

TP559, 06/02

**Rational.** software

The IBM logo, consisting of the letters 'IBM' in a stylized, striped font, is positioned in the top right corner of the page.

## Managing Use Cases Using the IBM Rational Solution for .NET

## Table of Contents

<b>Introduction.....</b>	<b>1</b>
<b>The Challenge of Managing Use Cases .....</b>	<b>1</b>
<b>Integrated Use Case Management Overview.....</b>	<b>3</b>
<b>Enabling Integrated Use Case Management in XDE Models.....</b>	<b>3</b>
<b>Managing Use Cases in IBM Rational XDE .....</b>	<b>5</b>
Use Case Document .....	5
Requirements Properties .....	6
RequisitePro Association .....	7
<b>Managing Use Cases in IBM Rational RequisitePro .....</b>	<b>8</b>
<b>Summary.....</b>	<b>9</b>
<b>References.....</b>	<b>9</b>

## ***Introduction***

---

Use cases describe a system's behavior when interacting with the outside world. In the past decade, use cases have been proven to be an effective way to document system functionality from a user's perspective, a perspective that provides both software teams and their customers a common understanding of the expected behavior of the system to build. By minimizing the risk of misunderstanding, use cases improve the chances a software team has to deliver a successful system.

Whether you are an analyst, a project leader or a developer, on a modern software development project, chances are you have heard of use cases. If you are part of a team who is writing use cases, you likely have faced some of the challenges in managing use cases.

This paper describes **Integrated Use Case Management (IUCM)**, a feature that helps you manage use cases with **IBM Rational RequisitePro®** and either **IBM Rational XDE™ Professional: .NET Edition** or **IBM Rational® XDE™ Modeler**.

Rational RequisitePro is IBM Rational's market leading requirements management tool. Rational XDE Professional: NET Edition is a unique eXtended Development Environment for .NET developers. It extends the Visual Studio .NET Integrated Development Environment (IDE) with UML modeling capabilities, to enhance communication and documentation of software design. Rational® XDE™ Modeler is a tool that enables architects and designers to create platform-independent Unified Modeling Language (UML) models of architecture, business needs, reusable assets, and management-level communication.

Throughout the course of this paper, we will provide solutions for effectively handling these commonly-asked use-case questions:

1. How do I organize my use cases?
2. How can I tell in which release a particular use case is implemented?
3. Which specific business needs does a use case address?
4. Are my use cases staying in touch with the evolving business needs they are supposed to solve?

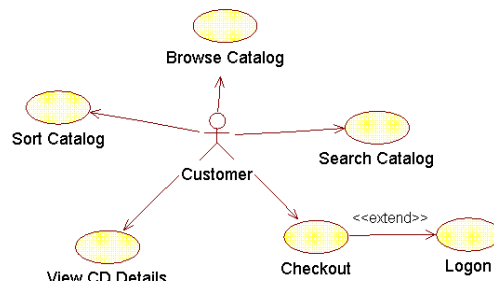
In our experience mentoring customers on using use cases effectively, we have found that the inability to answer these questions causes frustration, schedule setbacks, and the delivery of products that miss the mark.

IBM Rational's IUCM solution provides you accurate and timely answers to those, sometimes difficult, questions.

## ***The Challenge of Managing Use Cases***

---

Use cases are the starting point of object-oriented design, where use case diagrams represent a high level picture of the system functionality (see Figure 1).



**Figure 1: A typical use case diagram**

From use case diagrams, developers typically create sequence diagrams to identify design classes they will implement in code to fulfill the system requirements. In order to create a sequence diagram, developers need to have access to more details than just the use case oval in the use case diagram. Therefore, each use case is further detailed into a set of steps that describes the dialog between the actor (*Customer* in Figure 1) who typically initiates the use case, and the system accomplishing the functionality described by the use case (e.g. *Sort Catalog* in Figure 1). That set of steps is often referred to as the *flow of events* of the use case. Typical use cases have a single basic flow of events (describing an expected user interaction with the system) and several alternate, sometimes called exceptional, flow of events (that record how the system should handle unusual events, like missing paper in a printer).

As you can see, use cases are part of the developer's world, as the initial point of their object-oriented design.

But, by describing a system's behavior, use cases are also an intuitive way to document functional system requirements. Recall the dialog we introduced earlier describing what the system should do when a user interacts with it. Because use cases encompass functional requirements, they should be included in the management of *all* system requirements.

