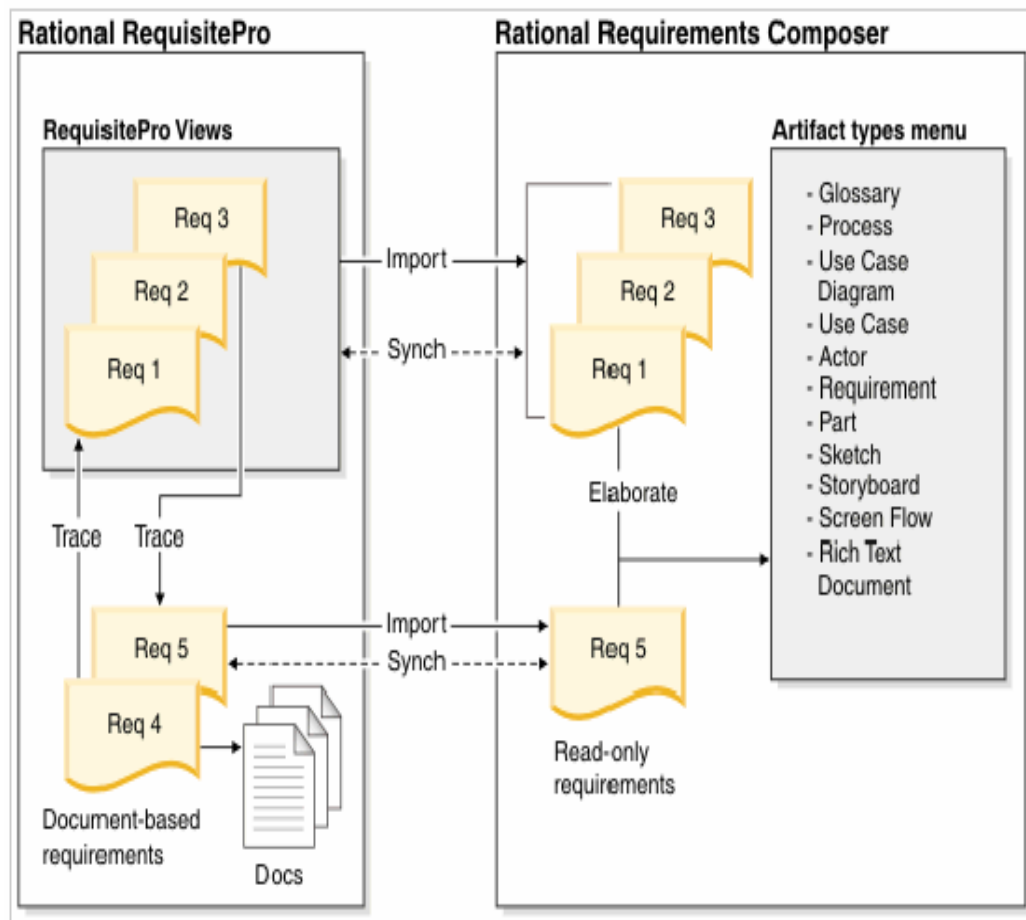
- Define supporting artifacts for requirements in RRC by using requirement definition techniques
- Create requirements from definition artifacts in RRC
- Add requirements to a project in RequisiteWeb. Add traceability to and from other requirements in Rational RequisitePro, including document-based requirements
- Update requirement text in RRC. The requirement text is read-only in Rational RequisitePro. View rich text that is created in RRC as rich text in RequisiteWeb.
- Modify attributes in either tool
- Changes are updated in both tools if the synchronization options were enabled when the integration was configured.



## Rational RequisitePro Integration Scenarios – Scenario 2

Author and manage requirements in a Rational® RequisitePro® project and then import them into a RRC project for collaboration, more definition and elaboration.

- Create requirements in Rational RequisitePro views.
- Add traceability to and from other requirements in Rational RequisitePro, including document-based requirements.
- Import requirements into RRC Requirements Composer from RequisiteWeb.
- Elaborate the requirements in RRC by using requirement definition techniques
- Modify attributes in either Rational RequisitePro or RRC



## RequisitePro Integration – Configure a connection

The screenshot displays the Rational Requirements Composer interface with two dialog boxes open for configuring a RequisitePro connection.

