



---

## **RaQuest features guide**

*by SparxSystems Japan*

## **Requirements Management Tool RaQuest 3.3 Features Guide**



1.	Introduction.....	3
2.	What can RaQuest do?.....	4
3.	Main Window.....	5
4.	Creating New Requirements.....	7
5.	Trees.....	21
6.	Requirements List.....	24
7.	Relationships between Requirements.....	32
8.	Matrix.....	36
9.	How to use Baseline.....	40
10.	Management of Changes.....	42
11.	Documenting and Utilization of Data.....	45
12.	Importing the External Requirements Information.....	50
13.	Notice of Changes.....	51
14.	Integration of Enterprise Architect and RaQuest.....	53

## **1. Introduction**

Welcome to SparxSystems Japan RaQuest Features Guide. This guide is intended to describe various features of RaQuest. It illustrates useful features of RaQuest accompanied with some examples.

To use RaQuest, you are required to have a license of Enterprise Architect 6.0 or later version, and install it in advance. Unregistered users or users who have not renewed the maintenance agreement at the end of previous maintenance period cannot use RaQuest. To use Floating version, you are required to have Enterprise Architect 6.1 version build 788 or later. For license manager, version 1.3 or later is needed.

The latest free trial version of Enterprise Architect and RaQuest will give you an opportunity to discover their power and benefits.

## 2. What can RaQuest do?



fig. 1

RaQuest is a requirements management tool. You can input summary and detailed attribute of a requirement. RaQuest enables you to list up requirements and print them out, or export them as HTML or Word documents. Moreover, you can also enter relationship between requirements. RaQuest can display relationship or matrix useful for analysis of impact or coverage from the relationship you entered. Herewith you can manage requirements more efficiently.

Though requirements management originally has nothing to do with UML, RaQuest is integrated with UML modeling tool Enterprise Architect. Therefore RaQuest can create Use Cases of UML from requirements or load them as requirements. It is also possible to relate requirements UML items. The features give you close and effective teamwork between requirements and UML diagrams, such as a Use Case diagram, or design elements. In this way, RaQuest has a functionality to transform requirements into UML smoothly.

Normally, the requirements are managed by a specific staff or a part of the team, because requirements management tool is too costly to distribute to all members of the team. RaQuest is designed to change such a development style. RaQuest is low-cost enough to be distributed to all members, so it can give you a totally new development style that all members related to the development can share the information and confirm them as needed. This development style is exactly what we aim for.

RaQuest is a coined word from “Requirements Adjustment Quest” with hope for tool to be able to do the requirements management easily (The pronunciation of RaQ means easy or happy in Japanese.)

### 3. Main Window

Run RaQuest, and open the project file. The following window appears (fig. 2). This is main window of RaQuest. A sample project file is used for fig. 2. The actual window and the contents displayed onto your computer would be different from fig. 2, depending on the build or the configurations of RaQuest.

If you have project files of UML modeling tool Enterprise Architect, you can also open them on RaQuest.

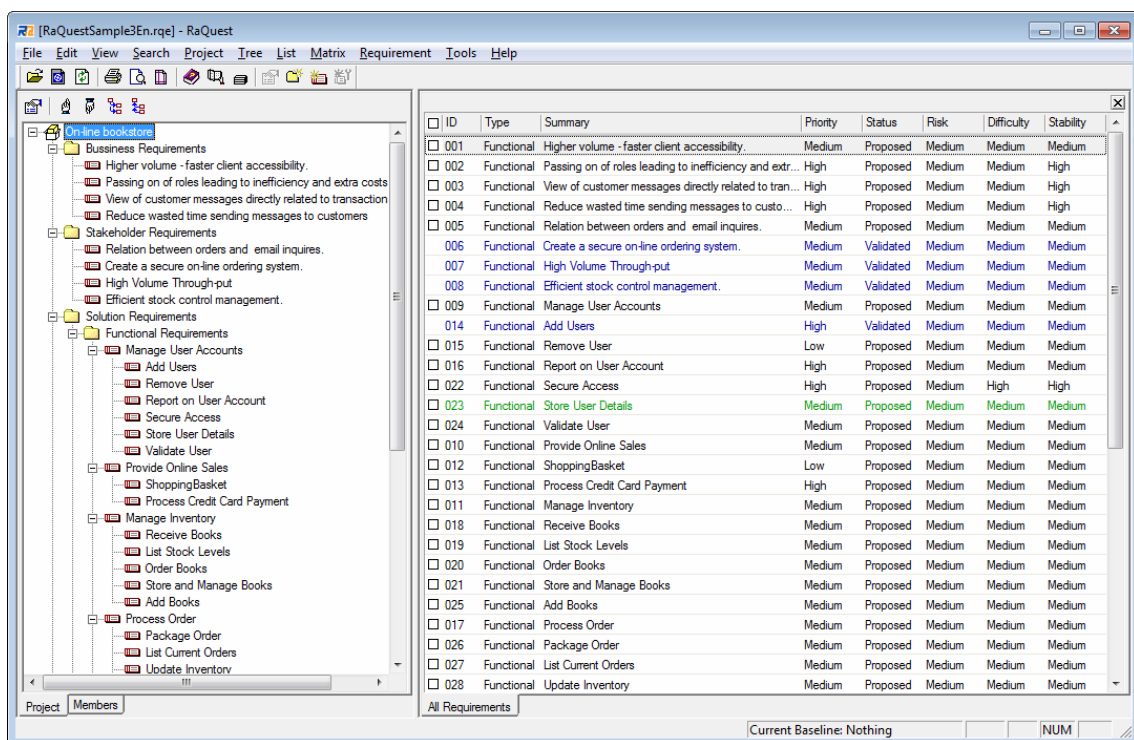


fig. 2

In the left pane of the main window, a project tree to the requirements and packages is displayed. In the right pane of the main window, requirement list is displayed.

When you check **Always Display Requirement** option, the main window is displayed as shown in fig. 3. The requirement selected in tree or list is always displayed in the lower part of the right pane of main window.

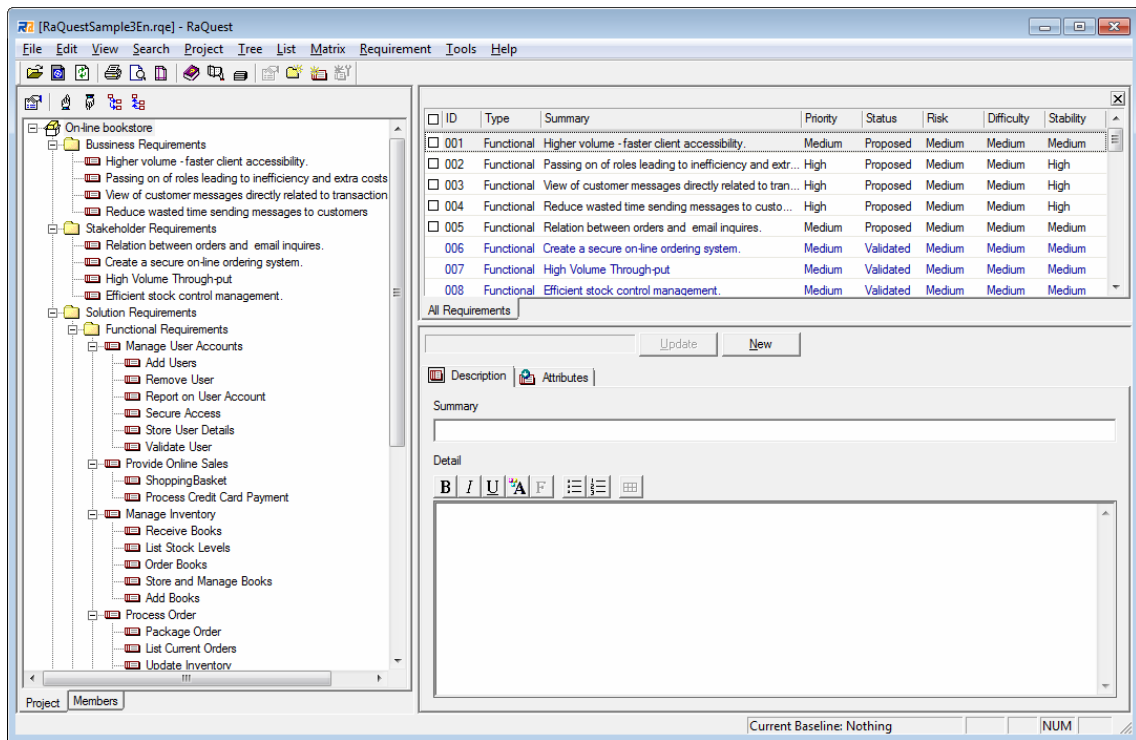


fig. 3

## 4. Creating New Requirements

This section describes how to create a new requirement and configure its option settings.

To create a new requirement, select **Requirement | New Requirement** menu item (fig. 4), or right-click a package or a requirement under where you would like to create the new requirement and select **New Requirement** (fig. 5). In addition, double-clicking the blank field of **Requirements** list is also available.

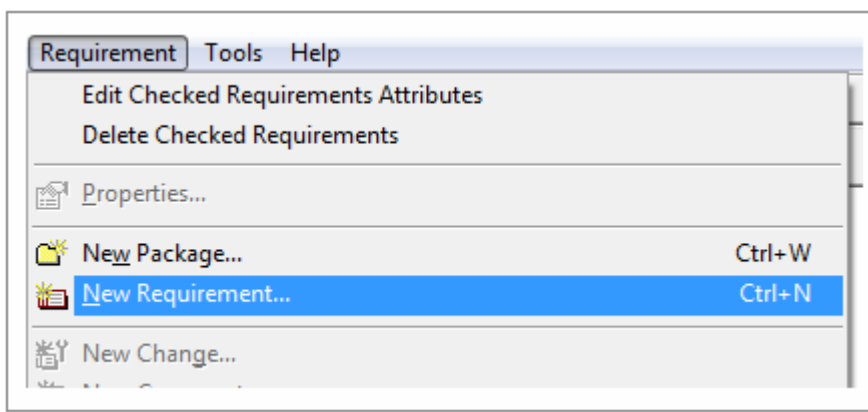


fig. 4

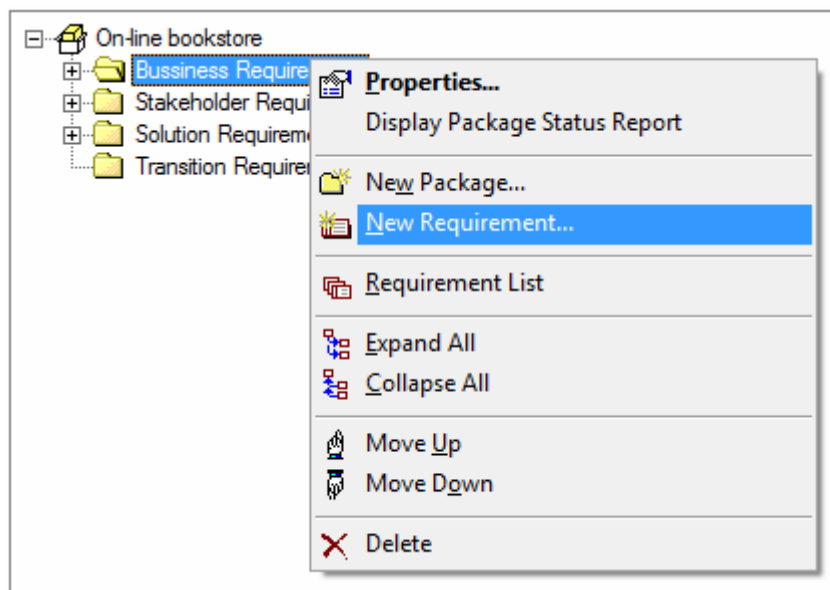


fig. 5

The following **New Requirement Properties** dialogs (fig. 6, fig. 7) allows you to set its summary, details, requirement type, status, limit, difficulty and priority.

You can also set **User Define Attributes** when you define it in advance (fig. 8).

When you check **Display Attributes by List Form** option or **Display Attribute by list form on this Project** option, you can set them in **Description** (fig. 9) tab and **Attributes** (fig. 10) tab as shown in and.

The screenshot shows the 'New Requirement' dialog box with the following fields and options:

- Summary: [Text Area]
- ID: [001] Revision: [1]
- Version: [1.0] Phase: [1.0]
- Due Date: [12/ 7/2010] Type: [Functional]
- Created: [ ] Status: [Proposed]
- Created by: [ssj]  Lock Requirement
- Updated: [ ]  Review Required
- Updated by: [ ]  Approved
- Approved by: [ ]
- Keep the dialog open when OK button is clicked
- Buttons: Original Requirement, Up, Down, OK, Cancel