So what does it mean to manage system requirements? Requirements management is a systematic approach to eliciting, documenting, and managing requirements. Studies show projects that do not apply effective requirements management practices end up not solving user needs, delivering late, or delivering over budget. In a nutshell, effective requirements management depends on three key things:

1. *Writing 'good' requirements.* Requirements must unambiguously represent customer needs if you hope to deliver a system that provides value to your customers. The flow of events of a use case puts functional requirements in a context that customers can relate to more easily than traditional formal requirements specifications. This helps customers validate that their needs will be addressed by the system.
2. *Communicating requirements clearly to everyone on the team.* This includes the people designing the system, the ones validating the system and the ones documenting the system to end-users. They must all be on the same page in terms of what the system should deliver. The flows of events of a use case are written using natural languages. This informality, although often disturbing to logical minds, increases the understanding of what the system shall do for both the technical people building the system, and the not-so-technical people using the system. Use cases effectively address the challenge of communicating requirements to audiences of varying technical levels.
3. *Managing requirement changes.* Change is pervasive. Customers change their minds; competitors come up with better solutions before you deliver yours, the business environment changes. Being open to this changing environment is a good thing. It reflects your adaptation to evolving customer needs, but it can also bog you down. Unmanaged change occurs when the impact of the change is not adequately assessed before it drastically affects the project. This is where Integrated Use Case

Management provides most of its value to teams who adopt use cases. By managing changes in use-case documents, IUCM increases your chances of delivering a timely solution that actually solves real customer problems.

Like other requirements, use cases have *textual properties* like priority, difficulty, risk, etc. Prioritizing the importance of implementing a use case versus another use case helps you know where to focus your efforts. Use cases also have *dependencies*. 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. Changing the importance of this business need could have an effect on the use case priority. By establishing a tangible dependency (called a traceability link) between the use case and its business need, you can better pinpoint the impact of either a use-case change or a change in businesses needs.

However, unlike other requirements, use cases also have *graphical properties* (use case diagrams) that are usually specified in a modeling tool, as opposed to a requirements management tool. Managing use cases across the two domains they belong to, requirements management and design, makes use cases harder to manage than other requirements.

## ***Integrated Use Case Management Overview***

---

*Integrated Use Case Management (IUCM)* solves the challenge of managing use cases by connecting use-case diagrams stored in Rational XDE models with their respective use case requirement information (flow of events, priority, difficulty, risk, etc.) stored in Rational RequisitePro.

By extending use-cases diagrams with sortable attributes, documents, and traceability links, IUCM gives you the means to include use cases with the rest of the project requirements.

With Rational RequisitePro you can still manage requirements in Microsoft Word documents, but these documents are linked to a requirements database, where additional requirement information can be more effectively organized and managed.

Because the flow of events in a use case is most often documented in Microsoft Word, Rational RequisitePro is a perfect tool to manage use cases. Use-case flow of events are written in a RequisitePro use-case document, and the functional requirements expressed in the flow of events can be assigned a priority, a difficulty, and dependencies with their respective business drivers, just like any other requirements. This additional information is stored and managed in the RequisitePro database. From the RequisitePro views into the database, you can more easily verify use-case coverage of all selected business drivers, as well as assess the impact of change.

The value of *Integrated Use Case Management* is to seamlessly integrate uses cases with the rest of your requirements information. Including use cases in your requirements management activities is key to understanding the true state of your project and better enables you to deliver quality software.

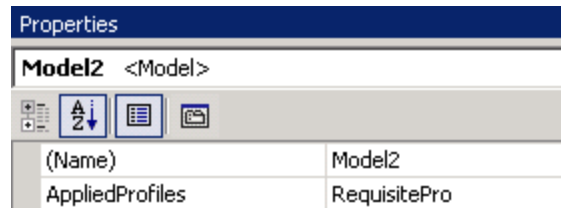
## ***Enabling Integrated Use Case Management in XDE Models***

---

Since XDE supports cross-model references, and especially if you work on large projects, we recommend you organize your design work into a set of distinct models, rather than a large monolithic model. The system use cases should be documented in their specific model and should be enabled for IUCM.

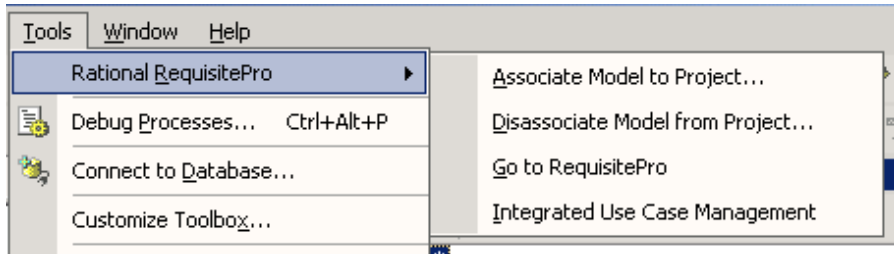
In the remainder of the article, the various steps for using IUCM are clearly marked with the prefix 'STEP'.

