Introduction to IRQA®4

Main functionality and use

Marcel Overeem

1/7/2011

Marcel Overeem is consultant at SpeedSoft BV and has written this document to provide a short overview of the main functionality and use of IRQA® 4.

Speedsoft BV
Phone (+31) – 6 30.48.99.48
Info: sales@speedsoft.nl
www.speedsoft.nl

SpeedSoft BV is distributor of IRQA® in the Benelux.

SpeedSoft

Table of contents

Introduction

Welcome to the IRQA® (Integral Requisite Analyzer) “Introduction to IRQA®” document. This is one of the most versatile, powerful Requirements Engineering support tools, developed by Visure Solutions S.L. This document will provide you a first introduction to the available capabilities and explain the concepts used in this solution.

IRQA® is a flexible tool, which under no circumstance imposes a methodology, but rather provides teams of analysts, users and developers a methodological framework of functionalities that may be adapted in each case to the specific needs.

IRQA® is not only a Requirement Management tool, but also provides complete support for the Requirements Engineering process, through a series of features that allow the basic activities related to that process to be carried out:

- Requirements capture
- Requirements analyses
- Specification of the solution
- Validation of the specification

It also provides the necessary support for other complementary activities, such as:

- Traceability
- Change impact analyses
- Test definition
- Report Generation
- Organization of the specification
- Configuration management
- Integration with design
- Reusability
- Workflow

This document is a first introduction to the Requirement Management and Requirement Engineering solution IRQA® and gives a short overview of the main capabilities, concepts and indented use of some of the capabilities.

Intended readers

This document is intended to give a first introduction to IRQA® for those who want a short overview of the capabilities of IRQA® and those that need to understand some more about the philosophy and ideas behind the use of these capabilities to be able to use them in an evaluation or first use.

The structure of this document

This document starts with a chapter explaining some of the main characteristics of IRQA®. The second chapter shows an overview of basic concepts and capabilities and the following chapters are clusters of capabilities of a particular subject.

Where the word IRQA® is used is meant the client of IRQA®.

Characteristics of IRQA®

IRQA® has, like all tools, its own characteristics by the way it behaves and the way the ideas for creating the ideal Requirement Engineering tool are worked out in functionality. This chapter will show a brief overview of these main characteristics of IRQA®.

Information centric

IRQA® treats information elements individually and stores them as individual objects in a database solution. This in contrast to document oriented solutions that organize and treat information on the level of documents. This will give you more control on the management of requirements and specifications.

Windows like behavior

Being a real modern Windows application IRQA® supports all the standard Windows functionality you are used to. Think of the use of different windows, Control and Shift keys, menu's, toolbars etc..

Process independency

The use of IRQA® is not restricted to any kind of development process. It can be used in a process manner as you like to configure it.

Configurable user interface

The way IRQA® presents information can be configured by the user. By choosing what elements to show in a view the user can define its own preferences. These Views can be stored by saving the made configuration. (See Sharing of Views) Also the way the desktop is organized can be configured by the user.

Immediate storage of changes

Any change made in the content within IRQA® will immediately be stored in the database. So no confirmation will be asked to save the made changes. The made changes will overwrite the old values of same version in version control.
Exceptions on this behavior are the storage of Views and Traceability matrixes. Also the saving of modifications to links in the Traceability matrix has to be explicitly done by the user.

Open architecture and integrations

IRQA® is setup with an open architecture in mind. Consequence is that it's not restricted to a particular database solution. Most database solutions, working with an ODBC connection can be used, like; Oracle, MS SQL Server, MS Access.

The integration with third party tools is possible by using the open API of IRQA. Using this API everybody can make integrations between third party tools and IRQA. Some integrations are already available or even standard, like the integration with HP's Quality Centre (former TestDirector).

Extended modules

IRQA® can be extended with modules like the Report Manager (See Reporting), the Quality Analyzer, Visual Prototyper, Embed-X etc. that are available through the IRQA® vendor.

Quality analyzer

The Quality analyzer is an available extended module (See extended modules) for automatically analyzing the quality of the written Requirements. It provides an analysis of the natural language. Spanish- and English language are standard available, other languages can be added, -ask your IRQA® vendor for availability of other languages.