fig. 6

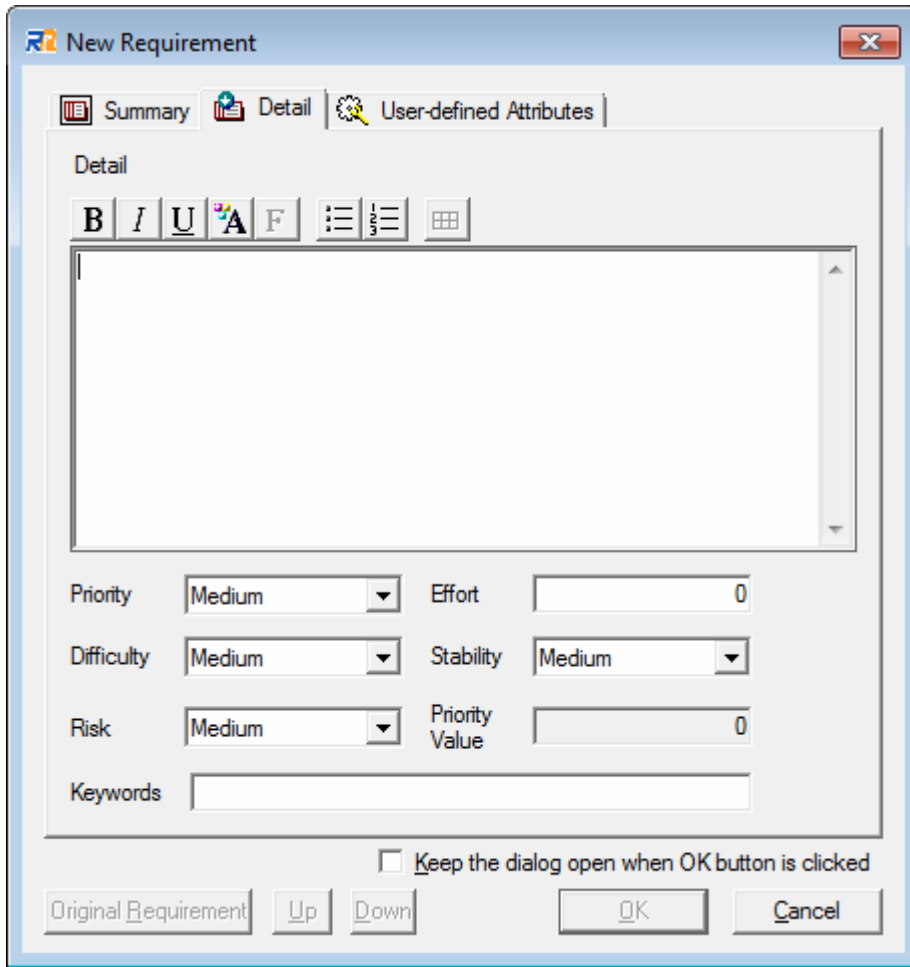


fig. 7

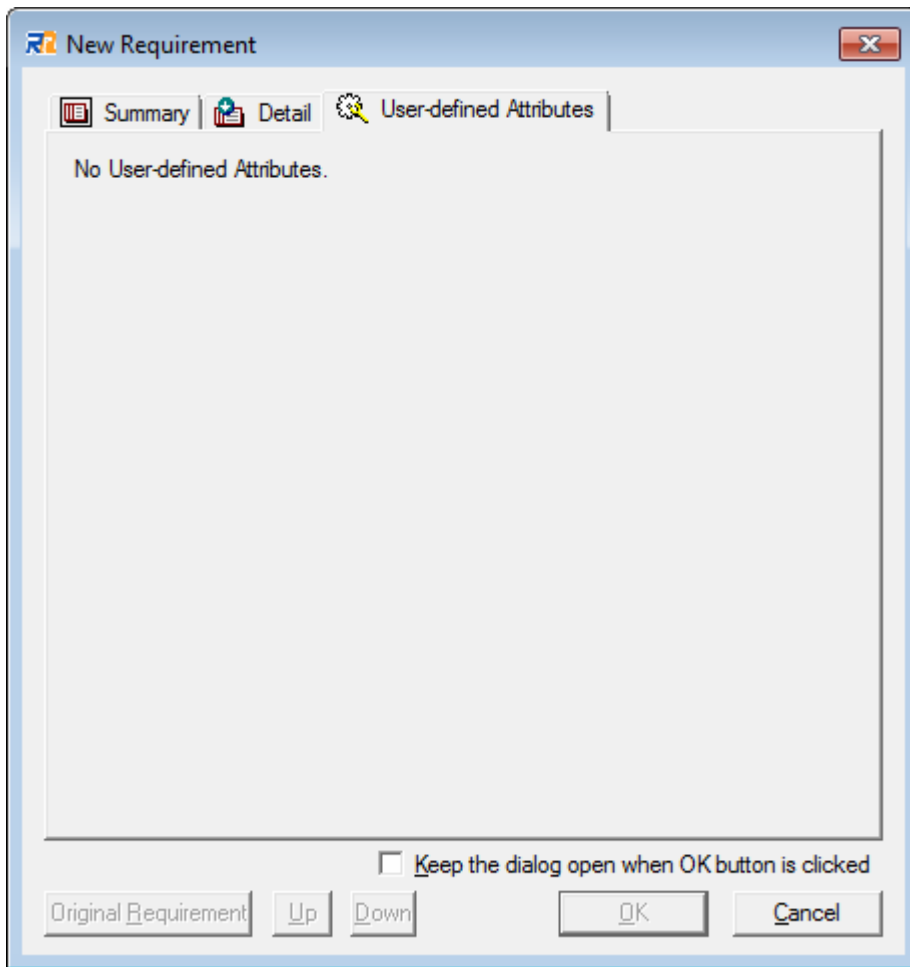


fig. 8

**User-Defined Attributes** tab is available, only when user-defined attributes are set in the project file.

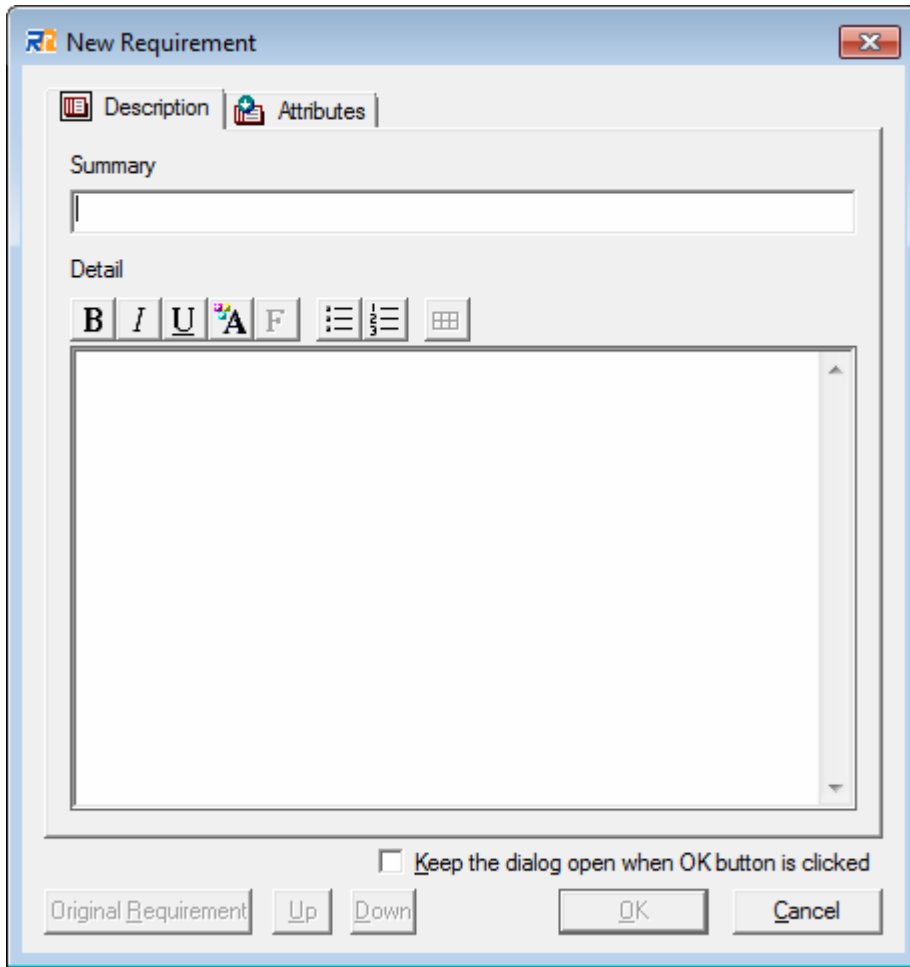


fig. 9

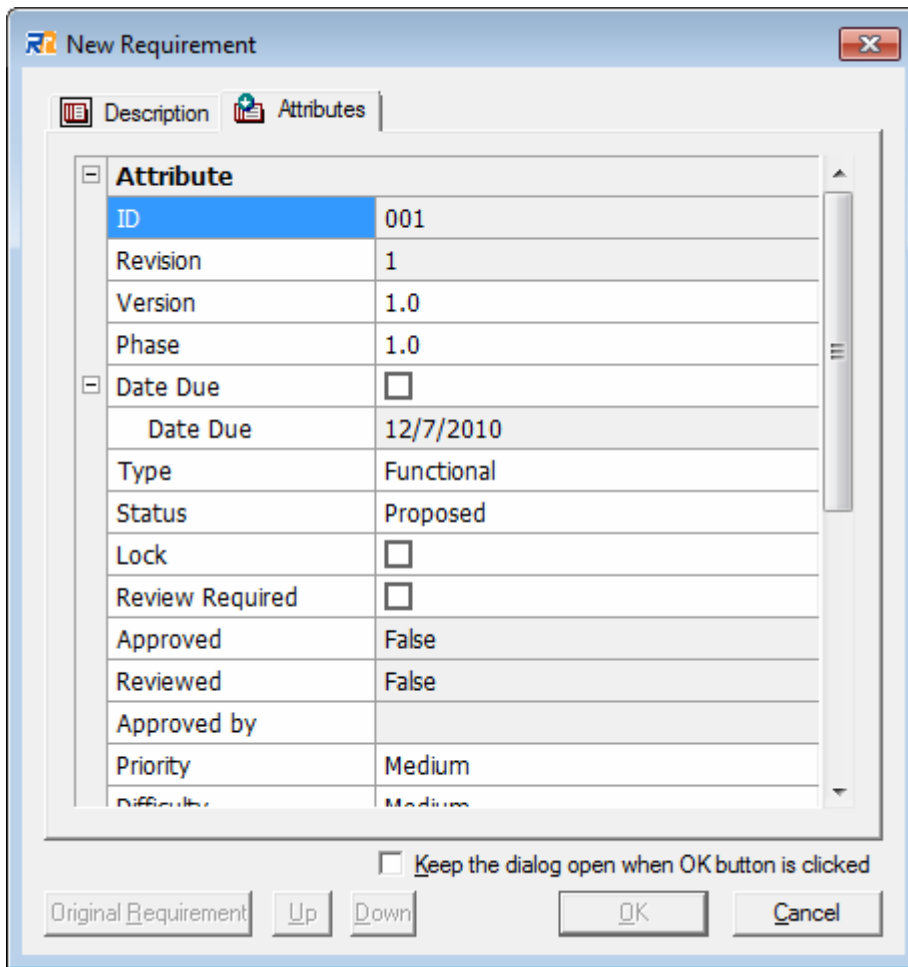


fig. 10

After creating a new requirement, you can set the details about the requirement (fig. 11). **Requirement Properties** dialog has **UML Items**, **Files**, **Members**, **Test Items**, **Comments**, **Review Requirements** and **Update Log** tabs. You can refer to them and set the details of each requirement in this dialog. Note that the dialog for new requirements does not have these tabs. You can set these tabs in the properties dialog only after creating the requirement.

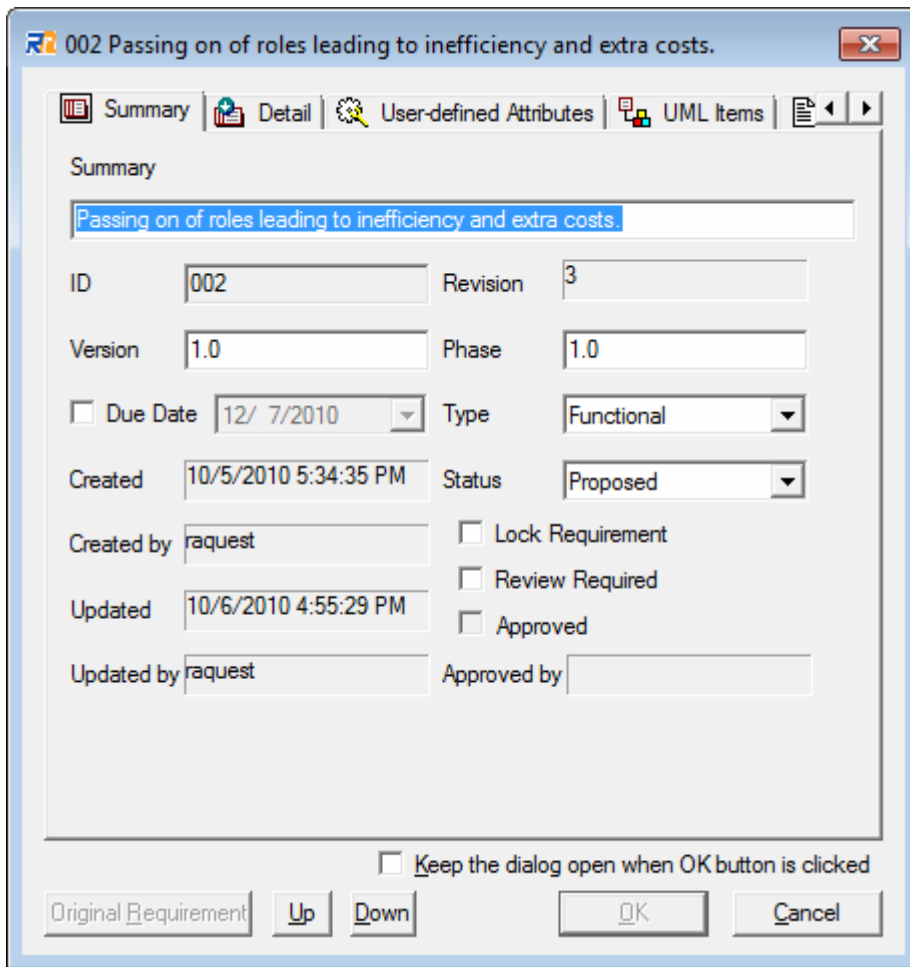


fig. 11

At the **UML Items** tab, as shown in fig. 12, you can register or refer to the relationships between requirements and UML items.

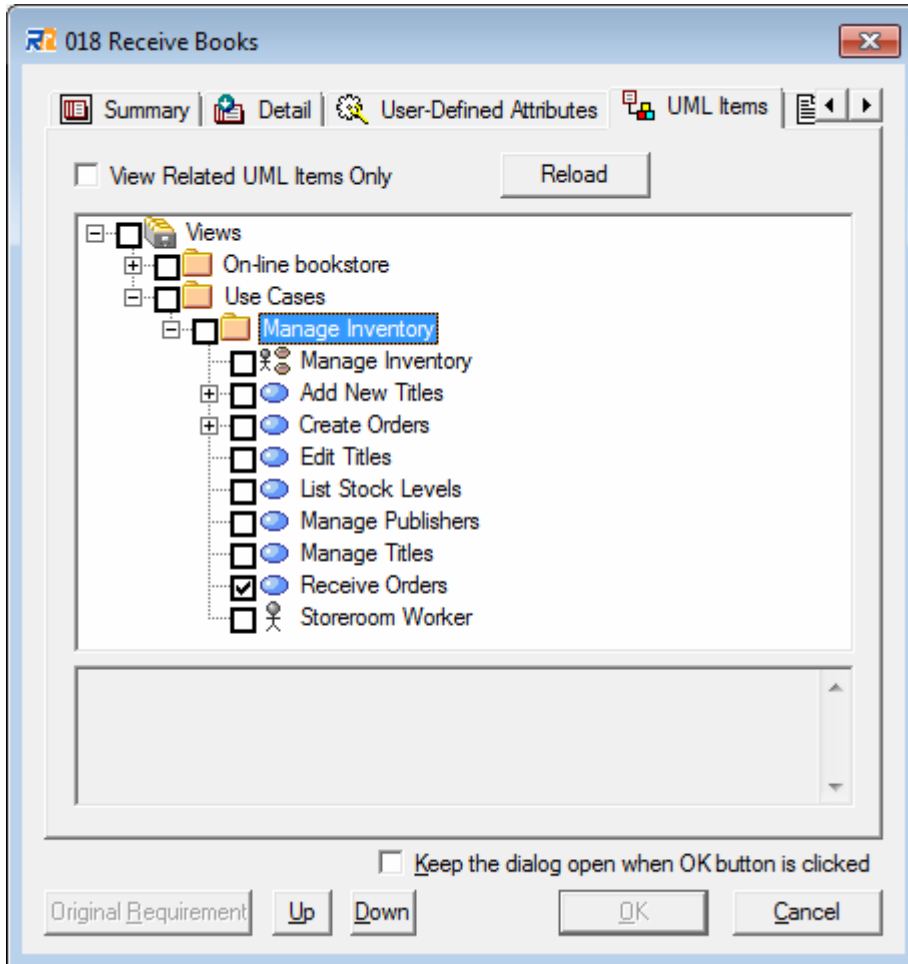


fig. 12

In the tree you can register relationships by checking the checkbox of UML items. Here you can also enter comments related to it.

You can refer to the relationships to UML item related here in matrix, list and the diagrams in Enterprise Architect. It is useful for the confirmation of the impact, and for checking the deliverable of the requirement.

At **Files** tab as shown in fig. 13, you can register files and URLs which are related to the requirements. RaQuest stores only the links, not the actual files or URLs or UML diagrams.

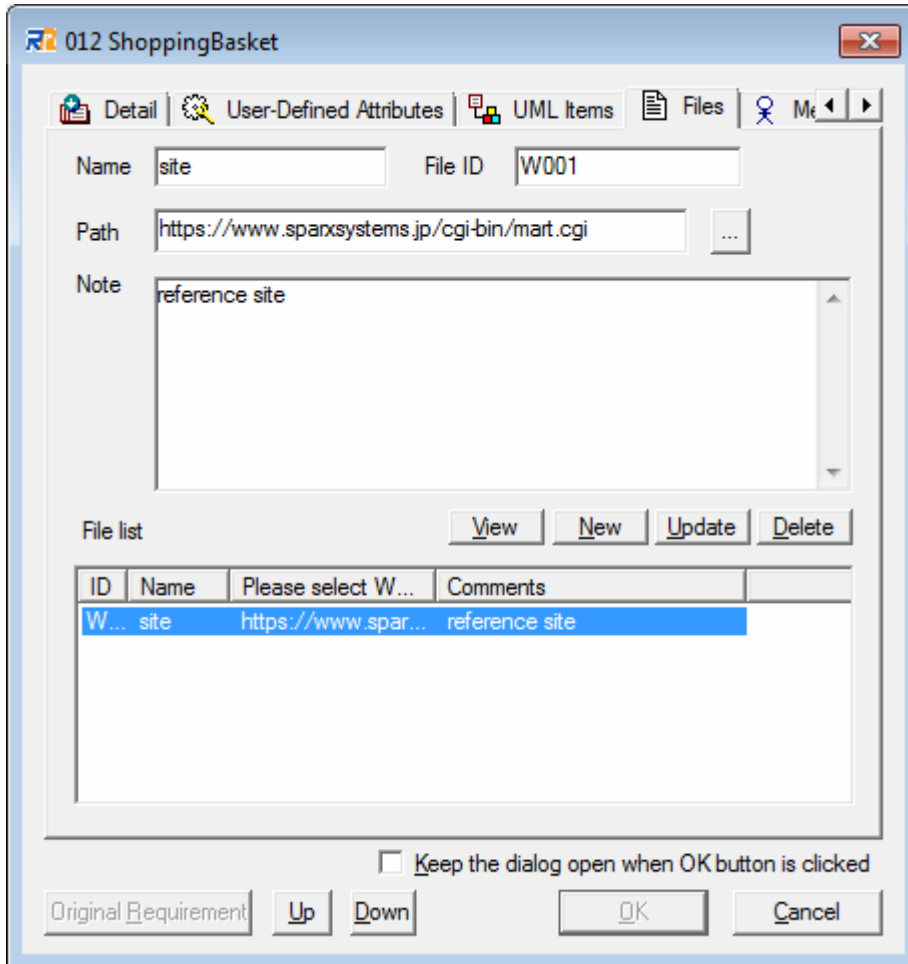


fig. 13

You can select a related file from **File List** to edit them. **View** button is also available to display the file. When a URL is selected, your default web browser starts automatically. When a file is selected, the application associated with the extension of the file starts automatically.

You can copy the files, which are registered here, to the specified directory by using **Collect Related Files** feature. This feature is also useful for bringing out the data, combining the related files and HTML files with the requirement items. (RaQuest generates Windows URL shortcut files, if URLs are registered.)

At **Members** tab, you can check and edit (add, modify or delete) members assigned to the requirement (fig. 14).