STEP: To enable IUCM in a Rational XDE model, include the 'RequisitePro' profile in the XDE model *AppliedProfiles* property.

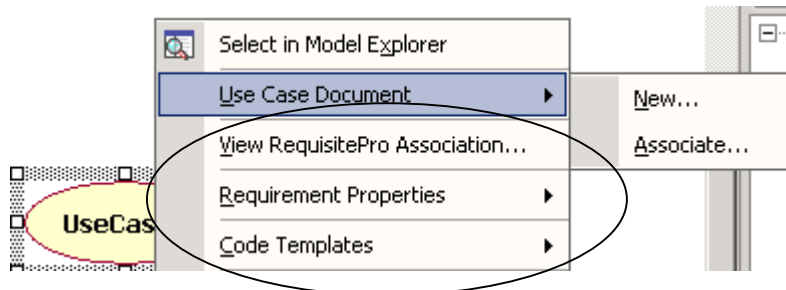


**Figure 2: Enabling IUCM in Rational XDE**

Once you enable IUCM in a Rational XDE model, every time you open this XDE model, requirement management capabilities are visible from two menus in XDE: the main menu (Figure 3), and the contextual use case right-click menu, available from the Model Explorer or from the use-case diagram (Figure 4).



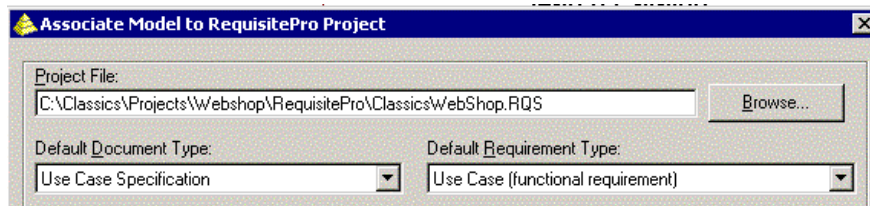
**Figure 3: Requirements options on the main Rational XDE menu**



**Figure 4: Requirements options on the use case contextual menu in Rational XDE**

Once a Rational XDE model is enabled for IUCM, the next step is to specify the Rational RequisitePro project that will manage the use-case requirement information for that model.

STEP: From the XDE *Tools* menu, select *Rational RequisitePro > Associate Model to Project* and specify a RequisitePro project.



**Figure 5: Rational RequisitePro project and use case information selection**

A Rational RequisitePro project includes document types and requirement types. Document types determine the Microsoft Word document templates you will use to document various requirements in Word, and requirements types group requirements by categories that share the same attributes (priority, risk, difficulty, status, etc.).

STEP: Select a document type to be used as use-case document template, when detailing new use cases in Rational XDE, and select a *Use Case* requirement type. Rational RequisitePro provides default project structures with which we recommend you start. As you become more familiar with the tool, you can create your own document types and requirements types.

Once you specify a Rational RequisitePro project in your Rational XDE model, you are ready to include use cases in your project requirements stored in RequisitePro.

### ***Managing Use Cases in IBM Rational XDE***

---

In Rational XDE, all requirements options are accessible from the right-click menu on a use case, either from a diagram or from the Model Explorer. These options include (see Figure 3):