The Quality analyzer checks the requirement descriptions on many configurable parameters and can show the quality per requirement, per user etc,. Also will it give suggestions to improve the requirements descriptions.
This module is of high value for improving the quality of the requirement descriptions; working with it will help users to improve their requirements writing capabilities and gives management insight in the requirements quality.

Exchanging contents with other tools

Another important facility is the possibility of exporting and importing contents through the use of XMI. This will also make the import from other RME tools like DOORS available. This will help, for instance, in the situation that Requirements have to be distributed through different suppliers even when using different tools.

Another exchange capability is the import and export to Word and Excel. This can be helpful when already these kinds of documents are used and the organization migrates to the use of IRQA.

Also a simple press on the button, when in a Requirement view (See Views), will export the shown content to Excel or export to Word from the document view.

A complete roundtrip from Word/Excell to IRQA and back is possible.

SpeedSoft

Concepts and capabilities of IRQA®

This chapter will describe briefly a collection of concepts and capabilities of IRQA®. It will give you an overview of the basic capabilities and concepts used.

Identification of elements like Requirements

A Requirement consists out of three parts separately discussed underneath.
Besides Requirements also some other elements, like services or test scenarios for instance, have a Code, Name and Description.
Other elements like actors or concepts do not have a Code; they are uniquely identified by Name.

Code

The first part of the identification of Requirements consists of a Coding. Each Requirement within the project will have its own unique coding number to make it uniquely identifiable within the project.
The coding can be made manually, -although the system will suggest one. The system will check if the typed code is indeed unique. If the code is already used it will warn you that the code is a duplicate and won't accept it.

Name

The second part of the identification of requirements consists of a name given to the Requirement by the user. The Name is like a label to the requirement so the user will be able to recognize what the Requirement is about. In practice it will be a very short formulation of the content, just to understand what it is about.

Description

The third part of a Requirement is called the Description. The Description contains the requirement text itself and supports complete RTF functionality.
The Description can also contain Hyperlinks to Associated Files (See Associated Files) outside IRQA® and can contain objects like pictures.

Versioning (check in & checkout) of elements

IRQA® is able to version **all** types of elements within the tool by the use of the check in and checkout mechanism. Not only Requirements, but also Models, Attributes, etc. can be versioned this way.
Checked in versions cannot be changed and will be kept in the database. Older versions can be viewed or even made active again. Only checked out elements can be changed. So for changing the content of a Requirement the Requirement has first to be checked out (create a new version).
Comments can be added during check in and the account name of the user who checks in will be stored.
Different versions can automatically be compared for differences. The Baselines to which the element version belongs can also be found in the Properties panel (See Properties panel).

Baselining

Baselining is an important capability for a Requirement Management tool. The possibility to freeze a set of Requirements is very helpful during the development process.
IRQA® can handle Baselines in different ways. A whole project can be Base lined, but it is also possible to Base line only parts of a project (which is pretty unique).
So can, for instance, only a certain Part of the project be Base lined. This is especially useful when working with Reusable Components (See Reusability) and in incremental/iterative types of development process.

Attributes

Attributes will give your more control in analyzing and management of Requirements. Attributes can also be used for the enforcement of Workflows (See Workflows).

Attribute types

In IRQA® types of Attributes can be defined by the user, standard types are; Boolean, Date, Float, Integer etc.. By making your own Attribute type you can make for instance an Attribute type for the use with priority attributes, defining your own priorities like; Low, Medium, High etc..

Multivalue Attributes

Within IRQA® you can also use Multivalue attributes, so giving you the possibility to be able to pick more than one value from the list instead of just one as with a non-multivalue attributes.

Mandatory Attributes

It is possible to make attributes mandatory. This will mean that a value has to be set at creation. This often is a wanted behavior.