001 Higher volume - faster client accessibility.

User-Defined Attributes | UML Items | Files | Members

Member: Chief Executive Officer | Time: 0

Role: Project Manager | % Complete: 0

Start: 10/13/2010 | Finish: 10/13/2010

Descr: [Empty text area]

Assigned Members: [New] [Update] [Delete]

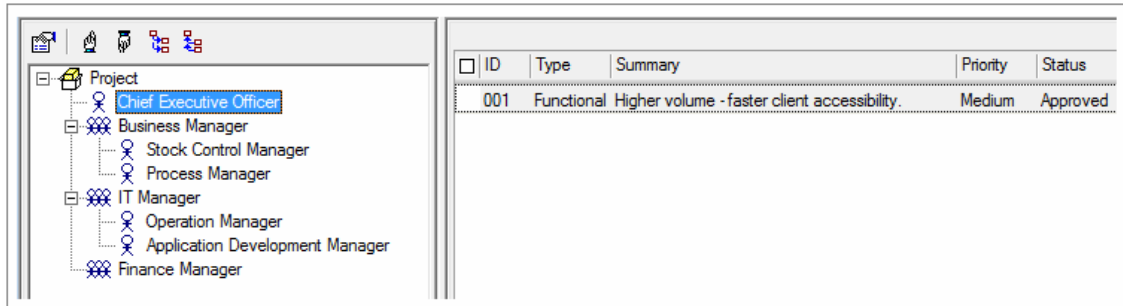
Assigned member	Role
Chief Executive Officer	Project Manager

Keep the dialog open when OK button is clicked

[Original Requirement] [Up] [Down] [OK] [Cancel]

fig. 14

**Members** tree is also available to assign members to the requirement (fig. 15).



**fig. 15**

The easiest way to assign a member to a requirement is to select a member in **Members** tree and drag and drop it to the requirement. The reverse drag-and-drop operation from a requirement to a member is effective as well. While you drag a requirement, the shape of the icon is changed. The result of your operation is displayed on the status bar.

You can also assign divisions to requirements in the same way (**Use Division as Member** option.)

At **Comments** tab, you can enter comments about the requirement (fig. 16).

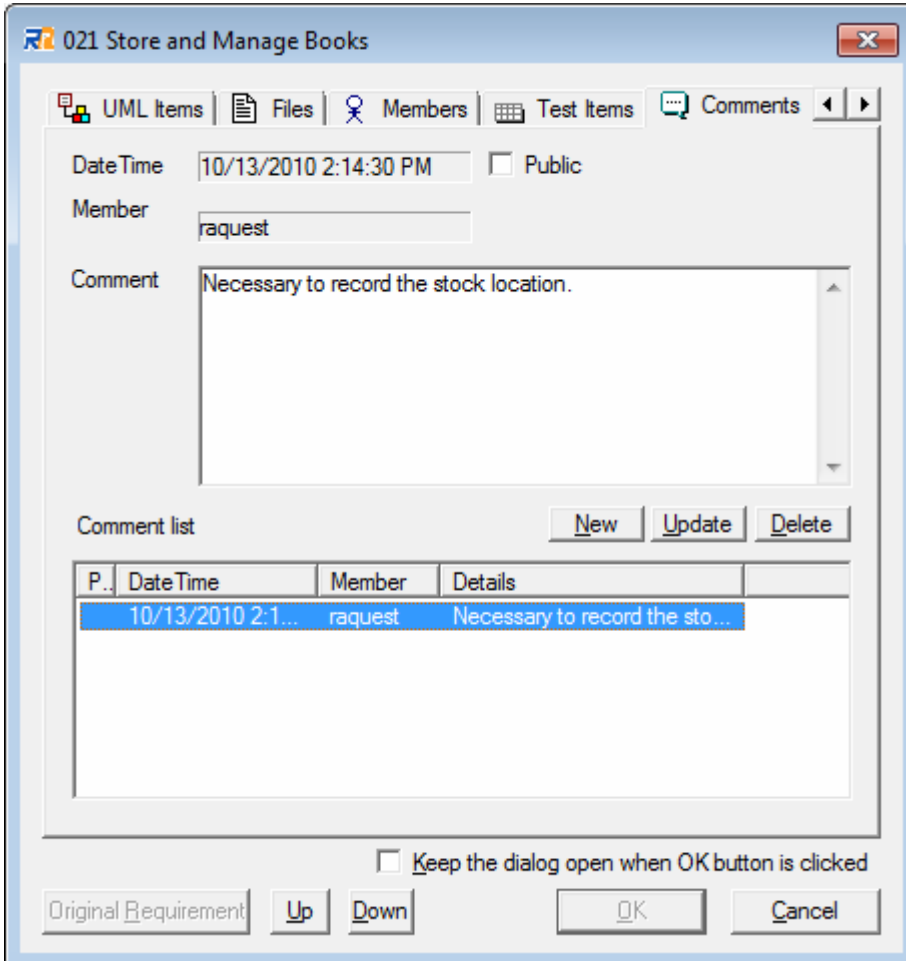


fig. 16

**Public** checkbox controls the publication. When the checkbox is checked, the comments are described in exported documents. When it is unchecked, the comments are not described in exported documents. This feature is useful for considering the comments within your development team. (The window is resizable. Above fig. 16 is resized for a better viewing.)

When **Review Required** checkbox is checked in **Summary** tab, you can confirm the source requirements in **Source Requirement List for Review** in **Review Requirements** tab (fig. 17).

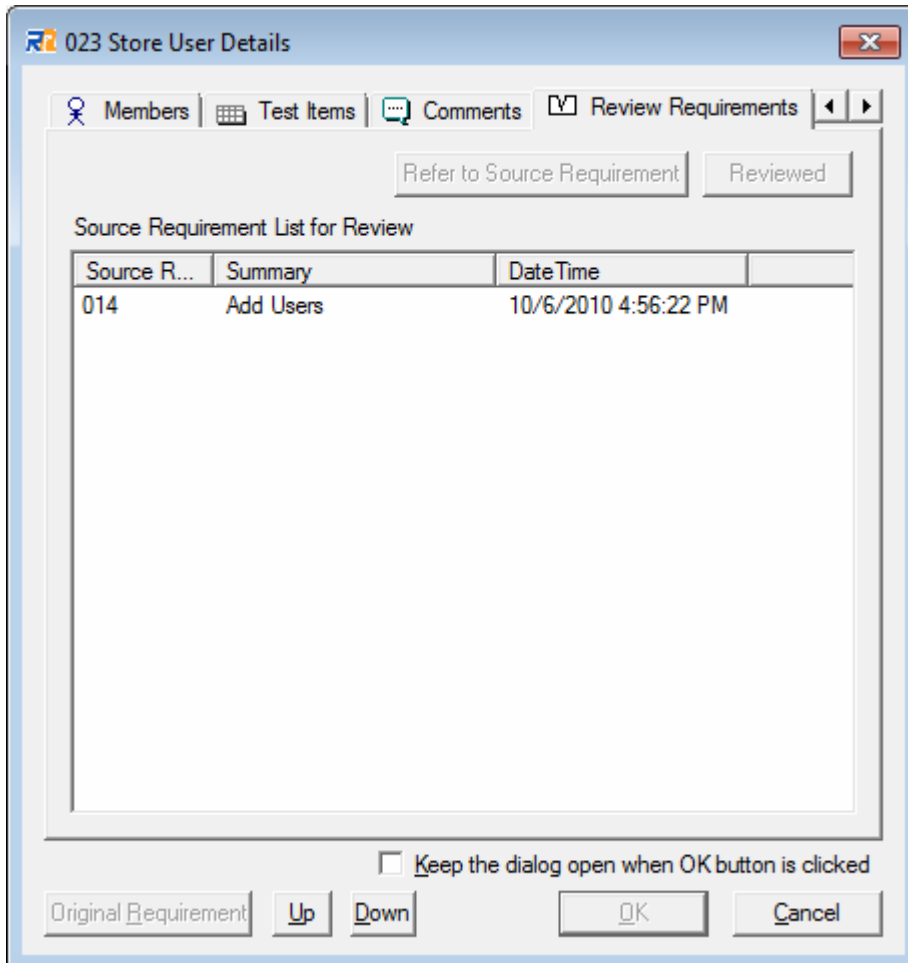


fig. 17

To refer the properties of the source requirements, select the requirement and click **Refer to Source Requirement** button. After the review, click **Reviewed** button. This completes the review of the source requirement. When there is no need to review the requirements, click on the **Reviewed** button to make its status reviewed. When all the requirements are gone from the list, the **Review Required** checkbox in the **Summary** tab will be unchecked.

**Update Log** tab allows you to refer to the update log for the current requirement (fig. 18).

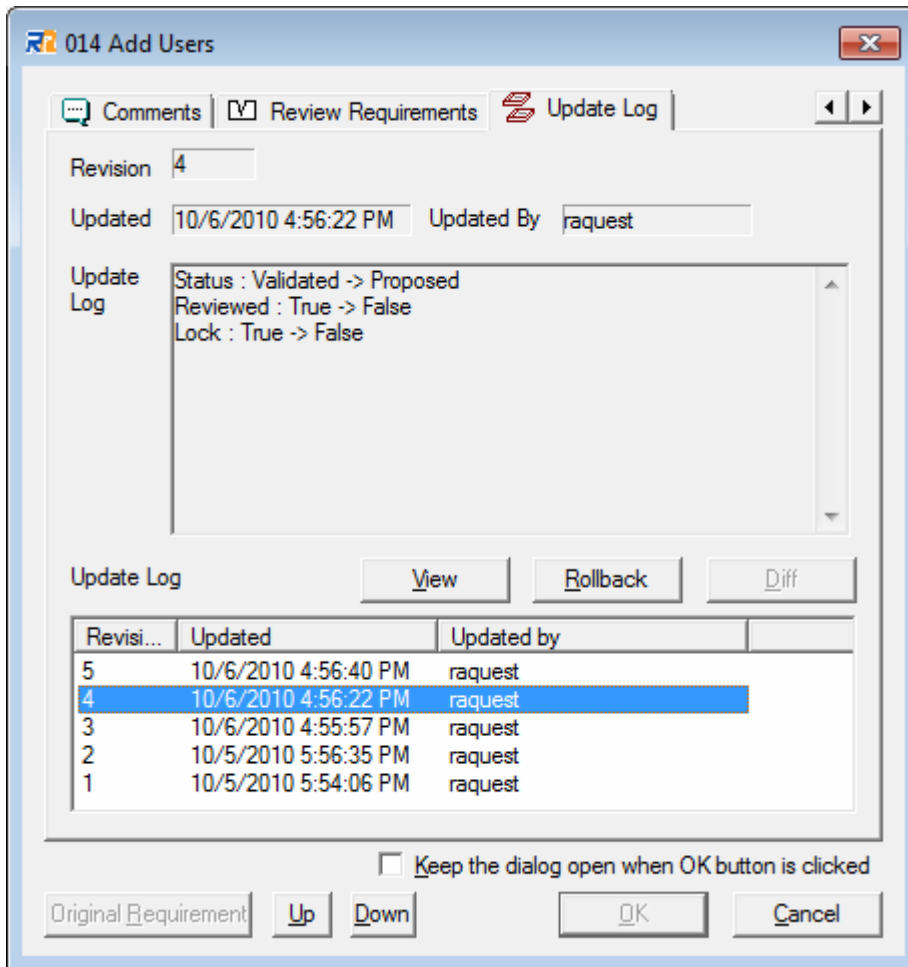


fig. 18

Update log is saved for each revision. Selecting the appropriate revision displays the update details at upper part of the tab. **View** button displays the requirement dialog in the status of the selected revision. **Rollback** button restores the requirement to the condition of the selected revision.

## 5. Trees

Trees are useful in categorizing and managing a lot of requirements efficiently.

Trees can have packages which are designed to store requirements for the easy categorization and management (fig. 19).

You can move packages and requirements as you need by drag and drop.

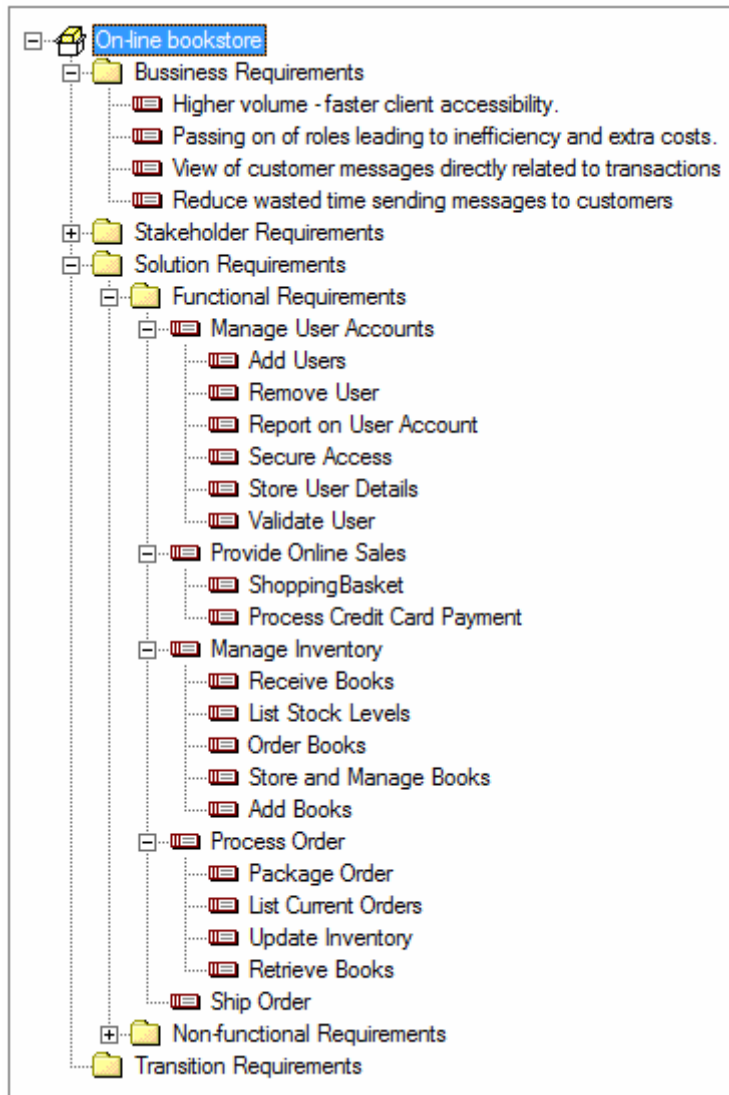


fig. 19

A requirement can have child requirements to define the details or the segmentation (fig. 20).

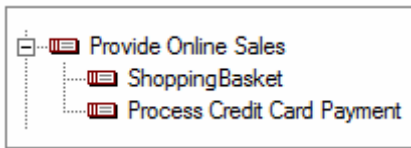


fig. 20

A requirement can also have packages (fig. 21). The packages under the requirement work as the index of the child requirements, so they play different role from usual packages.

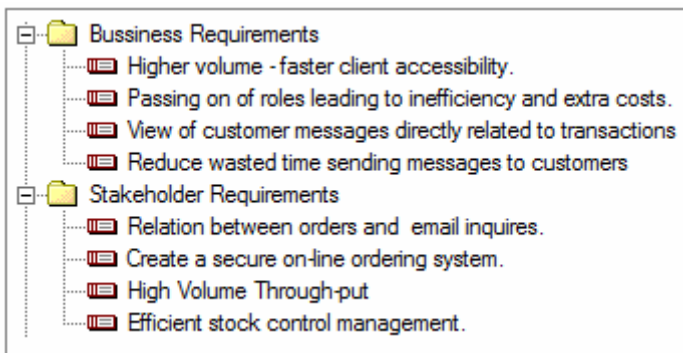


fig. 21

It is important to manage a lot of requirements efficiently as above.

At the left pane tree is displayed in a tab style. In this tab, project and members are displayed at the start-up. **Project** tree, as described above, shows whole packages and requirements of the project. **Members** tree shows divisions and members as shown in fig. 22.

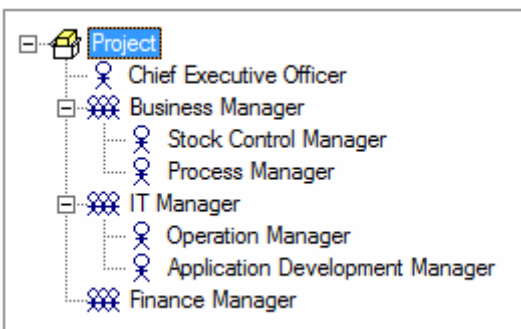


fig. 22

As for the details about **Members** tree, refer to the next section.

You can use the other trees like **by Type**, **by Status** and customizable tree (fig. 23). Regarding the custom tree, you can specify the condition as needed.

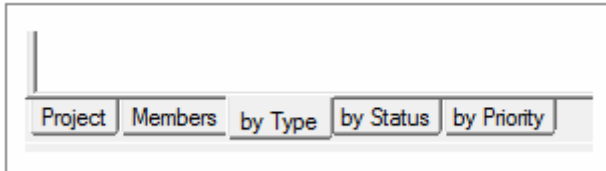


fig. 23

Up to 10 custom trees can be registered in **Tree** menu (fig. 24).

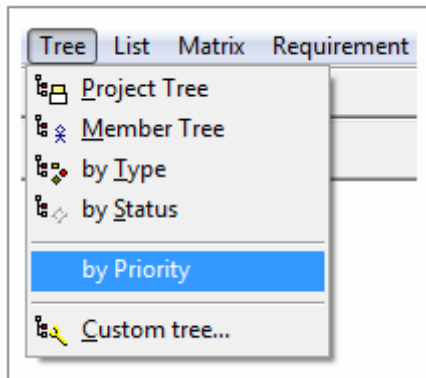


fig. 24

## 6. Requirements List

**Requirements** list is one of the most effective tools to understand requirements' outline. Requirements in the list are displayed with the different colors based on each status (fig. 25).