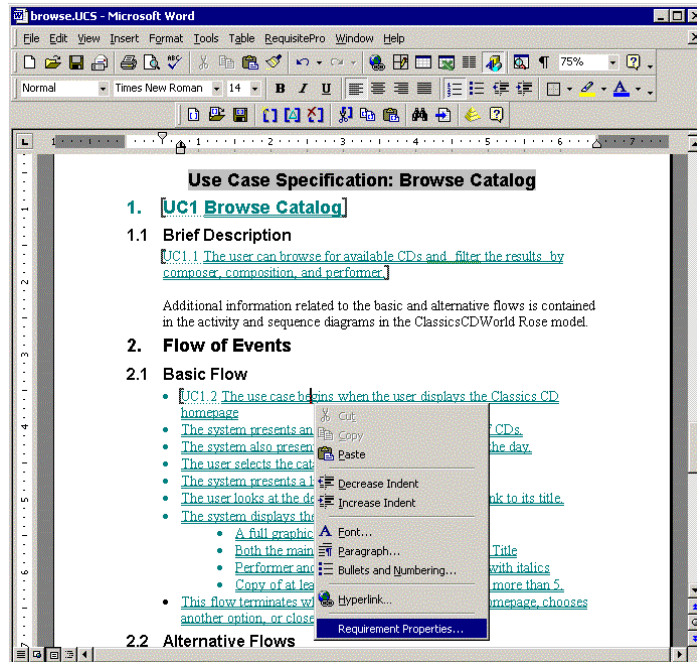
- *Use Case Document* to create a new use case document or associate the use case with an existing Rational RequisitePro use-case document
- *Requirement Properties* to view and edit attributes and traceability (dependency) links to the use case
- *View RequisitePro Association* to view the RequisitePro context for that use case (including the associated use case document template and use case attributes)

Let's look at these capabilities in more detail.

#### **Use Case Document**

IUCM goes beyond simply attaching a file to a use case in Rational XDE, which you could do by simply choosing the *Add UML > Add URL* contextual menu option and selecting any document. Using IUCM, documents attached to use cases are Rational RequisitePro documents. This provides the following advantages:

- *Use-case documents are based on a proven document template.*  
IUCM provides the use-case document template recommended in the IBM Rational Unified Process®. 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 6) making it easier to identify requirements contained throughout the document..



**Figure 6: Functional 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 relate to.*  
By tracing use cases to business requirements, feature-level requirements, or even other use cases you can more easily measure the impact of change on related requirements and verify coverage.

STEP: To associate a use-case document to a use case in Rational XDE, right-click on the use case (either from the XDE Model Explorer or from the use case diagram), and select *Use Case Document* ➔ *New*.

Microsoft Word, controlled by Rational RequisitePro, is launched and your template-based use-case document is displayed and ready for editing. You can also associate an existing RequisitePro document to a use case in XDE by using the *Use Case Document* ➔ *Associate* menu option.

## Requirements Properties

The second option in the use case contextual menu is “Requirements Properties.” Requirements, and similarly use cases, are not just text. They have additional properties, such as priority, difficulty, and dependencies to other requirements, that allow you to more effectively scope manage, organize and track requirements.

**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 on carry through to implementation and are more costly to change the further the project is in its lifecycle. Attributes provide a simple way to assign objective weight to use cases and requirements alike.



**Traceability** helps measure the impact of change and ensures requirements coverage. For instance, if business needs change, which use cases are impacted? By establishing traceability links, you can query the requirements to answer questions like “Are all business needs implemented at the use case level?”

(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. This paper, available at <http://www.rational.com/products/whitepapers/022701.jsp>, outlines various traceability approaches, to match your specific 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 allows you to concentrate on the more stable requirements first, inherently diminishing the amount of change.

STEP: To set requirements properties of a use case in Rational XDE, right-click on the use case and select *Requirement Properties* → *Open* from the contextual menu. In the dialog box (see Figure 7), click the *Attributes* tab, and set use-case attribute values. Note that you can change the out-of-the-box use case attributes and their default values in the Rational RequisitePro project associated with the XDE model. From this same dialog box, click the *Traceability* tab to establish traceability between the use case and other requirements.

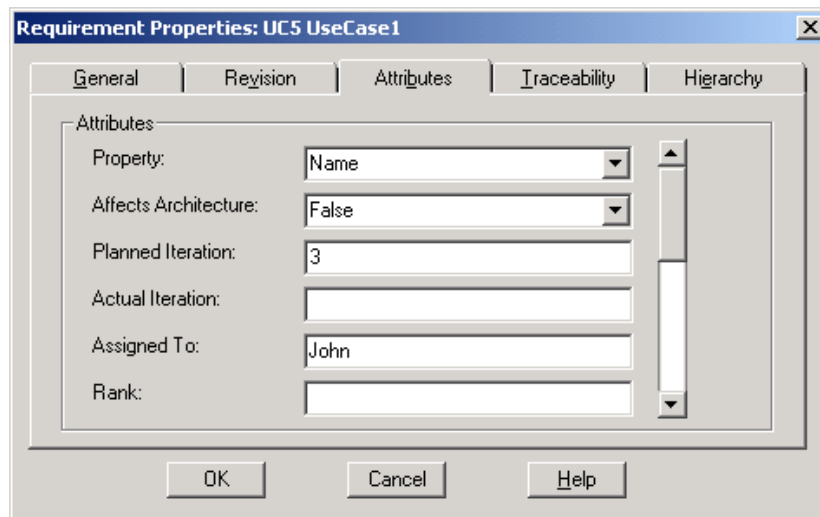


Figure 7: Viewing and editing use case requirement properties in Rational XDE

### RequisitePro Association

The third option in the use case contextual menu is “View RequisitePro Association.” This displays the document type and the requirement type that will be used to record requirement information for the selected use case. This information is displayed using the dialog on figure 5 above. Use cases stored in different packages can use different RequisitePro project to store their requirements information.

