Use Case Management with Rational Rose and Rational RequisitePro

A Rational Software White Paper

Rational®
the **e-development** company™

Table of Contents

Introduction

1. How do you organize your use cases?
2. Can you tell in which release a particular use case is implemented?
3. How do you know that the entire system functionality is tested?
4. Which tests are affected by a change in a use case?

If the inability to answer these questions has caused you frustration, schedule setbacks, or the delivery of products that have missed the mark, read on.

Integrated Use Case Management enhances Rational Rose use-case modeling with powerful requirements management capabilities. By extending use cases beyond diagrams with sortable attributes, documents, and traceability, *Integrated Use Case Management* helps manage large numbers of use cases across your team. This is the tightest and most robust integration in the market between a visual modeling tool and a requirements management tool.

Why Manage Requirements?

Requirements management is a systematic approach to finding, documenting, and managing requirements. Without it, more than two-thirds of projects end up missing user needs, are late or over budget (Standish Group, *CHAOS* Report available at www.standishgroup.com). Why so many? Primarily because managing requirements is about managing change – and managing change is hard. Plus change is pervasive. We live in a dynamic world: customers change their minds, competitors come up with better solutions before we deliver ours, the business environment changes. Being open to this changing environment is a good thing, but it can bog you down. Change in itself isn't bad. The evil beast is *uncontrolled* change – change whose impact is not measured before it happens. By managing requirements you are more likely to deliver a timely solution that solves your customers' real problems.

Why Manage Use Cases?

By providing a user's view of what the system should do, use cases *are* requirements. As such, use cases should participate in the management of *all* system requirements. Most software projects have numerous use cases, all of which have different priorities and dependencies – just like any other requirement. For example, a use case describing the processing of an order on the Web might stem from the business need of generating more revenue through the Web. By establishing a tangible dependency between the use case and its business need, you can better respond to change affecting either of these requirements. And prioritizing the importance of implementing this use case versus another helps you know where to start.

Managing use cases along with all other requirements is key to understanding the state of your project and better enables you to deliver the right system. The value of *Integrated Use Case Management* is that it seamlessly integrates uses cases with your requirements information.

Setting up the Integration

Integrated Use Case Management begins by associating your Rational Rose model with a Rational RequisitePro project. This association provides the context for selecting use-case document templates and use-case attributes from the Rose environment. You can establish this association either at the model level or at the package level, where each package may be associated with its own RequisitePro project. The package association lends itself to large software projects that might use either multiple RequisitePro projects (typically one per subsystem) or different use case document templates (for business use cases vs. system use cases).

A Rational RequisitePro project consists of a number of Microsoft Word documents and a database (Microsoft Access, Microsoft SQL Server or Oracle) to organize the requirement information. Use case documents in RequisitePro contain use-case textual descriptions, just like what you may be writing today. Requirements in these documents are linked to a database that stores additional requirement information, such as attributes, traceability links, versioning, change history, project security and more. From the RequisitePro database, you can query the requirement information to check coverage and measure the impact of change. You can also easily navigate to the RequisitePro Word environment and back to Rose.

Requirements Management Capabilities in Rational Rose

Requirement management capabilities are visible from the standard shortcut menu in the Rational Rose browser –right-click on a use case in Rose to view the new use-case menu options (see Figure 1). They include:

- *Use Case Document* to create a new use-case document or associate the use case with an existing RequisitePro use-case document

- *Requirement Properties* to view and edit attributes and traceability links to the use case

- *View RequisitePro Association* to view the Rational RequisitePro context for that use case (the associated use case document template and use case attributes – set via a requirement "type")
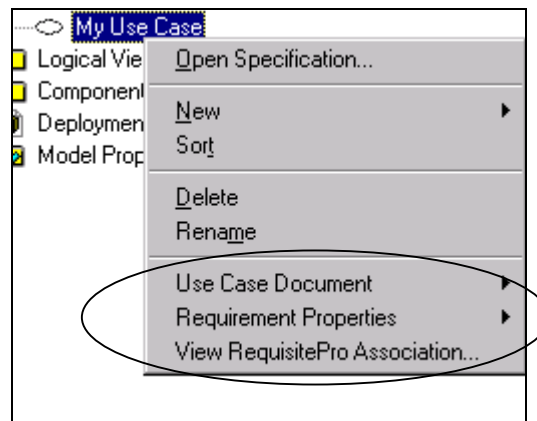

Figure 1: New options on the use-case shortcut menu in Rose

Lets look at these capabilities in more detail.