<input type="checkbox"/> ID	Type	Summary	Priority	Status	Risk	Difficulty	Stability
<input type="checkbox"/> 001	Functional	Higher volume - faster client accessibility.	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 002	Functional	Passing on of roles leading to inefficiency and extra costs.	High	Proposed	Medium	Medium	High
<input type="checkbox"/> 003	Functional	View of customer messages directly related to transactions	High	Proposed	Medium	Medium	High
<input type="checkbox"/> 004	Functional	Reduce wasted time sending messages to customers	High	Proposed	Medium	Medium	High
<input type="checkbox"/> 005	Functional	Relation between orders and email inquires.	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 006	Functional	Create a secure on-line ordering system.	Medium	Validated	Medium	Medium	Medium
<input type="checkbox"/> 007	Functional	High Volume Through-put	Medium	Validated	Medium	Medium	Medium
<input type="checkbox"/> 008	Functional	Efficient stock control management.	Medium	Validated	Medium	Medium	Medium
<input type="checkbox"/> 009	Functional	Manage User Accounts	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 014	Functional	Add Users	High	Validated	Medium	Medium	Medium
<input type="checkbox"/> 015	Functional	Remove User	Low	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 016	Functional	Report on User Account	High	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 022	Functional	Secure Access	High	Proposed	Medium	High	High
<input type="checkbox"/> 023	Functional	Store User Details	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 024	Functional	Validate User	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 010	Functional	Provide Online Sales	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 012	Functional	ShoppingBasket	Low	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 013	Functional	Process Credit Card Payment	High	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 011	Functional	Manage Inventory	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 018	Functional	Receive Books	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 019	Functional	List Stock Levels	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 020	Functional	Order Books	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 021	Functional	Store and Manage Books	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 025	Functional	Add Books	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 017	Functional	Process Order	Medium	Proposed	Medium	Medium	Medium

fig. 25

You can sort requirements in the list by clicking the headers of the list (**ID**, **Summary**, **Status**, **Effort**, etc). The combinations of headers are also possible. For example, if you click **Status** header first and then click **Priority** header, requirements are sorted as shown in.

<input type="checkbox"/>	ID	Type	Summary	Priority / Status	Risk	Difficulty	Stability
<input type="checkbox"/>	002	Functional	Passing on of roles leading to inefficiency and extra costs.	High Proposed	Medium	Medium	High
<input type="checkbox"/>	003	Functional	View of customer messages directly related to transactions	High Proposed	Medium	Medium	High
<input type="checkbox"/>	004	Functional	Reduce wasted time sending messages to customers	High Proposed	Medium	Medium	High
<input type="checkbox"/>	014	Functional	Add Users	High Validated	Medium	Medium	Medium
<input type="checkbox"/>	016	Functional	Report on User Account	High Proposed	Medium	Medium	Medium
<input type="checkbox"/>	022	Functional	Secure Access	High Proposed	Medium	High	High
<input type="checkbox"/>	013	Functional	Process Credit Card Payment	High Proposed	Medium	Medium	Medium
<input type="checkbox"/>	001	Functional	Higher volume - faster client accessibility.	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	005	Functional	Relation between orders and email inquires.	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	006	Functional	Create a secure on-line ordering system.	Medium Validated	Medium	Medium	Medium
<input type="checkbox"/>	007	Functional	High Volume Through-put	Medium Validated	Medium	Medium	Medium
<input type="checkbox"/>	008	Functional	Efficient stock control management.	Medium Validated	Medium	Medium	Medium
<input type="checkbox"/>	009	Functional	Manage User Accounts	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	023	Functional	Store User Details	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	024	Functional	Validate User	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	010	Functional	Provide Online Sales	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	011	Functional	Manage Inventory	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	018	Functional	Receive Books	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	019	Functional	List Stock Levels	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	020	Functional	Order Books	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	021	Functional	Store and Manage Books	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	025	Functional	Add Books	Medium Proposed	Medium	Medium	Medium
<input type="checkbox"/>	017	Functional	Process Order	Medium Proposed	Medium	Medium	Medium

fig. 26

You can see that the requirements are sorted first by status, and then, for a given condition, by priority. To narrow down requirements, filter feature is available. For example, to display requirements with a keyword “User” in their summary, set as follows in the **Filter** dialog (fig. 27).

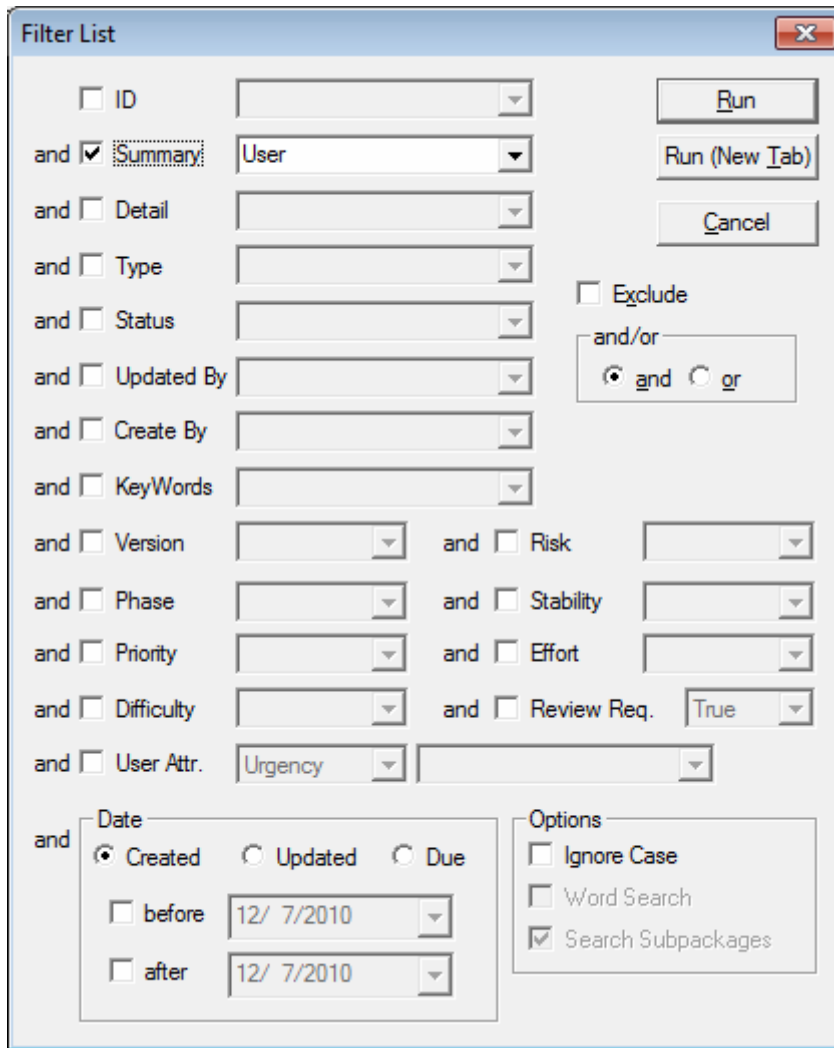


fig. 27

After clicking on the **Run** button in the **Filter** dialog, requirements with a keyword “term” in their summary are displayed as shown in fig. 28.

<input type="checkbox"/> ID	Type	Summary	Priority	Status	Risk	Difficulty	Stability
<input type="checkbox"/> 009	Functional	Manage User Accounts	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 014	Functional	Add Users	High	Validated	Medium	Medium	Medium
<input type="checkbox"/> 015	Functional	Remove User	Low	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 016	Functional	Report on User Account	High	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 023	Functional	Store User Details	Medium	Proposed	Medium	Medium	Medium
<input type="checkbox"/> 024	Functional	Validate User	Medium	Proposed	Medium	Medium	Medium

fig. 28

In this way, it is easy to check a lot of requirements in the **Requirements** list.

In addition to the filter feature above, the detail search feature is also available (fig. 29).

The screenshot shows a 'Search' dialog box with the following elements:

- ID
- and  Summary: User
- and  Detail
- and  Type
- and  Status
- and  Updated By
- and  Create By
- and  KeyWords
- and  Version
- and  Risk
- and  Phase
- and  Stability
- and  Priority
- and  Effort
- and  Difficulty
- and  Review Req.: True
- and  User Attr.: Urgency
- and  Created  Updated  Due
- before: 12/ 7/2010
- after: 12/ 7/2010
- Exclude
- and/or:  and  or
- Direction:  Up  Down
- Options:  Ignore Case,  Word Search,  Search Subpackages
- Buttons: Run, Cancel

fig. 29

By putting checks in the checkboxes at list, you can make changes to the attributes of multiple requirements or make relationships with other requirements all at once.

Also, the list can be printed out. The order of printed requirements is the same as that of the **Requirements** list on screen. Therefore sort and filter the requirements as desired before printing. The following is a preview for **Requirements** list (fig. 30).

ID	Type	Summary	Priority	Status	Risk	Difficulty	Stability
014	Functional	Add Users	High	Validated	Medium	Medium	Medium
015	Functional	Remove User	Low	Proposed	Medium	Medium	Medium
016	Functional	Report on User Account	High	Proposed	Medium	Medium	Medium
023	Functional	Store User Details	Medium	Proposed	Medium	Medium	Medium
024	Functional	Validate User	Medium	Proposed	Medium	Medium	Medium

fig. 30

When a project file is opened, all requirements are displayed. To display only the requirements under a package, first select the package in **Project** tree and then select **List | Requirements List** menu item (fig. 31). Only requirements under the specified package are displayed in **Requirements** list (fig. 32).

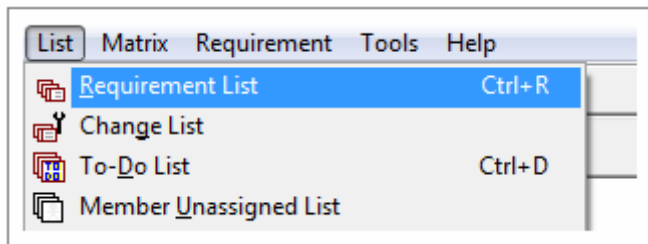


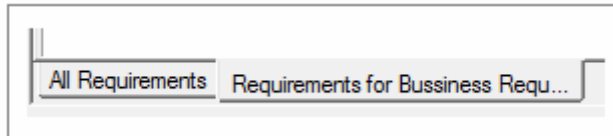
fig. 31

ID	Type	Summary	Priority	Status	Risk	Difficulty	Stability
001	Functional	Higher volume - faster client accessibility	Medium	Proposed	Medium	Medium	Medium
002	Functional	Passing on of roles leading to inefficiency and e...	High	Proposed	Medium	Medium	High
003	Functional	View of customer messages directly related to tr...	High	Proposed	Medium	Medium	High
004	Functional	Reduce wasted time sending messages to cust...	High	Proposed	Medium	Medium	High

fig. 32

The **Requirements** lists are saved automatically after the display, and tabs for the **Requirements** lists are added at the bottom of **Requirement** lists. You can switch **Requirements** lists as needed,

using the tabs (fig. 32).



**fig. 33**

Not only requirements but also comments and files can be displayed in the lists as well. For example, to check the comments of the requirements in a list, first select the package in **Project** tree, and then select **List | Comments List** menu item. Select a member in the **Member** tab, you can also check a list of requirements to which a member is assigned by using **Member Requirements** feature. RaQuest has a lot of features for listing up requirement items, which enable you to manage requirements efficiently.

You can freely customize the items displayed in **Requirements** list as shown in fig. 34.

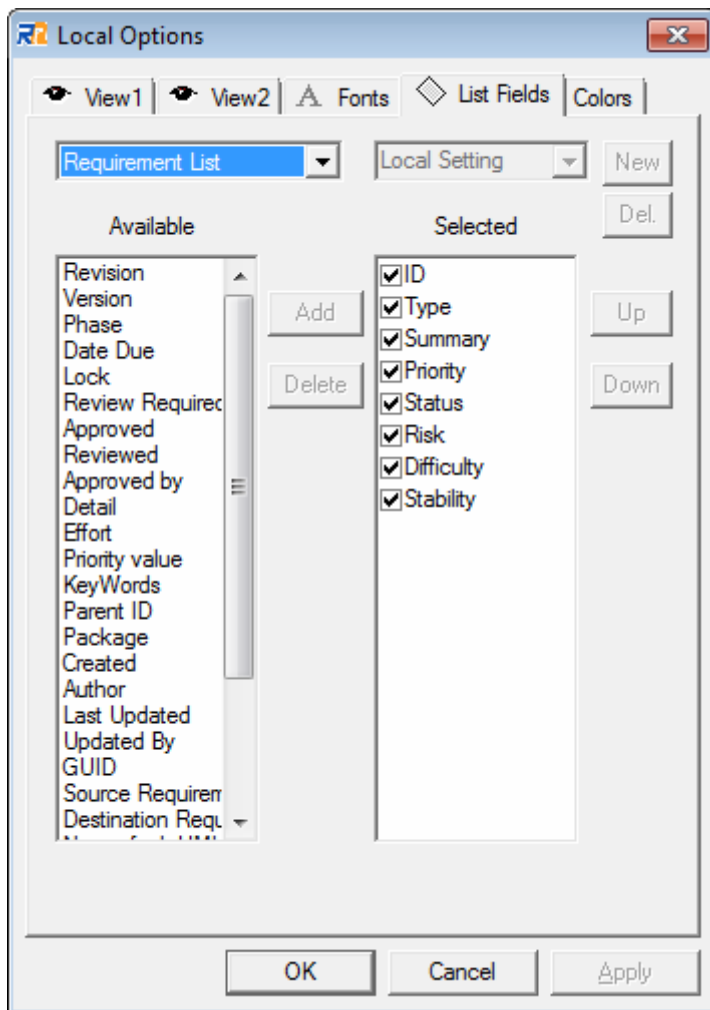


fig. 34

You can also freely change colors of items in **Color** tab in options (fig. 35).

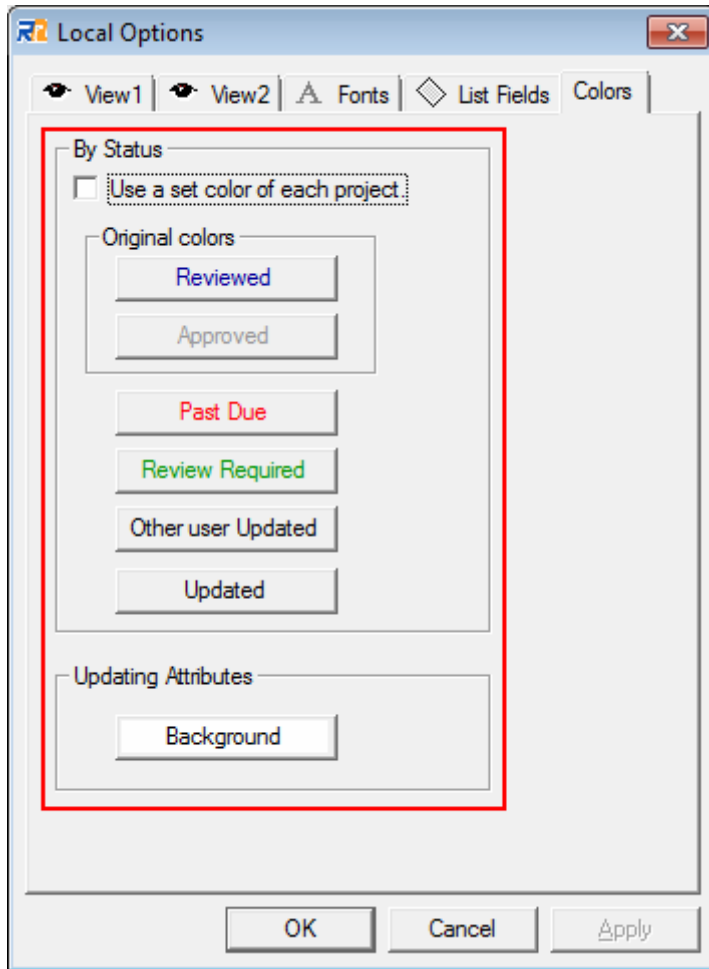


fig. 35

These features will help you manage requirements more efficiently.

## 7. Relationships between Requirements

As described so far, RaQuest has the ability to manage a lot of requirements efficiently. In addition to those features, one of the most important features of RaQuest is relationship.

When you modify a requirement, requirements having relationships with the other requirement are influenced in sequence and these influenced requirements should be reviewed. This section provides you with the details about such relationships among requirements.

To define a relationship between requirements, drag and drop the destination requirement to the source requirement between **Project** tree and **Requirements** list (the operation is bidirectional) or within **Project** tree or **Requirements** list.

In the following case as shown in fig. 36, for example, the requirement “High Volume Through-put” is influenced by the requirement “High volume – faster client accessibility.” It means that the change in the requirement “High volume – faster client accessibility” may affect, and may need to re-review, the requirement “High Volume Through-put,” and if it is the case, the requirement “High Volume Through-put” needs to be drag-and-dropped to the requirement “High volume – faster client accessibility” to define the relationship between them.

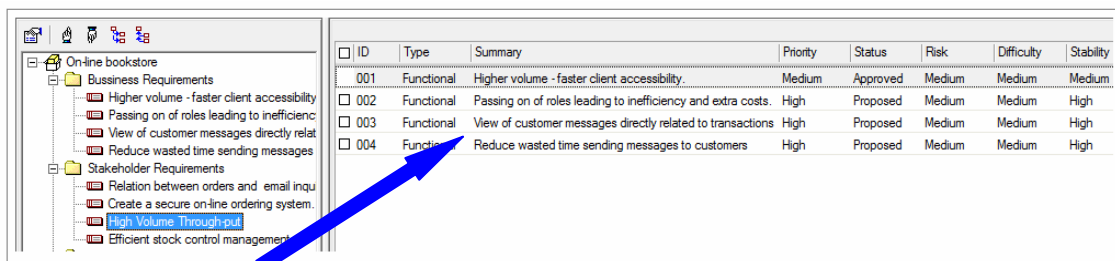


fig. 36

Depending on the **Project Options' General Feature1** tab settings, the relationships are automatically defined or deleted when the child requirement is created or moved.