**Project Login Dialog:**

- Server:
- Project:
- Username:
- Password:

**Synchronization Options Dialog:**

- Push requirement changes to RequisitePro
- Pull requirement changes from RequisitePro

**Connections for Synchronizing Requirements:**

Tool	Action
RequisitePro:	Server: http://localhost:11080/ReqWeb Project: RRCReqpro <a href="#">Edit Connection...</a>
DOORS:	<a href="#">Create connection...</a>

A red arrow points from the [Edit Connection...](#) link in the Connections list to the Project Login dialog. The [Edit Connection...](#) link is also circled in red.

## RequisitePro Integration – Synchronize attribute groups and requirement types

The screenshot displays the Rational Requirements Composer interface. A dialog box titled "Synchronize Attribute Groups with RequisitePro" is open, showing the synchronization process. The dialog has two main panes: "Composer Attribute Groups" and "RequisitePro Requirement Types".

**Composer Attribute Groups:**

- FEAT: Feature
- SUPL: Supplementary
- SUPP: Supplemental
- TERM: Term
- UC: Use Case (highlighted)

**RequisitePro Requirement Types:**

- FEAT: Feature
- NONE: Default for documents without requirements and design constraints.
- STRQ: Stakeholder Request
- SUPL: Supplementary
- TERM: Glossary Item
- UC: Use Case

Buttons between the panes include "Export >>" and "<< Import".

Below the panes, there is a text area with the following content: "Attribute group used for functional requirements. When adopting the use case approach, all functional requirements are documented in use case documents."

At the bottom of the dialog are buttons: "< Back", "Next >", "Finish", and "Cancel".

The background interface shows a project named "Classic CD" with a tree view of "Attribute Groups" (Feature, Market, Origin, Supplemental, Supplementary, Term, Use Case) and "Members" (George, Munaf, Vishy). A "Connections for Synchronizing Requirements" section at the bottom shows a connection to RequisitePro with the following details:

Requirement Source	Server	Project	Action
RequisitePro:	http://localhost:11080/ReqWeb	RRCReqpro	Manage Requirements... Import Requirements... Synchronize Attribute Groups...
DOORS:	Create connection...		

## RequisitePro Integration – Share requirements

The screenshot displays the Rational Requirements Composer interface with two dialog boxes open for RequisitePro integration. The left dialog, 'Import Requirements from RequisitePro', shows a tree view of packages under 'RRCReqpro', including 'Coverage Analysis', 'Features and Vision', 'Glossary', 'Impact Analysis', 'RRC Requirements', 'Stakeholder Requests', 'Supplementary Requirements', and 'Use Cases'. The right dialog, 'Create Requirements in RequisitePro', lists several requirements with checkboxes, such as 'Shop for CD - Possible New Account', 'Arrange Shipment - Inventory not available', and 'Check Order Status - Invalid Customer Credentials'. Below these dialogs is a 'Connections for Synchronizing Requirements' section with fields for 'RequisitePro' (Server: http://localhost:11080/ReqWeb, Project: RRCReqpro) and 'DOORS' (Create connection...). At the bottom right, three buttons are circled in red: 'Manage Requirements...', 'Import Requirements...', and 'Synchronize Attribute Groups...'. Red lines connect these buttons to the 'Finish' buttons of the two dialog boxes above.

## RequisitePro Integration – Elaborate and Collaborate in RRC

The screenshot displays the Rational RequisitePro interface. The main window shows a requirement titled "UseCase2 (321)" with the text "Example requirement". Below this, there are sections for "Related Features" and "Additional Information".