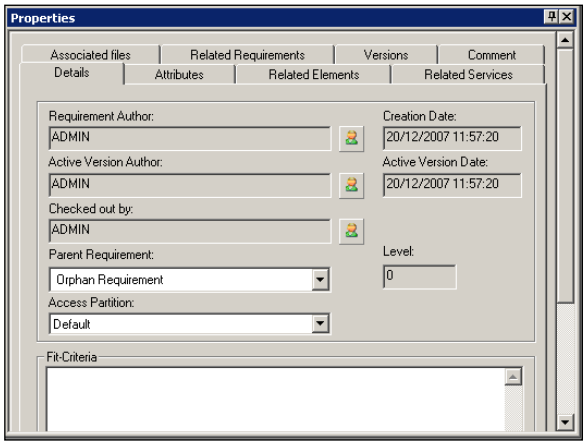
Colors

The different values of Attributes can also be associated with different colors by the user, for instance making the value High red; will give you a strong visual distinction to other values.

Properties Panel

Most elements, like Requirements, Services, Test scenarios, have a Properties panel. This Property panel shows, depending on the type of element, different Panel tabs with additional information on the element.

As example we take the Properties panel of Requirements. This Property panel has 8 different Panel tabs.
Four of the Panel tabs show the relations to other elements, like the Related Requirements-, Related Elements-, Related Services and the Associated Files Panel tabs. Also there are Panel tabs for showing the Versions and Check in Comments of the Requirement.
The Details Panel tab shows more details of the Requirement and the Attribute Panel tab show the Attributes that are associated to the Requirement.



Example of a Properties panel

Context information - Glossary

A Glossary can be build by defining Concepts. The Concepts are presented in a list and have their own Descriptions. Other elements like Requirements can be linked to these Concepts.

Also for analyzing and developing Concept definitions the use of the Concept Diagrams (or Entity-relationship diagram) is very powerful. It will give you the possibility to analyze and define the concepts in your domain of interest. The Concept Diagram appears like a Class Diagrams, but only the upper part of the class symbols is used.

Automatic analyzer

This capability of IRQA® will give you the possibility to let IRQA® automatically search the Projects contents and to propose possible relationships between Requirements or Services and defined Concepts. Giving these Requirements and Services a context.

Test scenarios

Besides managing Requirements IRQA® is also capable of managing Test scenarios.
The Test scenarios can be stored and managed like Requirements and Services. The Test scenarios provide information in the Properties panel (See Property panel) like: Pre- & Post conditions, Responsible, Environment, Parent child relations between Test scenarios and of course the Scenario descriptions themselves.
This way IRQA® provides Traceability (See Traceability) between Requirements or Services and Test scenarios.
The use of external Test solutions could become unnecessary in those cases where no automatic testing is used.

Workflows

IRQA® can enforce processes by the use of Workflows. These Workflows are based on user defined Mandatory Attributes (See Attributes) and are designed through graphical State Diagrams.
In the State Diagram each possible Attribute value is represented by a state in the State Diagram. The arrows contain the allowed transitions that make the State change. Also it is possible to let IRQA® execute a script, for instance to send a mail when the transition appears.

This functionality can be used to give only certain users the right to change the value of an Attribute and also the order in which the values of an Attribute can be changed.
Often organizations don't want everybody to be able to change the status of a Requirement, for example.

Workflows can help you to create and enforce your own processes like Change management process.

Associated files

IRQA® gives you the possibility to associate files within the Descriptions of for instance Requirements. (See also Description). This works like a Hyperlink.
Association to an external file can also be done in the Associated Files tab of the Properties Panel (See also Properties panel).
Also is it possible to link to a specific point within the document (Bookmark).
The external files can even be under version control as part of a version management system like, CVS, Subversion or others.

SpeedSoft

Structuring and Navigation

One of the most important and also most distinctive features of IRQA® is the powerful and easy to use possibilities for structuring and navigation. Especially the graphical structure design and navigation with Domains and Blocks is unique to IRQA®.

The management of the contents can be done on different methods and on different levels, as will be shown in this chapter.

Repositories

The database administrator can, in the IRQA® Admin Centre, define different repositories (databases) to store projects. Each repository will be separately stored depending on the used database solution.