RaQuest has a feature to illustrate relationship map among requirements and you can easily check relationships with this diagram feature. Right-click a requirement and select **Show Relationships Map** to display a relationships map which visually shows the relationships between selected requirement and the other requirements (fig. 37). Requirement you selected is drawn in red at the diagram (fig. 38).

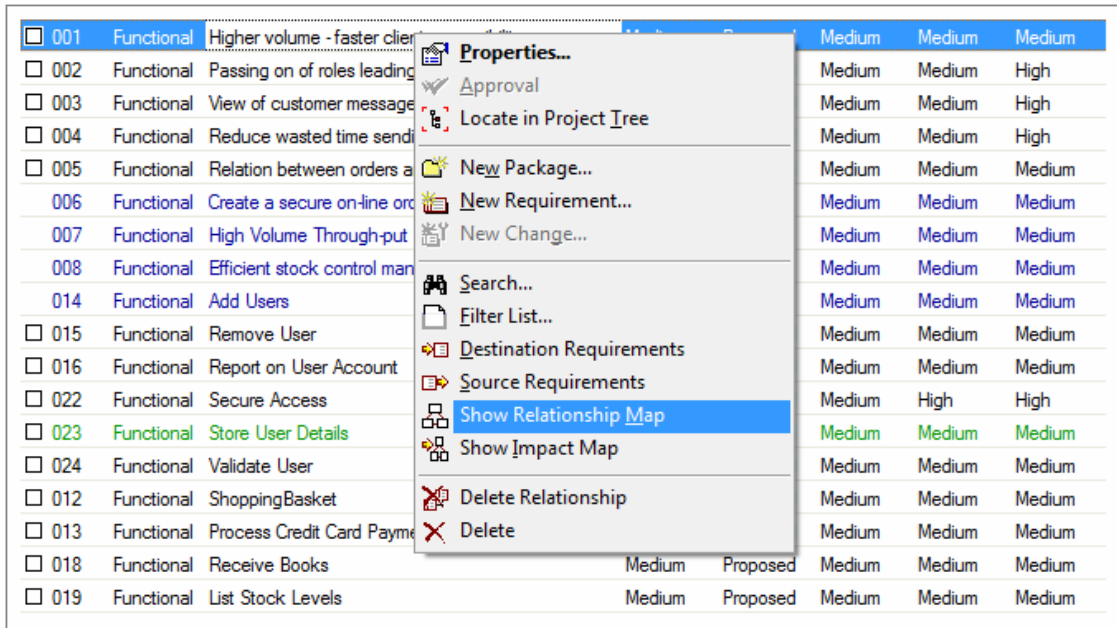


fig. 37

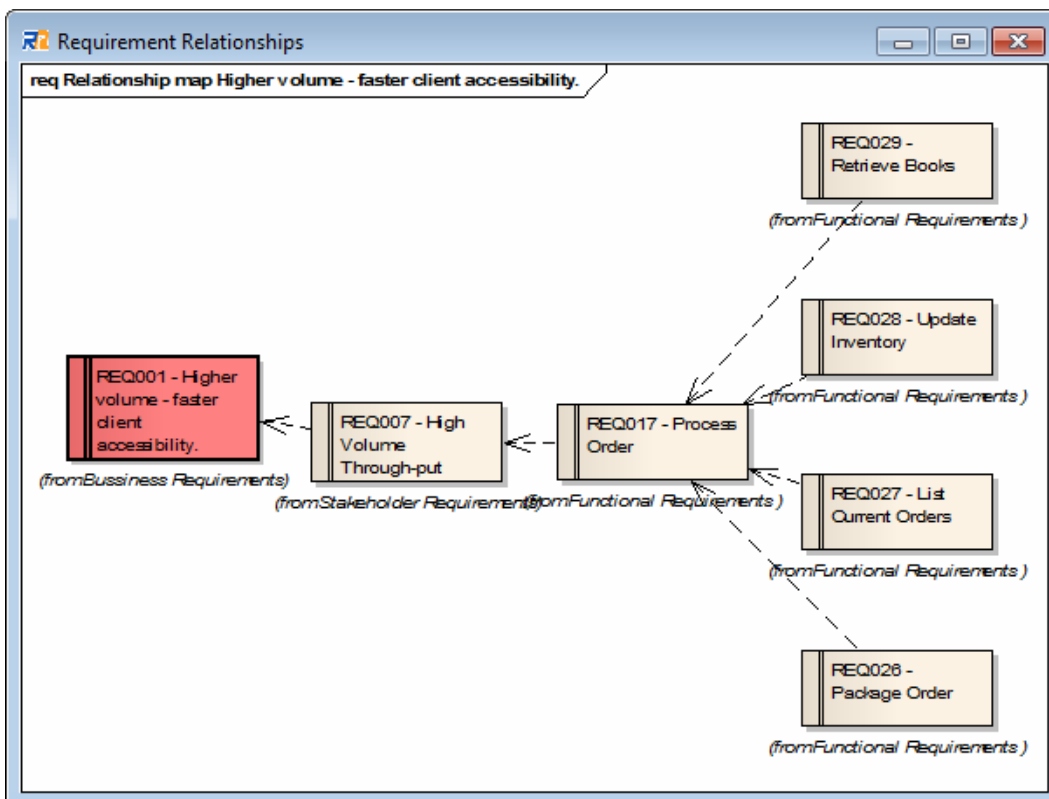


fig. 38

When you modify a requirement, requirements associated with the modified requirement might be influenced. In this case, you should review the requirements associated with the modified requirement. RaQuest has a feature to illustrate a diagram to show which requirements should be reviewed (fig. 39).

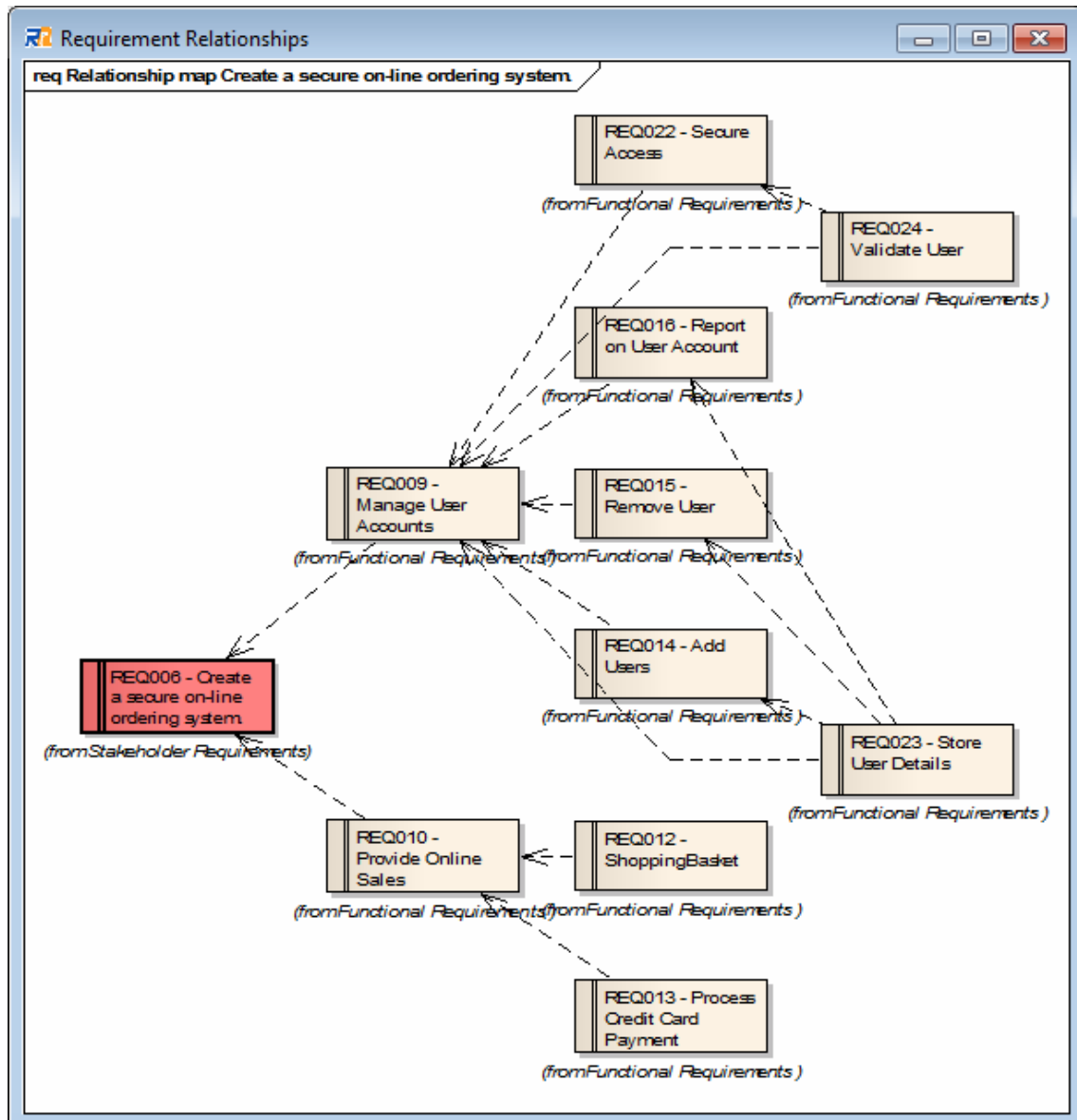


fig. 39

Above is illustrated with **Show Impact Map** feature. A requirement in red is the source

requirement, and the requirements in green are the destination requirements which should be reviewed. When the review is already over, the color of requirements is not changed.

When the requirement with **Review Required** status is depending on more than one requirement, it may be difficult to find out which requirements are influencing the requirement. You can find that out from the **Review Requirements** tab of the **Requirement Properties** dialog (fig. 40).

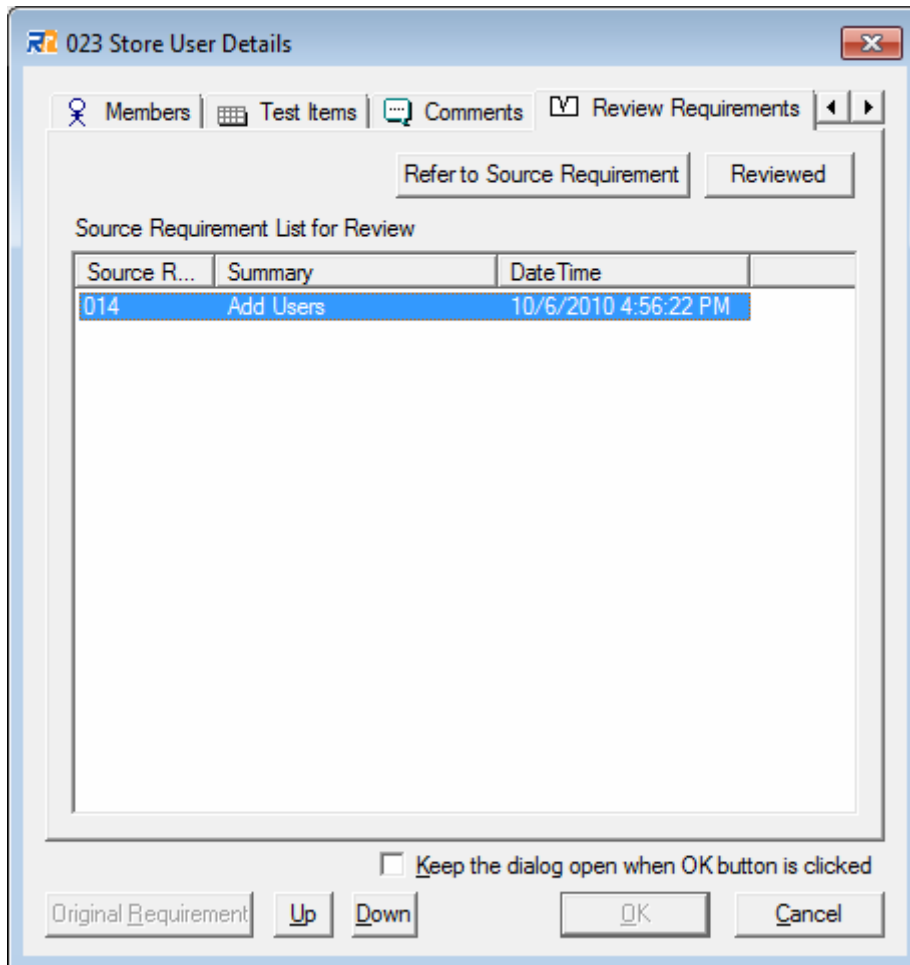


fig. 40

## 8. Matrix

As described in Section 7, RaQuest can show the relationships between requirements as the diagrams. In addition, RaQuest can also show the relationships between requirements in matrix form (fig. 41).

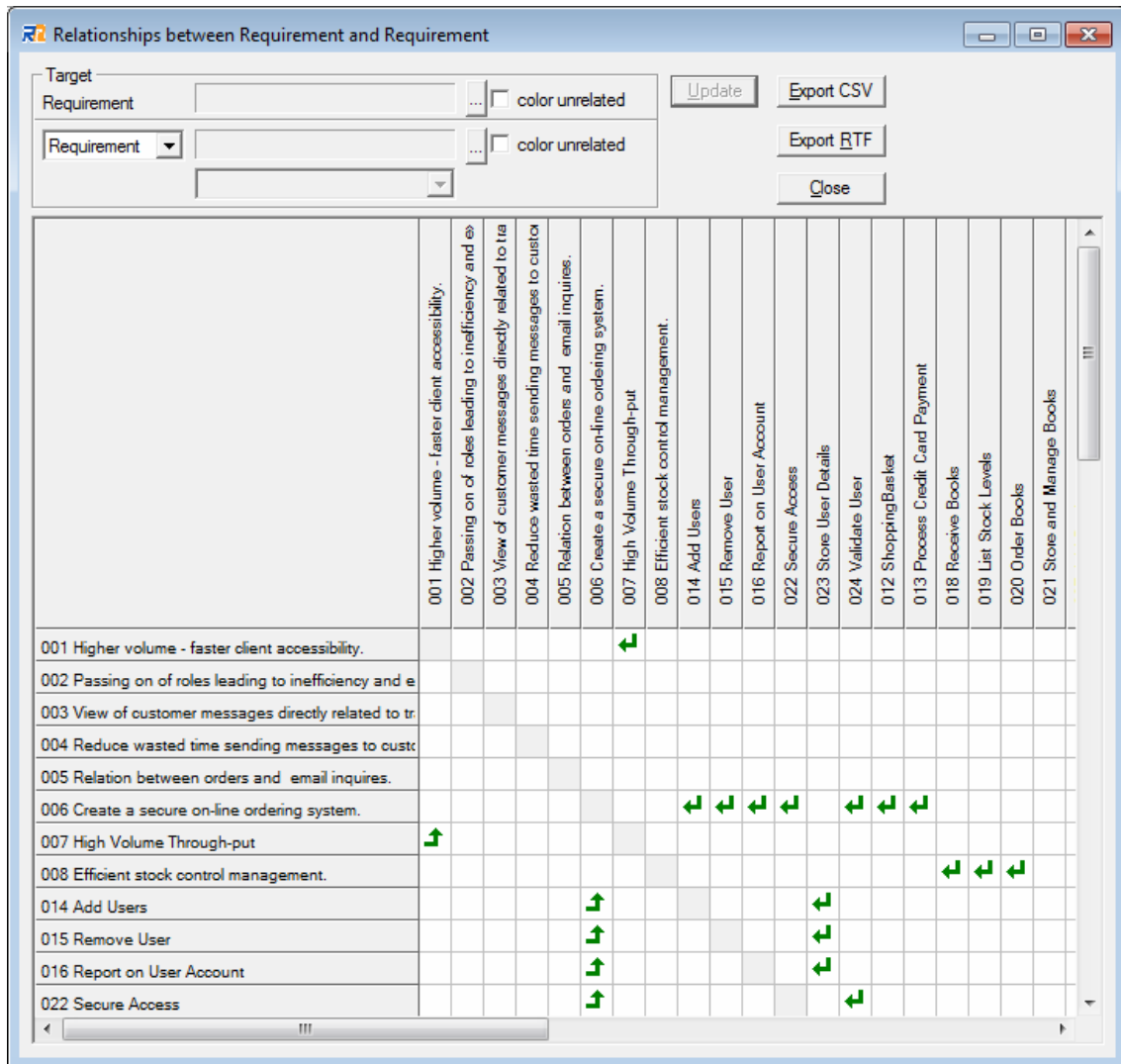


fig. 41

This matrix feature can also display relationships between requirements (fig. 42) and team members, relationships between requirements and Use Cases (fig. 43), and relationships between requirements and UML item (fig. 44). For example, if you use Use Case matrix, it is easy to check if Use Cases are defined for all requirements.

If you check the **color unrelated** checkbox and update it, the elements with no relationships will be

colored. So you can easily understand the unrelated elements.