Use Case Document

If you are an existing Rose user, you may be writing use-case documents and attaching them to your use-case model via the Rose *External File* property. The new *Integrated Use Case Management* capability goes beyond simply attaching a file to a Rose use case. Because the documents attached to your use cases are Rational RequisitePro documents, you benefit from the following advantages:

- *Use case documents are based on proven use-case document templates.*
  Integrated Use Case Management provides Rational Unified Process use case templates. These templates contain informative guidelines as well as use case formatting, saving you time and providing consistency from document to document.

- *Requirement text is clearly marked.*
  Requirement text is visually differentiated from additional descriptive information in the document (See Figure 2.). This makes it easier to "see the trees in the forest."
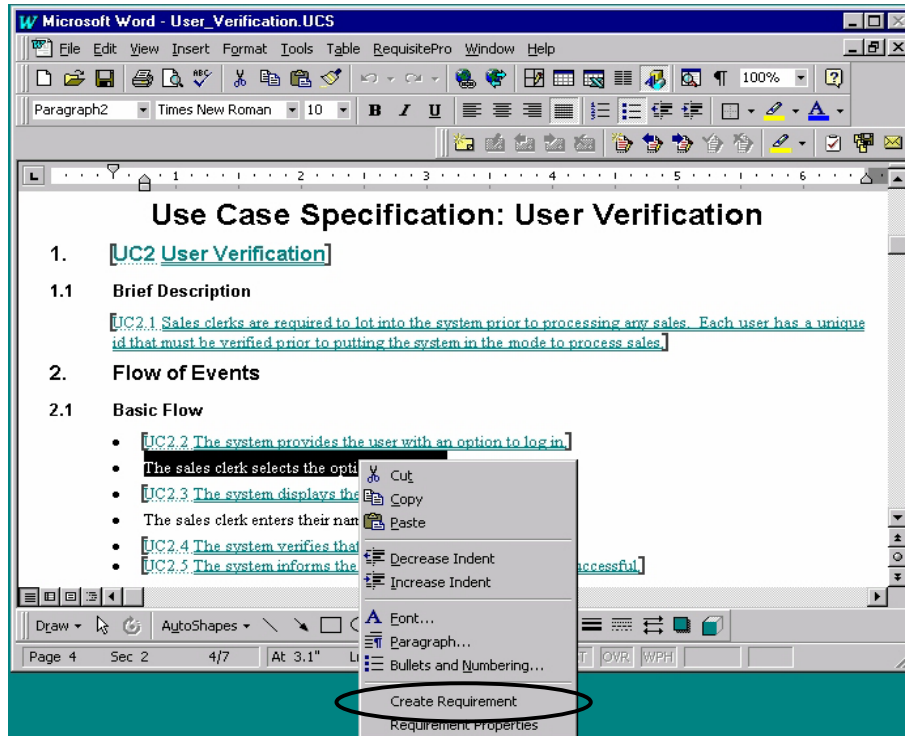

Figure 2: Requirements in a Rational RequisitePro document

- *Any modification to use case documents is automatically tracked.*
  Information about who modifies what, when, and why is stored in the Rational RequisitePro database. These revisions help you gain control of use case changes.

- *Requirements in use case documents can be linked to other requirements they may relate to.*
  By tracing use cases to business requirements, feature-level requirements, tests, or even other use cases you can more easily measure the impact of change on related requirements and verify coverage.

To associate a use case document with a Rose use case, right-click on the use case in the Rose browser, and select *Use Case Document➔New* from the shortcut menu. The RequisitePro Word environment is launched and your template-based document is displayed, ready for editing. You can also associate an existing RequisitePro document to a Rose use case by using the *Use Case Document➔Associate* menu item.