**Related Features:**

- The related high level requirements are:
  - [Add and remove CDs available for sale](#)
  - [Easy browsing for available titles](#)
  - [Search for CDs by multiple criteria](#)
- The child system requirements are:
  - [Shop for CD - Allow suggestion options](#)
  - [Shop for CD - No matching ids](#)
  - [Shop for CD - No new account](#)
  - [Shop for CD - Possible New Account](#)
  - [Shop for CD - Post Condition](#)
  - [Shop for CD - Quit](#)
  - [Shop for CD - Remove item](#)
  - [Shop for CD - Search by selected criteria](#)
  - [Shop for CD - View Current Items](#)

**Additional Information:**

The associated sketches for this requirement are shown below:

A screenshot of a web catalog is shown, titled "CLASSICSCD.COM". The catalog features a navigation menu with "CATALOG", "SHOPPING CART", and "CASHIER" buttons. Below the navigation, there is a "Shopping Cart" section with a table:

Item Name	Short Description	\$\$ Price
Item Name	Short Description	\$\$ Price

Next to the table is a "Checkout" button. To the right of the table is a "Sort List By" section with radio buttons for "Composer", "Composition", and "Performer".

On the right side of the RequisitePro interface, there is an "Information" panel showing a comment:

1. **Need additional related requir...**

Vishy Aug 5, 2010 10:47:4...  
For Munaf

Hi Munaf ,

We need to add additional requirements here

Thanks

Vishy

## RequisitePro Integration – Manage in RequisitePro

The screenshot displays the Rational RequisiteWeb application in a Windows Internet Explorer browser. The browser address bar shows the URL: `http://localhost:11080/ReqWeb/MainFrame.jsp`. The application header includes the Rational logo and navigation links: [Rational Home](#), [RequisitePro Home](#), [User Profile](#), [About](#), and [Log Out](#).

The main interface is titled "RRCReqpro" and features a search bar and a toolbar with actions like "Create Child", "Change Type", "Create Discussion", "Delete Requirement", and "Generate Bookmark". A left-hand navigation tree shows a hierarchy of folders: Coverage Analysis, Features and Vision, Glossary, Impact Analysis, RRC Requirements (expanded to show SUPL1: Supplementary1, UC1: UseCase1, and UC2: UseCase2), Stakeholder Requests, Supplementary Requirements, Use Cases, and Requirements Management Plan.

The main content area displays the "Requirement Properties" for the selected requirement, "UC2: UseCase2". The "Name" field contains "UseCase2". The "Text" field is marked as "Text is Read Only: This requirement contains rich text." and contains the following content:

- Requirement: Basic Flow**
- Brief Description**  
The use case begins when the [Composer: Shopper](#) chooses to find CDs
- Applicable Roles**  
The primary and secondary roles associated with this requirement are [Composer: Shopper](#) and [Composer: Catalog](#) System respectively
- Key Scenarios**  
1. HELPFUL SUGGESTIONS  
The system presents the following:

The footer of the application includes the IBM logo and the copyright notice: "(C) Copyright IBM Corp. 1995, 2010. All Rights Reserved." The browser status bar shows "Local intranet" and a zoom level of "100%".

## RequisitePro Integration – Navigate between both tools

The screenshot shows the RRRReqpro application window. On the left is a tree view with folders like 'Coverage Analysis', 'Features and Vision', 'Glossary', 'Impact Analysis', 'RRC Requirements', 'Stakeholder Requests', 'Supplementary Requirements', 'Use Cases', and 'Requirements Management Plan'. Under 'RRC Requirements', 'UC2: UseCase2' is selected. The main area displays the details for 'UseCase2', including a 'Text' field with a red warning 'Text is Read Only: This requirement contains rich text.', a rich text editor containing the requirement content, and metadata fields for 'Package', 'Location', and 'Type'.