The feature can not only display relationships, but also define new relationships. To define a new relationship, click the appropriate cell in matrix. You can also narrow down the requirements or Use Cases by selecting the target package in case of having lots of requirements and Use Cases.

It is also possible to export the data in CSV or RTF format.

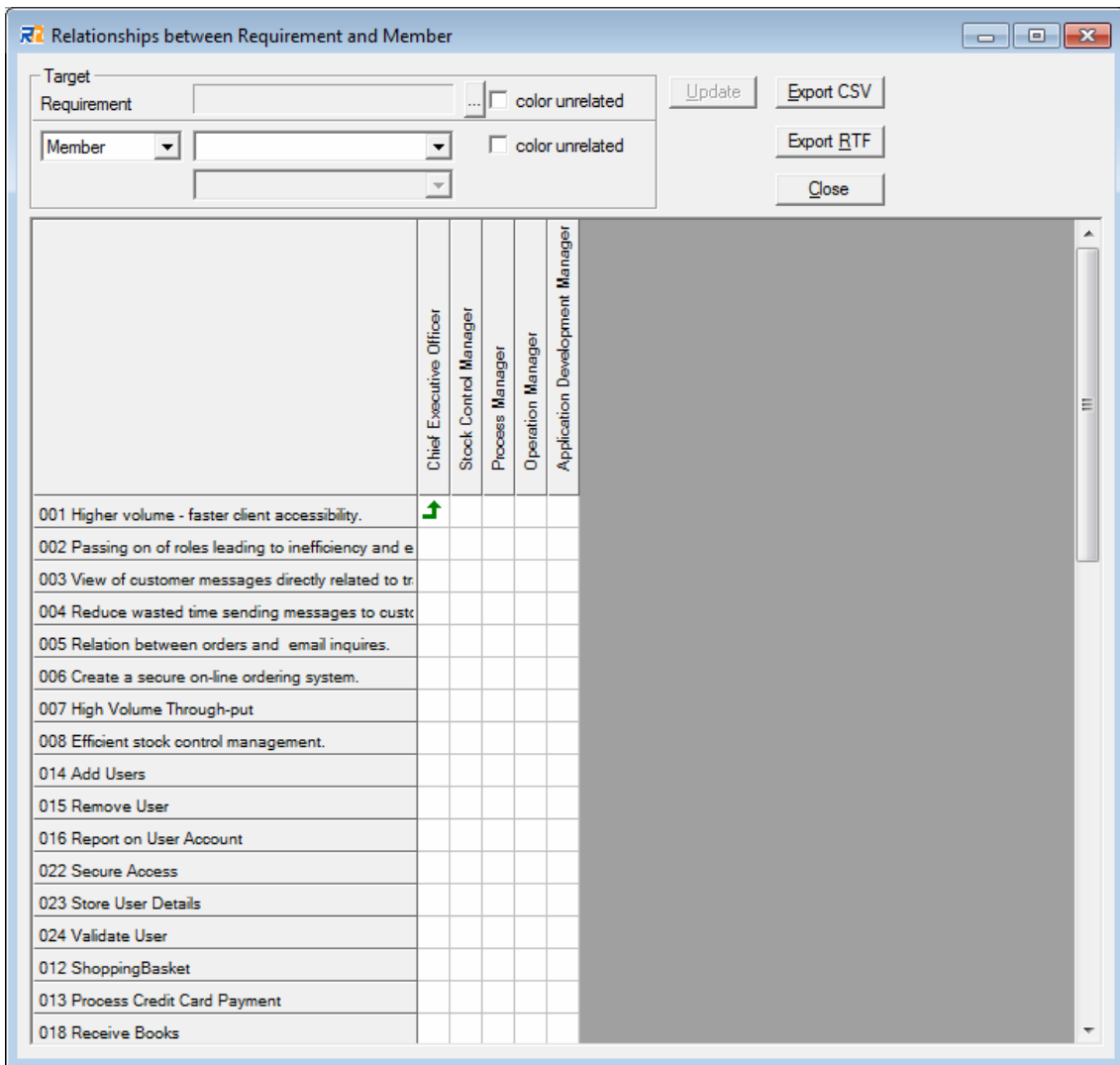


fig. 42

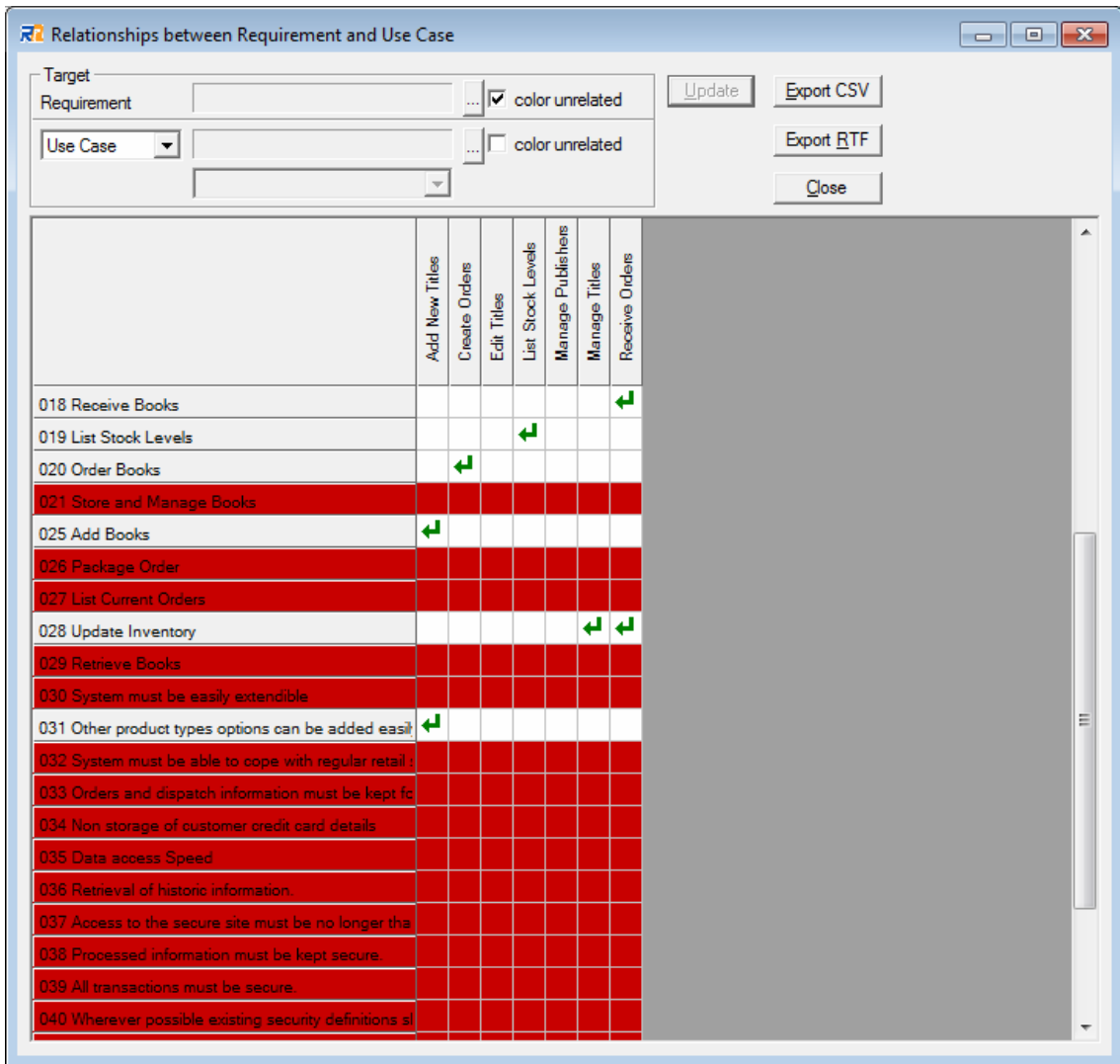


fig. 43

The requirements, which have no relationship with any Use Cases, are colored in fig. 43.

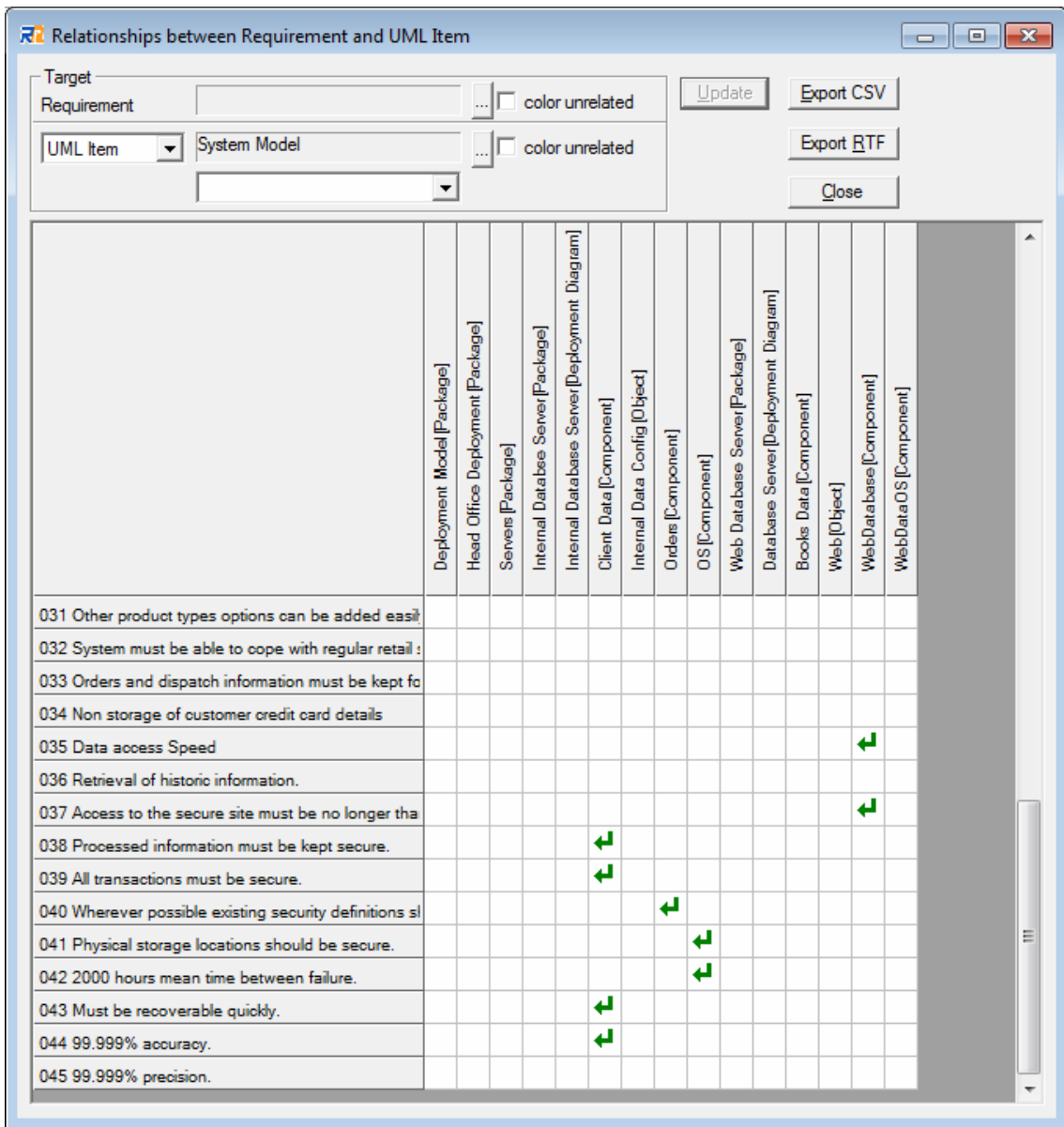


fig. 44

## 9. How to use Baseline

Requirements are always changing as the development goes on. To save the snapshot at one point, baseline feature is useful.

Baseline is not specified at the beginning. For this manual, we will use a development, which has some phases.

Firstly, if you set a new baseline as "Phase 1," the management dialog of the baseline is displayed as below (fig. 46).

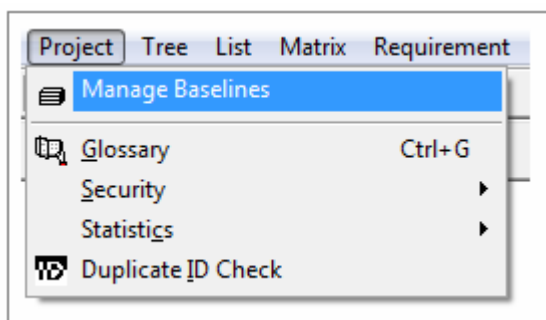
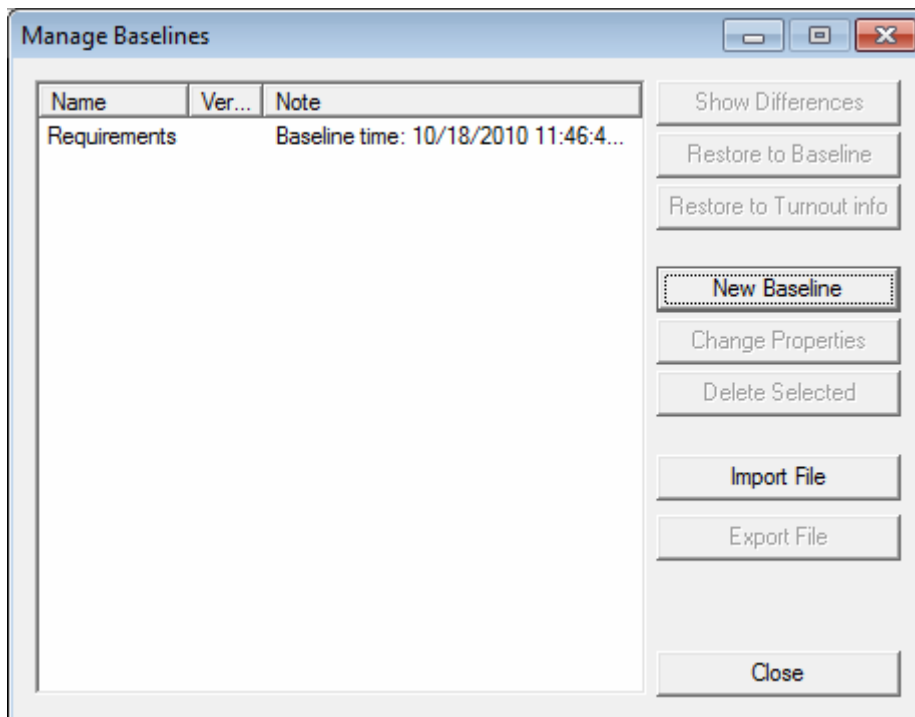


fig. 45



**fig. 46**

Continue your work with this condition. When the first phase is completed, execute **New Baseline** again for the second phase, and save the snapshot at the end of the Phase 1. Then, continue to work the baseline Phase 2, and perform the same work after that.

If you want to confirm the difference with the information of a former baseline while at work, it is possible to display the difference with a former baseline saved when executing **Show Differences of Baseline Management** dialog.

Or, if you need to return to the condition of a requirement at a certain point, it is possible by performing **Restore to Baseline**. You can also change a requirement after that and save it as a new baseline.

## 10. Management of Changes

Requirements in the current system are always shifting with time. The passage of time (even if the system is under the development) may require you to change the requirement, or the lack of understanding about the system may cause the change of the requirement.

For these kinds of requirement changes, direct modification is available for the certain development phase, within the same baseline. However, if the requirement needs to be changed across the baseline, the original requirement cannot be changed because of its status “Approved.” In such a case, you need to create and manage a **Change for the Requirement**.

As a concrete example, a requirement in version 1.0 needs to be changed in version 2.0. In this case, it is convenient to create the **Change for the Requirement** and manage it.

A change is effective only for a requirement with “Approved” status. Select the requirement with “Approved” status to create the change (fig. 47).