If you wish to associate some use cases with a Rational RequisitePro project, and other use cases with a different Rational RequisitePro project, you have the option of associating packages with their own RequisitePro project. All use cases in one package will be associated with the same Rational RequisitePro project, defined in the package-level association. 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 (i.e. system-level use cases and low-level use cases).

## ***Managing Use Cases in IBM Rational RequisitePro***

Once you've attached a use-case document, or assigned requirement properties to a use case in Rational XDE, the use case becomes part of your requirement set in Rational RequisitePro. As such, you can use RequisitePro's requirement management capabilities to sort your use cases (by priority, by iteration, etc.), to query on specific use cases (i.e. only the use cases planned for the next iteration), and even produce use case metrics.

An *Attribute Matrix View* in Rational RequisitePro (see Figure 8) provides a view of all, or a selected 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 I organize my 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. This answers another question presented at the beginning of the article: *How can I tell in which release a particular use case is implemented?*

Requirements:	Property	Affects Architecture	Iteration	Assigned To
UC1: Browse Catalog	Name	True	1	Terry
UC2: The use case begins when the user displays...	Basic Flow	False		
UC3: Access Product Promotion		False		
UC4: Arrange Shipment	Name	True	5	Terry
UC5: Check Order Status	Name	True	4	Terry
UC5.1: Membership Information Requirement	Brief Description	False	4	Terry
UC5.2: Choose Orders Option	Basic Flow	False	4	Terry
UC5.3: Membership Prompts	Basic Flow	False	4	Terry
UC5.4: All Customer Orders Displayed	Basic Flow	False	4	Terry
UC5.5: Specific Order detail	Basic Flow	False	4	Terry
UC5.6: Item Details	Basic Flow	False	4	Terry
UC5.7: End of Session	Basic Flow	False	4	Terry
UC5.8: Invalid Customer Information	Alternate Flow	False	4	
UC5.8.1: Invalid Customer Message	Basic Flow	False	4	
UC5.8.2: Back Navigation	Alternate Flow	False	4	
UC5.9: Valid Membership Requirement	Pre-Conditions	False	4	Terry
UC6: Checkout	Name	False	4	Terry

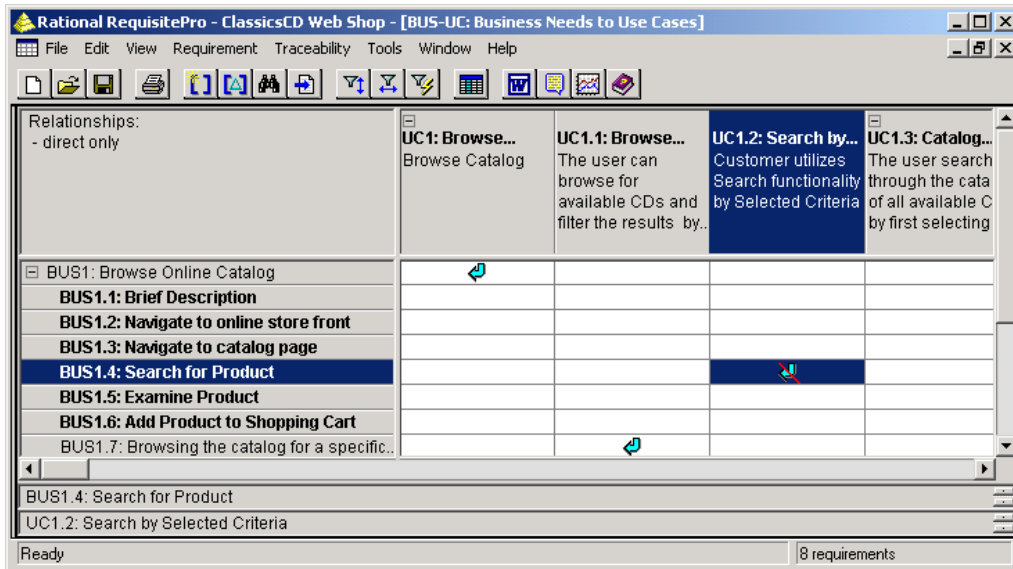
UC1.6: Pre-conditions  
This use case must be initiated from the ClassicsCD homepage.