Requirements Properties

The second new addition to the use-case shortcut menu is "Requirements Properties." Requirements, and similarly use cases, are not just text. They have additional properties, such as attributes, traceability links, revision history, and particularly for use cases – diagrams.

Attributes help manage scope. Especially helpful in iterative development, attributes provide an easy way to scope manage each iteration of your project. They make the process of deciding the use cases to implement in a particular release more objective. Too often organizations decide which use cases to implement based on personal agendas, emotions, or pet peeves. Poor decisions made early carry through to implementation and are more costly to change the farther the project gets in its lifecycle. Attributes provide a simple way to assign non-emotional weight to use cases and requirements alike.

Traceability helps measure the impact of change and ensure requirements coverage. For instance, if a business need changes, what use cases might be impacted? By establishing traceability links, you can query the requirements to answer questions like "Are all business needs implemented at the use case level?" or "Are there test requirements for all the use cases?"

(Note for Rational Unified Process users: you may want to review the *Traceability Strategies for Requirements Management With Use Cases* white paper available in RUP. The paper outlines various traceability approaches, depending on your needs.)

Revisions helps you track who changes what, when and why, to provide an audit trail of requirement changes. This helps you measure the stability of requirements and concentrate on more stable requirements first, inherently diminishing the amount of change.

To set use case attributes and relationships in Rational Rose, right-click on the use case in the Rose browser, and select *Requirement Properties*➔*Open* from the shortcut menu. In the dialog box (see Figure 3), click the *Attributes* tab, and set attribute values. Note that you can change the out-of-the-box use case attributes and their values in the RequisitePro project associated with your model. From this same dialog box, click the *Traceability* tab to establish traceability between the use case and other requirements.
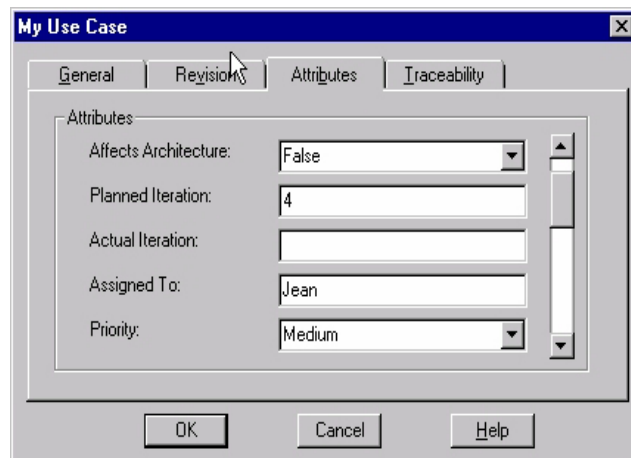

Figure 3: Viewing and editing use case requirement properties in Ratioanl Rose

Benefits of Managing Use Cases

Once you've attached a use case document, or assigned requirement properties to a use case in Rational Rose, the use case is part of your requirement set in RequisitePro. As such you can use all Rational RequisitePro capabilities to sort your use cases (by priority, by iteration number, etc.), to query on specific use cases (only the use cases planned for the next iteration), and even produce requirements metrics.

Using an Attribute Matrix View in Rational RequisitePro (see Figure 4), you can view all or a select subset of use cases and their respective attributes. This helps you organize the use case information answering the first question at the beginning of this article: *How do you organize your use cases?* You can run queries to determine which use cases are assigned to which designer, how difficult they are to implement, or in which release they should be implemented *(Can you tell in which release a particular use case is implemented?)*.

Requirements:	Property	Priority	Assigned To	Planned Iteration	Affects Architecture
⊞ UC2: User Verification	Name	Medium	Mary	2	True
⊟ UC3: User Maintenance	Name	Low	Jean	3	True
UC3.1: Store managers may add, delete and modify.	Brief Description	High	Jean	3	False
UC3.2: The system provides user administration...	Basic Flow	Medium	Jean	4	False
UC3.3: The system displays the login screen.	Basic Flow	High	Jean	3	False
UC3.4: The system validates the store manager's...	Basic Flow	Medium	Jean	3	False
UC3.5: The user maintenance screen is displayed...	Basic Flow	Low	Jean	4	False
UC3.6: The system provides options for adding,...	Basic Flow	High	Jean	5	False
UC3.7: The system displays the new information.	Basic Flow	Medium	Jean	3	True
UC3.8: The system updates the database with the...	Basic Flow	Medium	Jean	3	False
UC3.9: The system informs the store manager that...	Basic Flow	High	Jean	4	False
UC3.10: The system is ready to perform another...	Basic Flow	Medium	Jean	3	True
UC3.11: The system prompts the user for the new...	Basic Flow	Medium	Jean	3	False