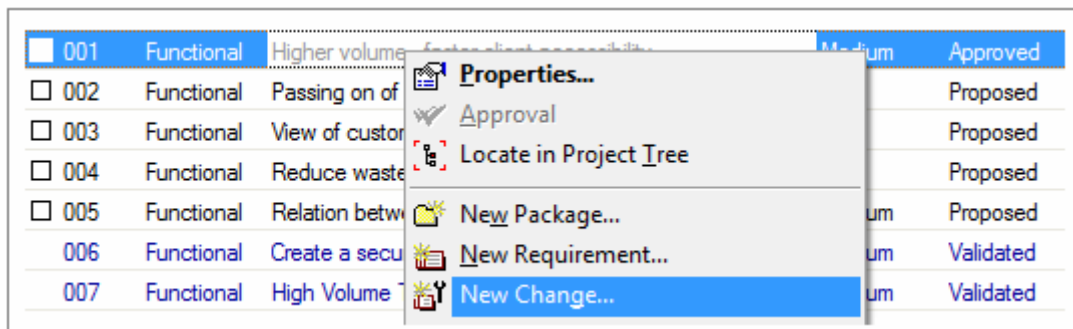


fig. 47

After creating the change, you can easily check the details of the original requirement from the properties dialog of the change for the requirement by pressing **Original Requirement** button (

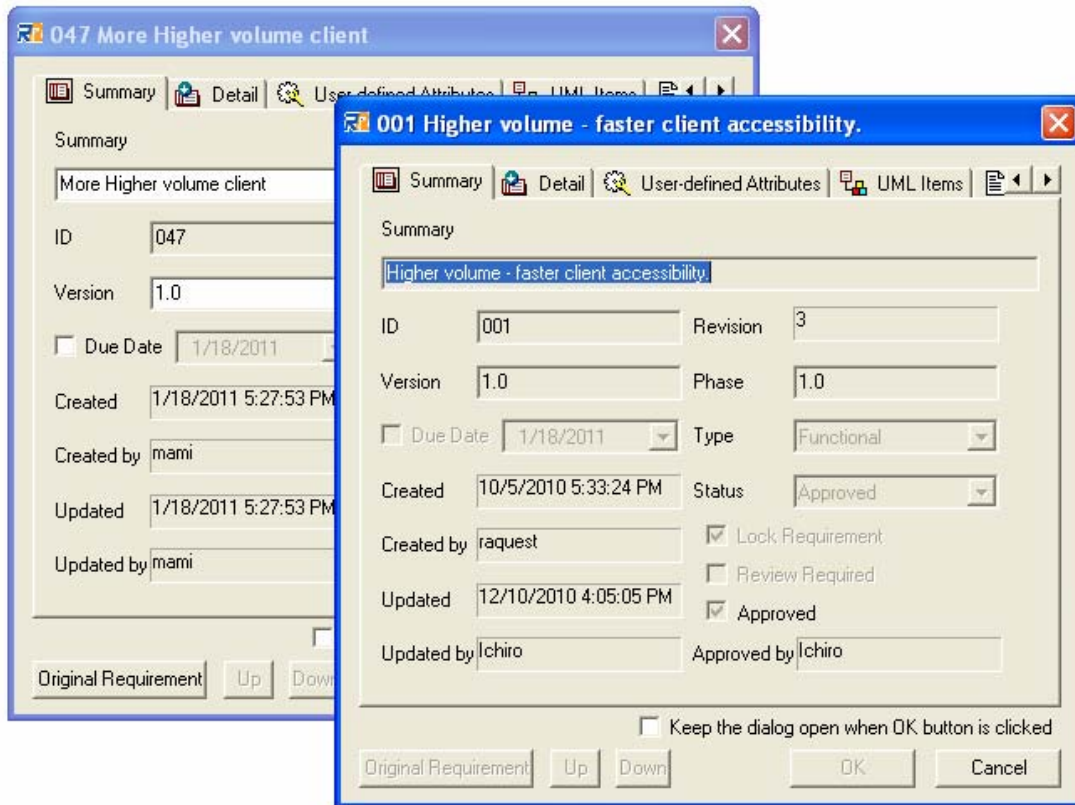


fig. 48).

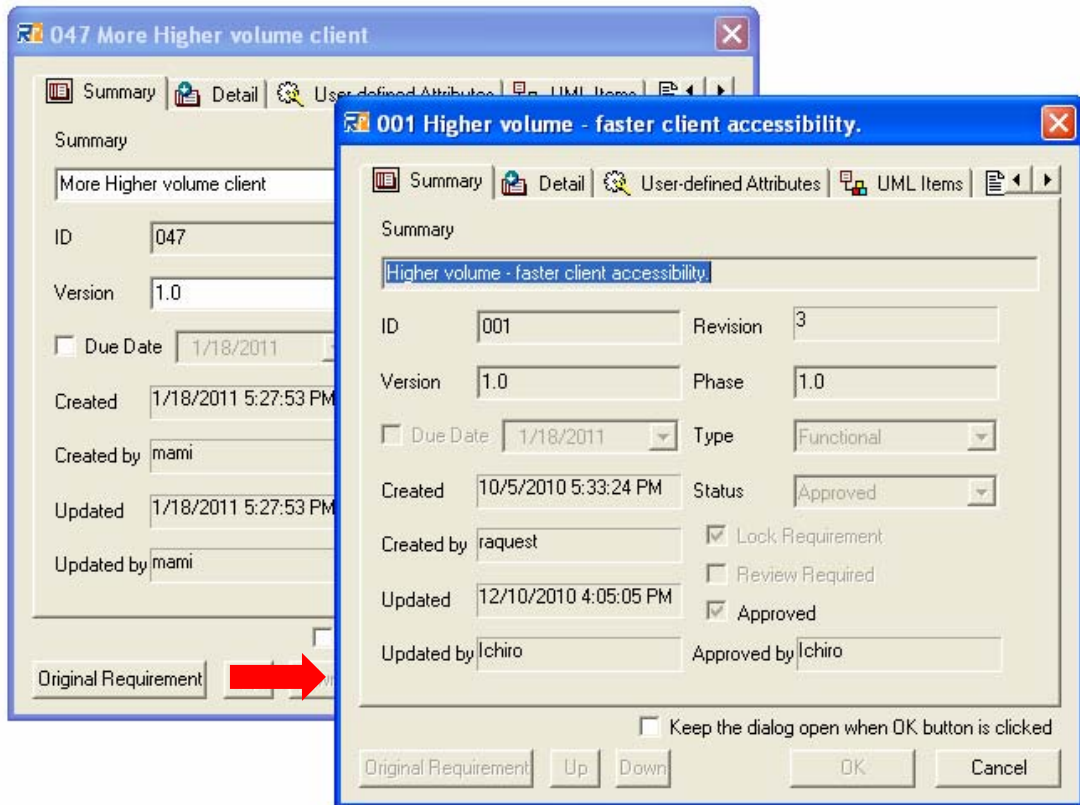


fig. 48

## 11. Documenting and Utilization of Data

Requirements created in RaQuest are stored in a project file. This project file is common to Enterprise Architect and is not readable by general documenting applications. But you may need to document requirements for customers or for discussions within your development team. RaQuest has a feature to document requirements. This section provides you with the way to document requirements.

RaQuest can export requirements in three formats: HTML (.htm), MS-Word (.doc), and RTF (.rtf). To export requirements in MS-Word format, MS-Word 2000 or later must be installed correctly in advance.

Let's export requirements in RTF format. First, select a package under where you would like to export requirements. Second, select **File | Export | RTF** menu option (fig. 49). Then you are required to specify the name and the location of the exported file and template.

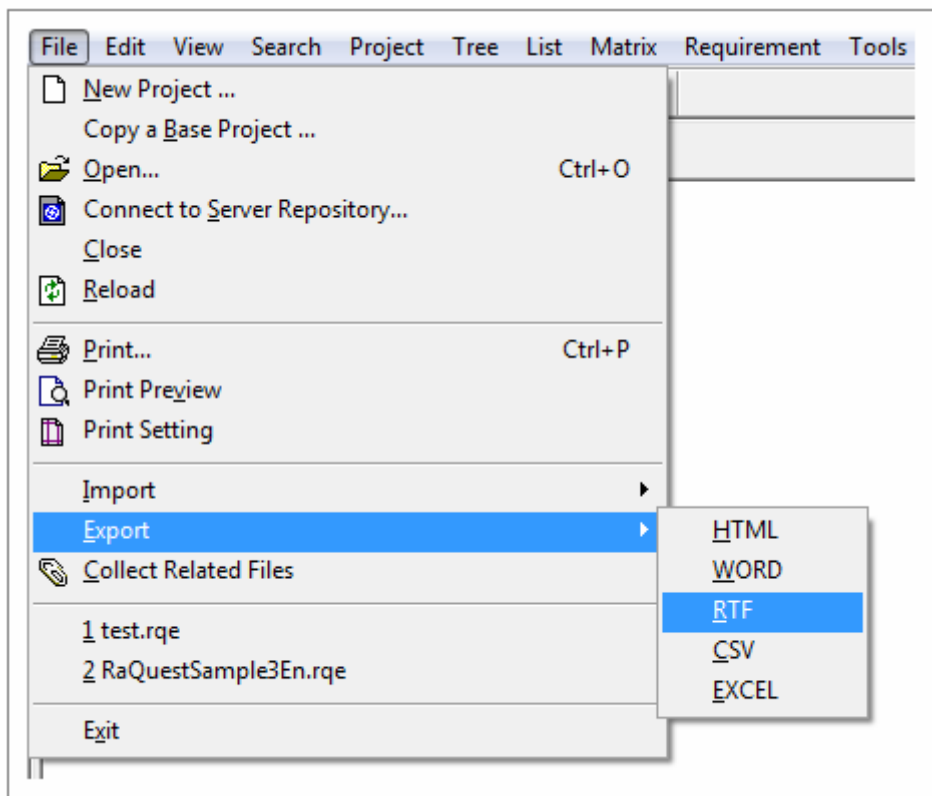


fig. 49

The following document (fig. 50) is a sample of an exported file in RTF format. You can customize it by using Enterprise Architect’s RTF template editor.

<a Sample of exported file in RTF>

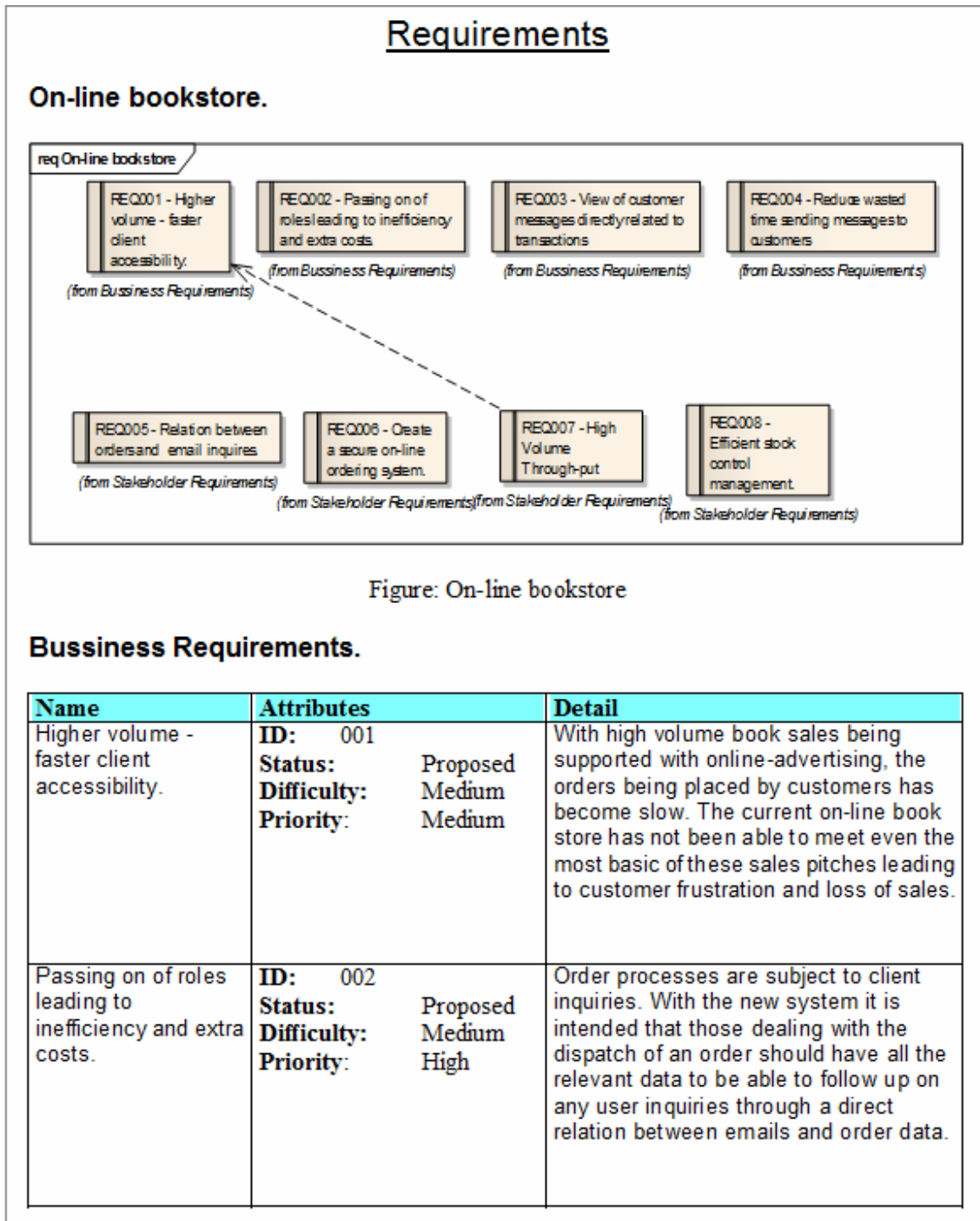


fig. 50

In the same way, you can also export requirements in MS-Word format. A sample of an exported file in MS-Word format is below. (fig. 51)

<a Sample of exported file in MS-Word format>

**On-line bookstore**

Summary: Execute user management/inventory management/order entry/order.

***Bussiness Requirements***

Summary:

**001 Higher volume - faster client accessibility.**

Detail: With high volume book sales being supported with online-advertising, the orders being placed by customers has become slow. The current on-line book store has not been able to meet even the most basic of these sales pitches leading to customer frustration and loss of sales.

Type: Functional Version: 1.0 Revision: 1 Phase: 1.0  
Date Due: Not defined  
Created: 10/5/2010 5:33:24 PM Author: raquest  
Last Updated: 10/5/2010 5:33:24 PM Updated By: raquest  
Status: Proposed Approved by: Lock: does not exist Review Required: unneeded  
Priority: Medium Difficulty: Medium Risk: Medium Effort: 0 Stability: Medium

Files: F001 minutes C:\file.txt

Members: Chief Executive Officer Project Manager

Destination Requirement ID: 007

fig. 51

You can also export requirements in HTML format. A sample of an exported file in HTML format is below (fig. 52).

10/18/2010 1:12:52 PM ssj

## On-line bookstore

Summary: Execute user management/inventory management/order entry/order.

### Business Requirements

Summary:

**001 Higher volume - faster client accessibility.**

Detail:	With high volume book sales being supported with online-advertising, the orders being placed by customers has become slow. The current on-line book store has not been able to meet even the most basic of these sales pitches leading to customer frustration and loss of sales.
Type:	Functional
Version:	1.0
Revision:	3
Phase:	1.0
Date Due:	Not defined
Created:	10/5/2010 5:33:24 PM
Author:	raquest
Last Updated:	10/18/2010 11:53:03 AM
Updated By:	ssj
Status:	Approved
Approved by:	ssj
Lock:	exists

fig. 52

RaQuest also has a feature to export requirements in CSV and Excel format. These files are useful for analyzing the data.

Select **File | Export | CSV** menu item. Then you are required to specify the name, location of the exported file, and so on.

The following fig. 53 is an example of an exported CSV file opened by MS-Excel.

Requirements Management Tool RaQuest 3.3 Features Guide

A	B	C	D	E	F	G	H
1	ID Type	Summary	Priority	Status	Risk	Difficulty	Stability
2	1 Functional	Higher volume - faster client accessibility.	Medium	Proposed	Medium	Medium	Medium
3	2 Functional	Passing on of roles leading to inefficiency and extra costs.	High	Proposed	Medium	Medium	High
4	3 Functional	View of customer messages directly related to transactions	High	Proposed	Medium	Medium	High
5	4 Functional	Reduce wasted time sending messages to customers	High	Proposed	Medium	Medium	High
6	5 Functional	Relation between orders and email inquires.	Medium	Proposed	Medium	Medium	Medium
7	6 Functional	Create a secure on-line ordering system.	Medium	Validated	Medium	Medium	Medium
8	7 Functional	High Volume Through-put	Medium	Validated	Medium	Medium	Medium
9	8 Functional	Efficient stock control management.	Medium	Validated	Medium	Medium	Medium
10	9 Functional	Manage User Accounts	Medium	Proposed	Medium	Medium	Medium
11	10 Functional	Provide Online Sales	Medium	Proposed	Medium	Medium	Medium
12	11 Functional	Manage Inventory	Medium	Proposed	Medium	Medium	Medium
13	12 Functional	ShoppingBasket	Low	Proposed	Medium	Medium	Medium
14	13 Functional	Process Credit Card Payment	High	Proposed	Medium	Medium	Medium
15	14 Functional	Add Users	High	Validated	Medium	Medium	Medium
16	15 Functional	Remove User	Low	Proposed	Medium	Medium	Medium
17	16 Functional	Report on User Account	High	Proposed	Medium	Medium	Medium

fig. 53

## 12. Importing the External Requirements Information

RaQuest can import the requirement data from the external tools like spreadsheet programs and such.

To import the requirement information from the tools like spreadsheet programs, the data has to be converted to the CSV file; RaQuest imports the CSV data. When importing, it is also possible to specify column in the CSV file to be allocated into the attribute of requirements.

If the project already has the same ID or GUID of requirement with the data of CSV file, the data can be overwritten. That means, by including ID or GUID to the exporting CSV/Excel data, explained in the previous section, it can synchronize the data between RaQuest and the external tools.

Importing the requirement information from the document files, rather than the spreadsheet tools files, is available only by MS-Word at the moment.

When Word Add-in is installed at the RaQuest's installation, right-click menu of the high-lighted texts has **Register to RaQuest** menu item. This option registers the high-lighted texts as a requirement of the currently opened project of RaQuest.

### 13. Notice of Changes

RaQuest can call external program on addition/deletion/attribute's changes of the requirements. By calling the mailing program, it is possible to give a notice of those actions.

By using mailing program Blat (<http://www.blat.net/>), you can set up option (エラー! 参照元が見つかりません。). When changes are made to the requirements, the following mail (エラー! 参照元が見つかりません。) will be transmitted.