Ready 47 requirements

**Figure 8: Use case Attribute Matrix View in Rational RequisitePro**

Once you have prioritized the use cases to be implemented in the next iteration, you should ensure that these use cases will stay in line with the evolving business needs they are intended to fulfill. The *Traceability Matrix View* in Figure 9 shows the relationships established between use cases and business needs. Using traceability matrices, you can query on business needs not yet traced to use cases, answering the question: *Which specific business needs does a use case address?* Additionally, if business needs change you know

immediately which uses cases are potentially impacted by that change, ensuring that your use cases always reflect the evolving business needs. A suspect link (red slashed arrow in Figure 9) indicates that use case UC1.2 may need to be revisited due to a change in business need BUS1.4. Querying on suspect links answers the last question presented at the beginning of the article: *Are my use cases staying in touch with the evolving business needs they are supposed to solve?*



**Figure 9: Traceability relationships between use cases and business needs**

## Summary

*Integrated Use Case Management* extends use-case diagrams with requirement information. This benefits the Rational XDE user by establishing a real-time window into modifying use-case attributes, traceability and view revision history, directly from Rational XDE. Rational RequisitePro manages use-case documents, ensuring any change to use cases is communicated to everyone on the team. *Integrated Use Case Management* provides requirements managers more accurate and timely information to work with because that information is available from the Rational XDE users' fingertips. By managing use cases in conjunction with other requirements:

- Your project is better scope managed
- Change is controlled
- Business need coverage is verified

In short, *Integrated Use Case Management* helps ensure you are implementing the functionality that was agreed upon, and that your system functionality evolves as its business drivers change.

## References

- Cockburn, Alistair, *Writing Effective Use Cases*, Addison-Wesley Publishing Co., 1<sup>st</sup> edition (October 5, 2000)
- Constantine, Larry, *Use Cases, Ten Years Later*, [www.stqemagazine.com](http://www.stqemagazine.com), March/April 2002
- Rational webinars on use cases at <http://www.rational.com/events/webinars/index.jsp>

- Online RequisitePro demonstrations at <http://www.rational.com/tryit/reqpro/seeit.jsp>
- Online Rational XDE demonstrations at <http://www.rational.com/tryit/xde/seeit.jsp>



### **IBM software integrated solutions**

IBM Rational supports a wealth of other offerings from IBM software. IBM software solutions can give you the power to achieve your priority business and IT goals.

- *DB2<sup>®</sup> software helps you leverage information with solutions for data enablement, data management, and data distribution.*
- *Lotus<sup>®</sup> software helps your staff be productive with solutions for authoring, managing, communicating, and sharing knowledge.*
- *Tivoli<sup>®</sup> software helps you manage the technology that runs your e-business infrastructure.*
- *WebSphere<sup>®</sup> software helps you extend your existing business-critical processes to the Web.*
- *Rational<sup>®</sup> software helps you improve your software development capability with tools, services, and best practices.*

### **Rational software from IBM**

Rational software from IBM helps organizations create business value by improving their software development capability. The Rational software development platform integrates software engineering best practices, tools, and services. With it, organizations thrive in an on demand world by being more responsive, resilient, and focused. Rational's standards-based, cross-platform solution helps software development teams create and extend business applications, embedded systems and software products. Ninety-eight of the Fortune 100 rely on Rational tools to build better software, faster. Additional information is available at [www.rational.com](http://www.rational.com) and [www.therationaledge.com](http://www.therationaledge.com), the monthly e-zine for the Rational community.

Rational is a wholly owned subsidiary of IBM Corp. (c) Copyright Rational Software Corporation, 2003. All rights reserved.

IBM Corporation  
Software Group  
Route 100  
Somers, NY 10589  
U.S.A.

Printed in the United States of America  
01-03 All Rights Reserved.  
Made in the U.S.A.

IBM the IBM logo, DB2, Lotus, Tivoli and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Rational, and the Rational Logo are trademarks or registered trademarks of Rational Software Corporation in the United States, other countries or both.

Microsoft and Windows NT are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a trademark of The Open Group in the United States, other countries or both.

Other company, product or service names may be trademarks or service marks of others.

The IBM home page on the Internet can be found at [ibm.com](http://ibm.com)