UC3: User Maintenance

Ready   27 requirements

Figure 4: Use case Attribute Matrix View in Rational RequisitePro

Once you have selected the use cases to be implemented in the next iteration, you should verify that test requirements are traced to use cases ensuring that all the functionality will be tested. The Traceability Matrix View in Figure 5 shows the relationships established between use cases and test cases. Using Traceability Matrices, you can query on use cases not yet traced to test requirements (answering the question *How do you know that the entire system functionality is tested?)*. Additionally, testers can query test requirements potentially impacted by modifications in use cases ensuring that they're testing the latest functionality. A suspect link (red slashed arrow in Figure 5) indicates that test case TC1.2 may need to be revisited due to a change in use case UC1.3. Querying on suspect links answers our last question, *which tests are affected by a change in a use case?*
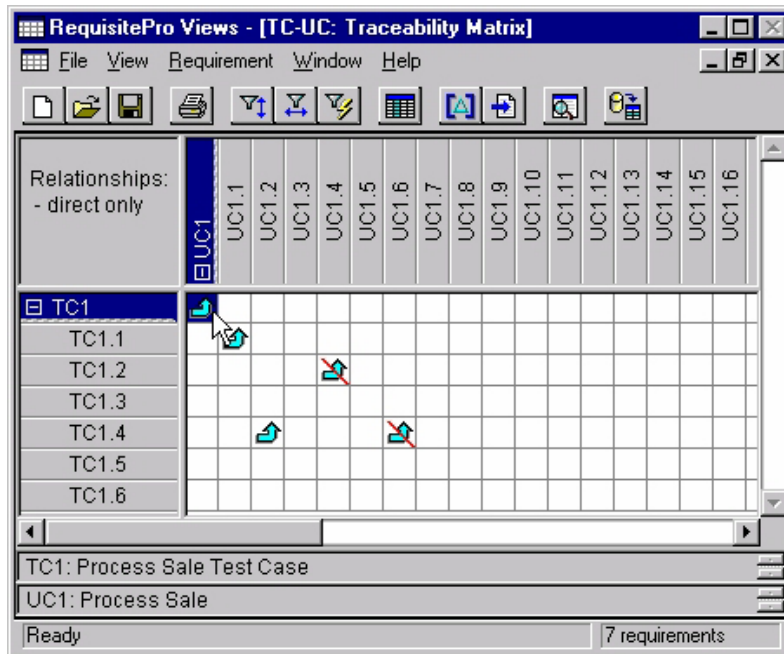
Figure 5: Traceability relationships between use cases and test cases in Rational RequisitePro

Summary

Integrated Use Case Management extends use cases with requirement information. This benefits the Rational Rose user by establishing a real-time window to modify use case attributes, traceability and view revision history from Rose. And the Rational RequisitePro use case document is just a click away. *Integrated Use Case Management* provides requirements managers more accurate and timely information to work with because that information is available from the Rose users' fingertips. By managing use cases in conjunction with other requirements your project can be better scope managed, change can be controlled and coverage can be verified. In short, *Integrated Use Case Management* helps ensure that you are implementing the functionality that was agreed upon, and that this functionality will be fully tested.

Rational®
the **e-development** company™

Dual Headquarters:

Rational Software
18880 Homestead Road
Cupertino, CA 95014
Tel: (408) 863-9900

Rational Software
20 Maguire Rd
Lexington, MA 02421
Tel: (781) 676-2400
Toll-free: (800) 728-1212
Tel: (408) 863-9900
Fax: (408) 863-4120

E-mail: info@rational.com
Web: www.rational.com
For International Locations: www.rational.com/worldwide