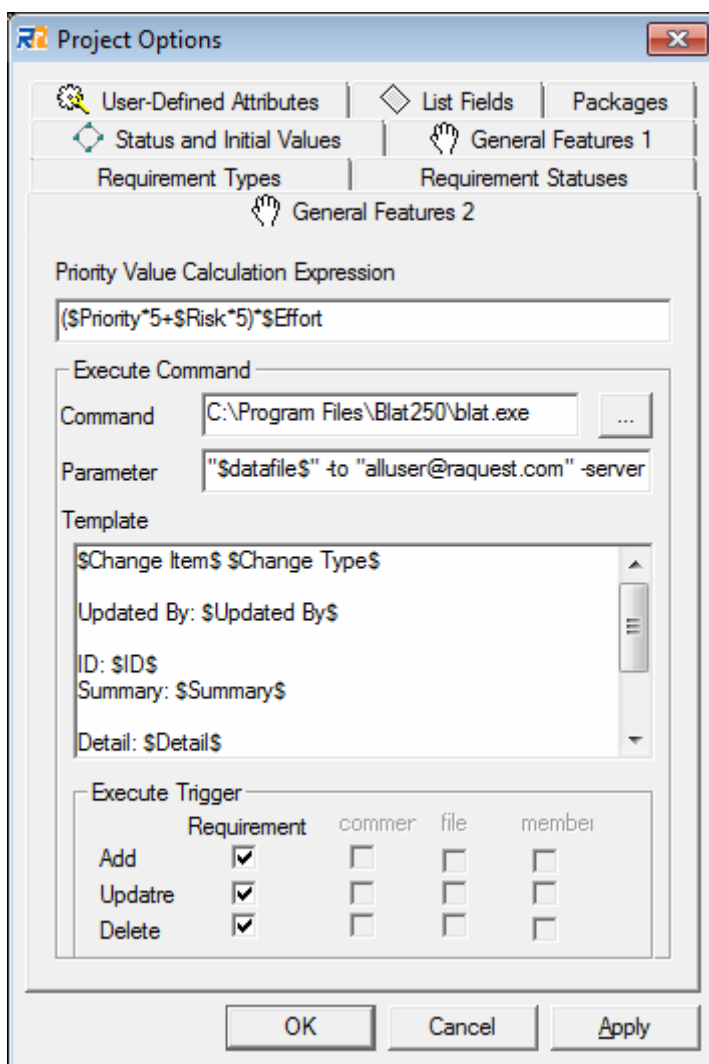


fig. 54

```
To: alluser@sparxsystems.co.jp↓  
Subject: Requirement Attribute update↓  
↓  
Requirement Attribute update↓  
↓  
Updated By: mani↓  
↓  
ID: 009↓  
Summary: Manage User Accounts↓  
↓  
Detail: The system is required to store and maintain a list of client accounts in a persistent repository↓  
↓  
Change Contents:↓  
Version : 1.0 -> 1.1↓
```

**fig. 55**

You can give a notice of changes efficiently by using this feature.

## 14. Integration of Enterprise Architect and RaQuest

As described above, not only project files are common to Enterprise Architect and RaQuest, but also Enterprise Architect and RaQuest can work together in various aspects. This section explains how they work together.

For example, RaQuest can generate Use Cases from requirements already defined (fig. 56). This feature makes it easy to illustrate diagrams of Use Cases. It is possible to pick up a certain type's requirements, like functional requirements, and export them as a Use Case.

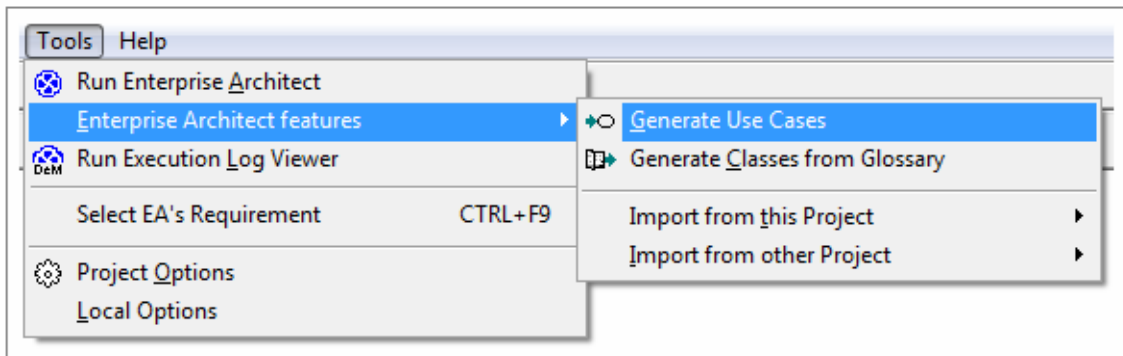
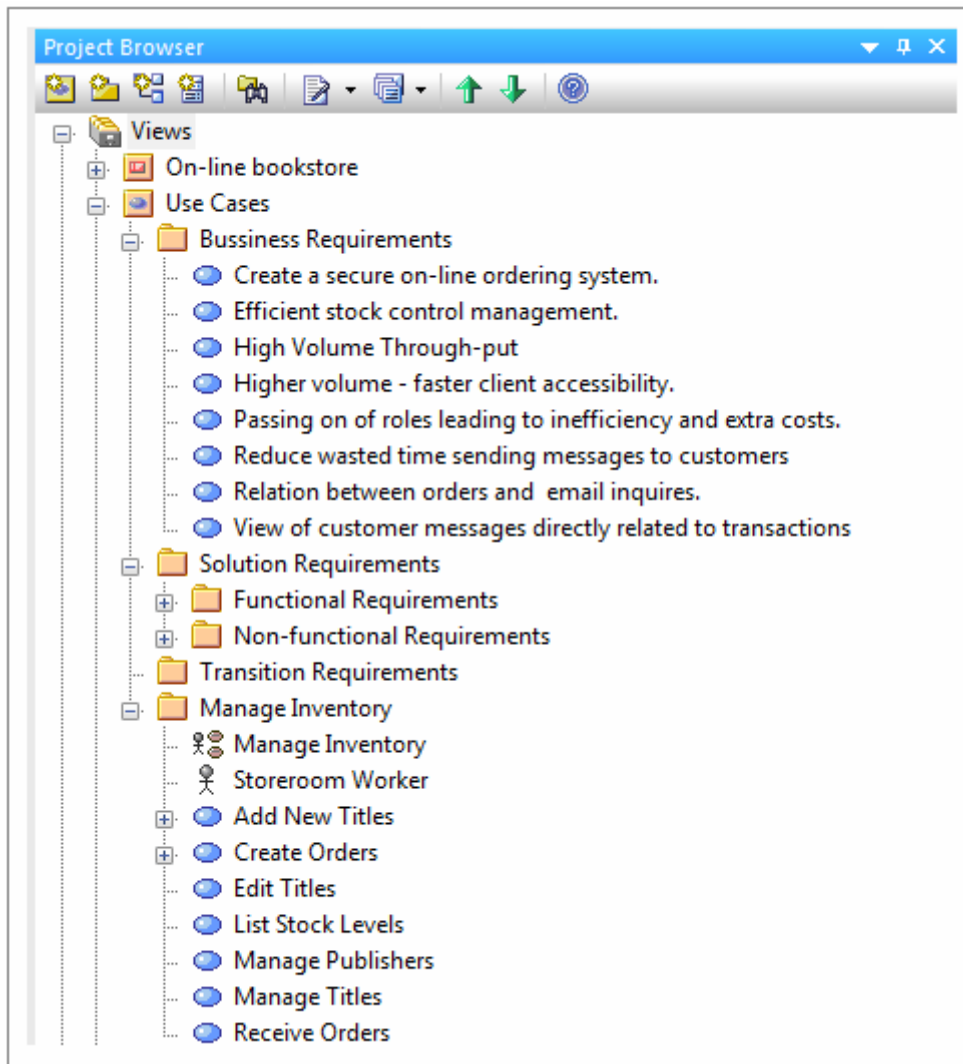


fig. 56

In this manual, we generate Use Cases from the sample project file as below (fig. 57). The location of the Use Cases can be specified as you want.



**fig. 57**

RaQuest can read existing Use Cases and requirements defined as the properties of Use Cases. This feature makes it easy to introduce RaQuest into existing projects or projects under development.

There is no barrier of data between Enterprise Architect and RaQuest. Enterprise Architect can use data of RaQuest as needed. You can make good use of abilities of Enterprise Architect with data of RaQuest.

The following shows the window which appears when the sample project file is opened by Enterprise Architect directly. You can see that requirements are stored in almost the same tree structure of RaQuest.

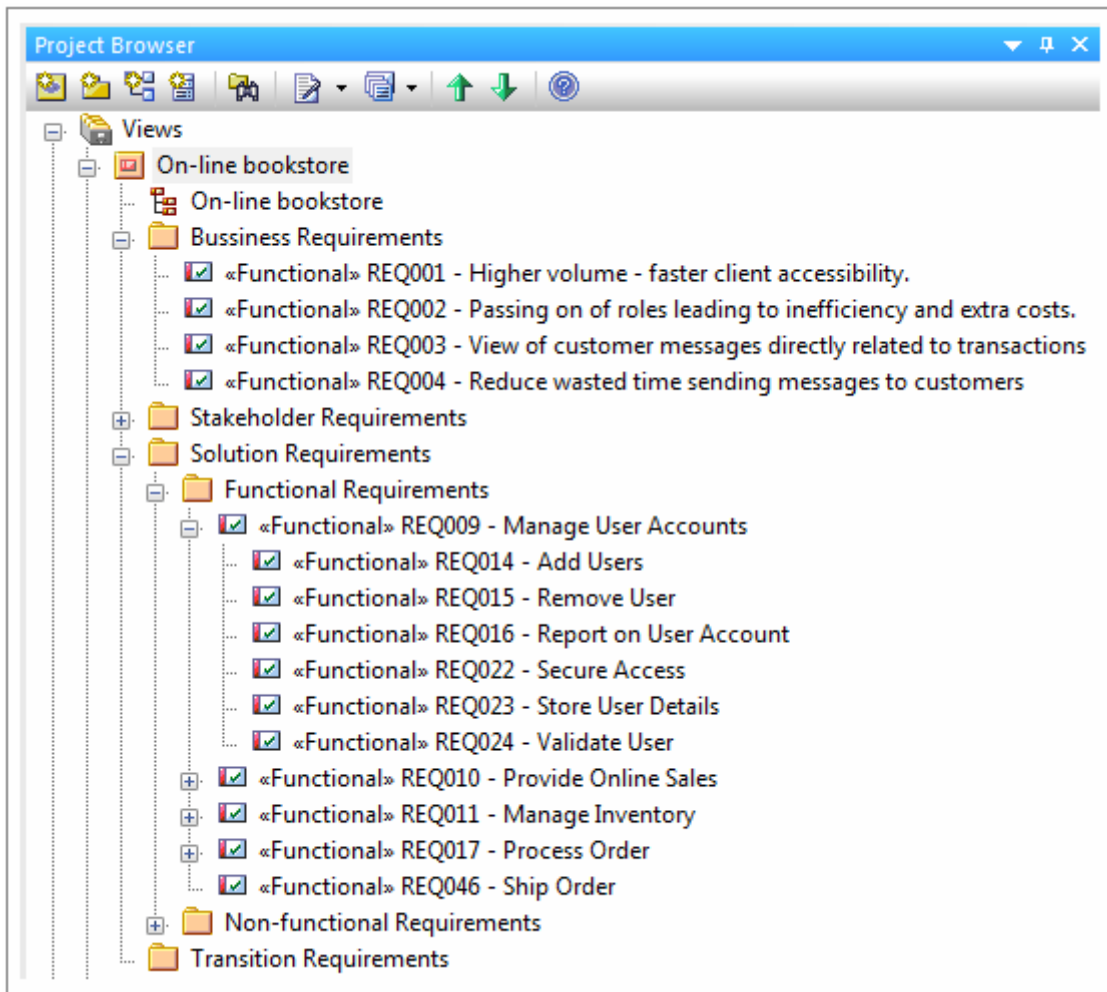


fig. 58

As an example of practical use of Enterprise Architect, you can also use **Generate RTF Documentation** feature of Enterprise Architect for the package made in RaQuest. Note that several attributes are not described in the documents generated by Enterprise Architect.

Except for the update log, the attributes, set in RaQuest but not existed in Enterprise Architect, are saved as tagged value.

The data compatibility enables you to use Enterprise Architect and RaQuest mutually and brings out the good points of Enterprise Architect and RaQuest.

RaQuest can also generate classes which have the same name as the terms from the glossary. This feature is useful in the various situations like illustrating relationships between requirements to analyze the system.

The operation for the **Glossary** dialog in RaQuest is the same as that in Enterprise Architect. Enter the terms into the **Glossary** dialog (fig. 59).

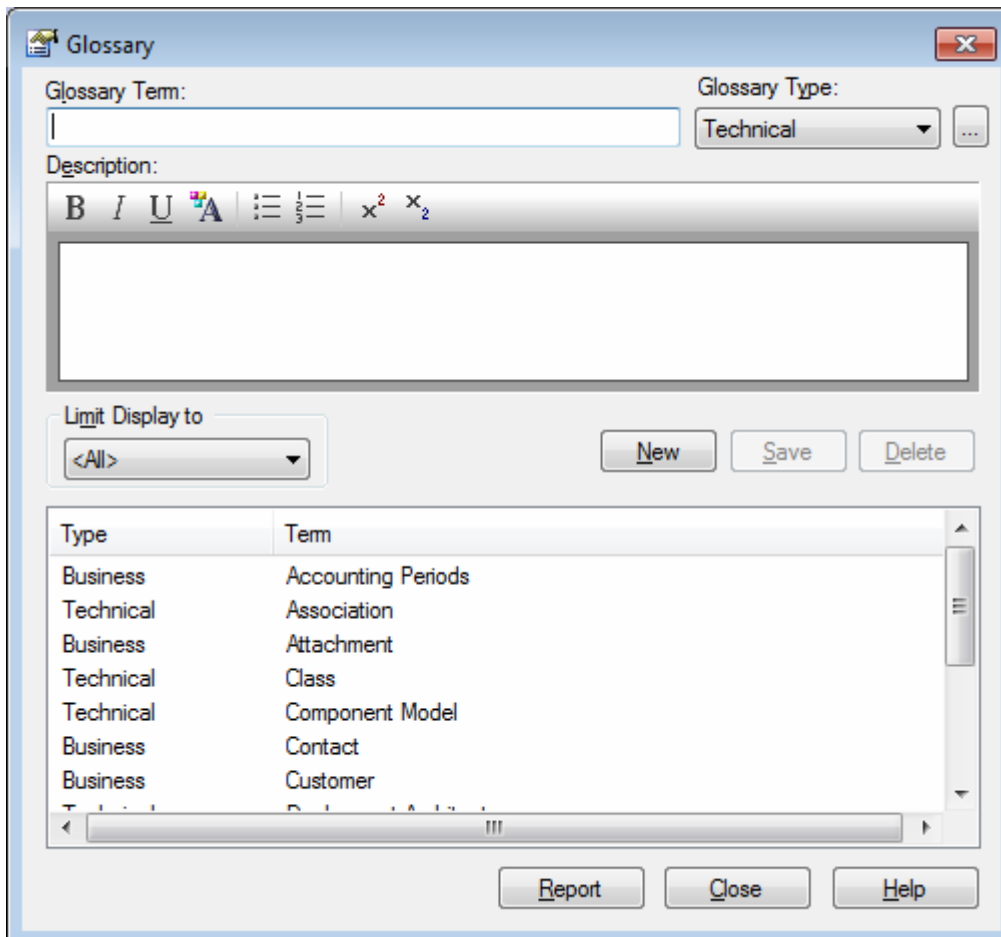


fig. 59

The terms registered in the glossary of RaQuest can be referred from the glossary of Enterprise Architect. Documenting them is also possible by Enterprise Architect.

**Generate Classes from Glossary** feature can generate classes from terms in glossary under the location you specified (fig. 60).

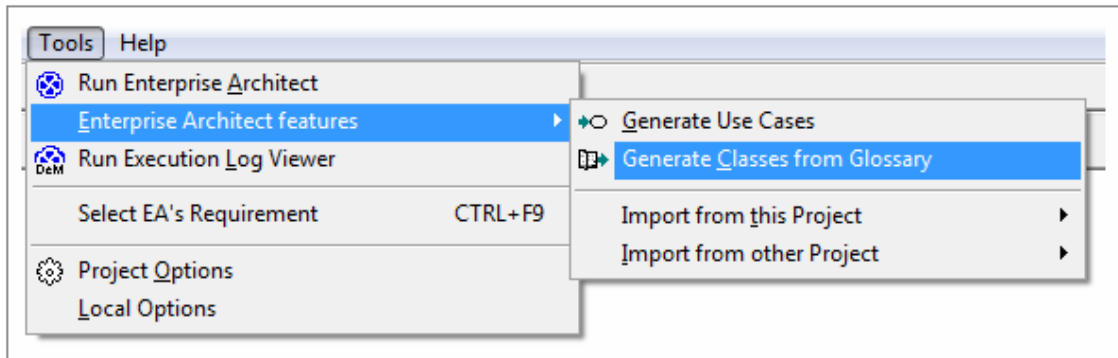


fig. 60

In future version, the feature to quote the terms from summary or comments of the requirements and to show the terms registered in the glossary in different colors will be implemented. These features will help you more.