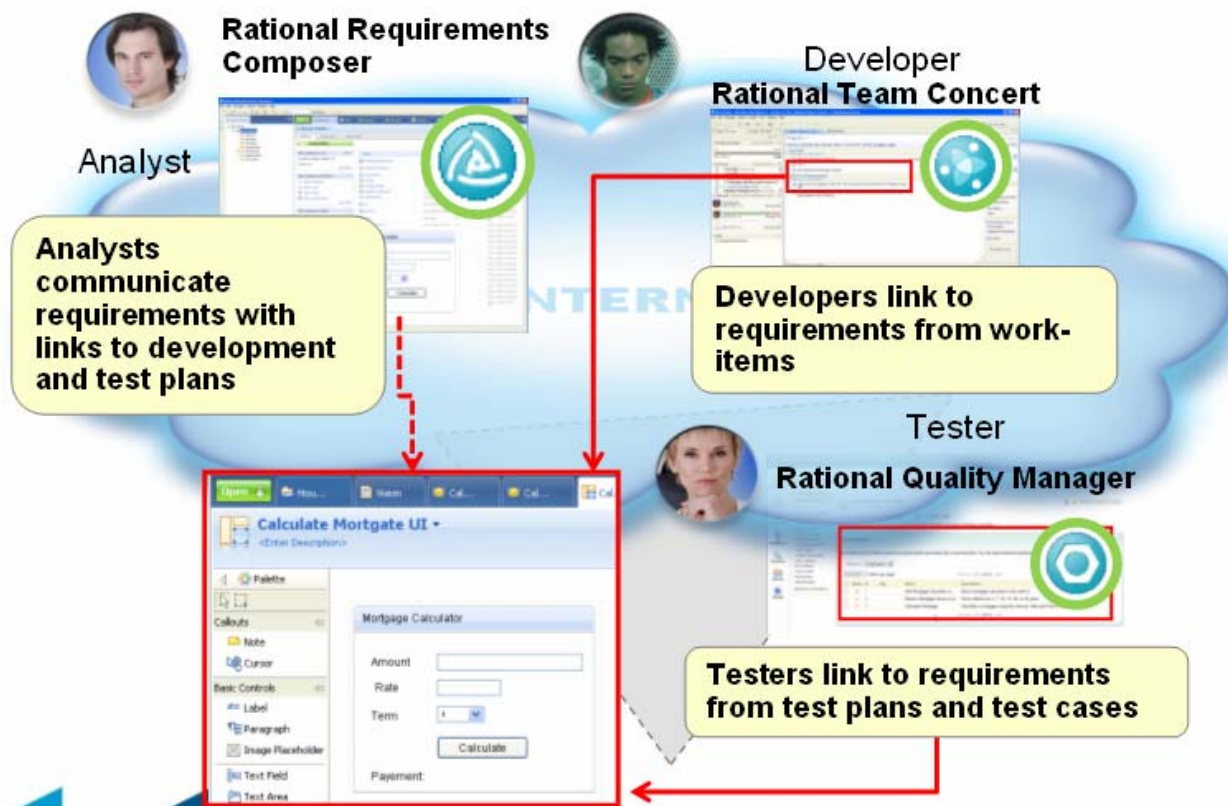
Name	Managed Requirement	Last Modified Date	Artifact Id	Requirement Type
UseCase2	UC2	Aug 5, 2010 10:42:20 AM	321	Use Case
UseCase1	UC1	Aug 5, 2010 10:20:36 AM	318	Use Case
Supplementary1	<u>SUPL1</u>	Aug 5, 2010 10:20:58 AM	320	Supplementary
Globalization (G11n)				Supplemental

Click to navigate to a RequisitePro requirement



## Collaborative ALM integrations provide traceability across Requirements, Work items, and Test cases!

- Attain cross-role alignment by linking requirements to work items and tests
- Link collections and requirements with:
  - **Rational Team Concert** release plans and work items
  - **Rational Quality Manager** test plans and test cases
- Dashboard viewlets show cross repository linkages
  - Requirements under test
  - Requirements being implemented



## Requirements Analysis to Solution Design Rational Software Modeler & Architect

### Requirements Driven Development

*Flexible integrated requirements-driven software design*

- ✓ Export use-case models and business process models
- ✓ Reuse requirements artifacts for product design elaboration
- ✓ Trace requirements to product designs

**Rational Requirements Composer**


JAZZ TEAM SERVER

Sketches and storyboards

Business processes

Use cases

Rich text




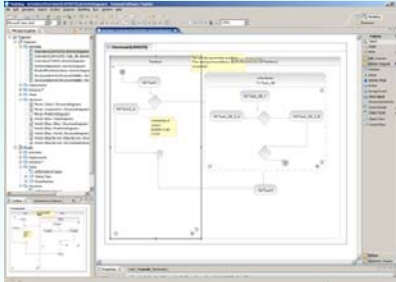
**Business Analyst**



**Rational Software Modeler**

Use cases

Activity Diagrams



**Software Architect**

## See for yourself at jazz.net

- Download Requirements Composer 2.0.0.2 iFix1
- Learn more from the blog and library
- Have discussion with developers and community
- Report defects or request enhancements
- Stay informed!

Projects >

Rational Requirements Composer / Rational DOORS Requirements Professional

Overview

What's New

Integrations **New!**

Release Plan

Related Resources

Rational Requirements Composer / Rational DOORS Requirements Professional 2.0.0.2 iFix 1 Online Help

**Rational Requirements Composer / Rational DOORS Requirements Professional**

**Collaborative requirements for the whole team**

**NEW!**

Rational are currently working to provide a complete next-generation requirements platform for better requirements practices and to enable larger sets of stakeholders and team members to participate in a requirements driven development life cycle. This will enable team members to capture, manage, analyze, and report on requirements.

This requirements platform is intended to deliver proven requirements management methods and capabilities for fast-paced, agile-at-scale project teams.

Rational Requirements Composer was the first product built on this platform and provides team members with the ability to define, and elicit user requirements. The next offering, provisionally known as DOORS Requirements Professional will provide market driven project teams with the latest requirements management and analysis capabilities completely over the web and help them achieve the cost saving and management benefits of requirements driven development.

→ Buy at [ibm.com](#)

**Download**  
2.0.0.2 iFix 1  
June 30, 2010

**Download 3.0 M7a**  
July 19, 2010  
This is a pre-release Early Access Trial milestone of Rational Requirements Composer / Rational DOORS Requirements Professional.

[All Downloads >](#)

**Join the conversation**  
[Ask questions in the forum](#)

**Go see it now!**

<http://jazz.net/projects/rational-requirements-composer/>

# Innovate2010

The Rational Software Conference



# THANK YOU



# Backup



## Requirements Composer – Architecture Principles

- **Enterprise-Scale**
  - Built upon a scalable Internet-based repository supporting artifacts and links
  - Linked artifacts allow for drill-down and query across enterprise-wide artifacts
- **Focused on defining correct requirements**
  - Elicit and capture requirements using a variety of techniques that can be combined to provide better elaboration of the business context for requirements
  - While artifact-centric the tool emphasizes on the relationship between artifacts to create a network of information to support traceability between requirements and analysis artifacts
- **Consumability**
  - Appealing visual editors over clearly defined, small and simple artifact models
  - Web-like navigation between artifacts, little need for ‘navigator view’
  - Support a path from more free-form to formal/validated artifacts
- **Openness and extensibility**
  - Artifacts are stored in an open, extensible information model available over the Internet-based repository
  - Neither tool or repository requires adoption of a particular programming language or framework

## Requirements Composer – Data Model Principles

- **Simplicity**
  - The data that tools create/manipulate/display is held in artifacts. There is no data outside of artifacts. Artifacts can be manifested in files, but natively they live on the internet/intranet
  - Artifacts have URIs/URLs that identify them and are used to access them. We follow the usual internet standards and guidelines for this.
  - Artifacts may have within their contents uris of other resources. This is how links/relationships are represented in this model.
- **Extensibility**
  - Artifacts are defined in terms of their textual formats. The way we define artifacts is the same way the rest of the internet community does. Examples are HTML, XML
  - Artifacts are shared between tools to extend an existing artifact by adding additional information to it in a safe manner.
- **Understandability**
  - The ability for a human to be able to understand a artifact structure, even if the artifact is primarily produced/consumed by a tool.
- **Stability**
  - The ability for a artifact "schema" to change over time without impacting clients.

## Requirements Composer – Consumability Principles

- Target user is a business person, not a developer/IT person
- Understands the business or business area and much more likely has an MBA not MSc
- No expectation that what they produce will execute (this is not a development tool)
- Familiar with tools like MSOffice, and web-based business applications
- They generally run on last year's laptop, no business need for high-performance machine