Projects

A repository can be separated in individual Projects. Each Project can have its own Project administrator, users, groups and access partitions (See Rights Management). Projects will be managed by the Project Administrators. Only users who are associated to a Project can access the Project.
Opening a Project will only show the content of that particular Project, contents of other Projects are not accessible. An exception to this is the use of Components with reusability in mind (See Reusability). Projects can only be created by the database administrator in the IRQA® Admin Centre.

Domains

A Project can be divided into, what is called in IRQA®, Domains. Activating a Domain will have as effect that all content that is not associated to that Domain will not be shown to the user. The profit is that the number of material to be controlled will be reduced dramatically. The use of domains can be very helpful when having large and complex projects.
Domains contain **all** types of elements and will therefore have effect on all elements; -it also contains the Block Diagrams (See Blocks).

In case certain elements should be accessible through different Domains (shared) then these elements can be associated to more than one Domain. An example of this could be Concepts and Concept Diagrams valid to more than one Domain or your requirements model.

Domains can be used for breaking up the Projects content into different system parts or process parts, for example.

Associate elements to a domain

When a Domain is made active all newly created elements will automatically be associated to that Domain. So you don't have to associate elements manually to the Domain.
Of course the solution will let you associate elements to more than one domain or disconnect from a Domain.

To be able to associate already existing elements to a Domain first no Domain must be active. This can be done by "activating" a relation between two of the Domains or to select "Whole specification" in the Domain toolbar (See Navigation). Now all elements in the project are available and can be associated to a Domain by associating them in the "Included Elements" tab of the Properties panel (See Properties panel) of the Domain.

!TIP
Working with Domains asks for some discipline, because forgetting to activate a Domain will of course cause that new created elements will not be associated to the Domain and afterwards should be associated manually. So for not forgetting to activate a Domain it can be helpful to let IRQA® standard open with a Domain Diagram, –instead of the standard Requirement view. This can be done by setting the Initial view to the wanted Domain Diagram.

Blocks

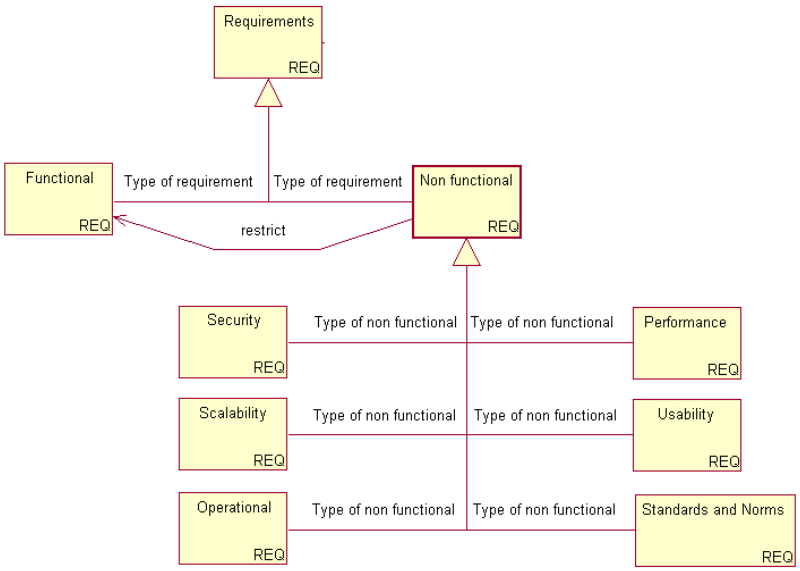
With the Blocks in a Block Diagram you can structure your project. You can, for example, differentiate your Requirements into different types of Requirements, like stakeholder- and system requirements or different sub types of non-functional requirements. But also system decompositions can be expressed with Blocks. Blocks only contain one type of element, instead of all element types as Domains do. The user can define and use Blocks as he wishes.

By activating a Block the contents of that Block will be shown. Again this will reduce the amount of elements under investigation as you can select only that part of the Project that is of interest. It could be that you are only interested in the stakeholder requirement on a particular moment. Also, as with Domains, will new created elements automatically be associated to the active Block.

The Blocks can be linked to express certain relationships between them. Two types of relationships are available; Generalization and Association.
The Blocks and their relationships are also used for Traceability (See Traceability).

Generalization relationship

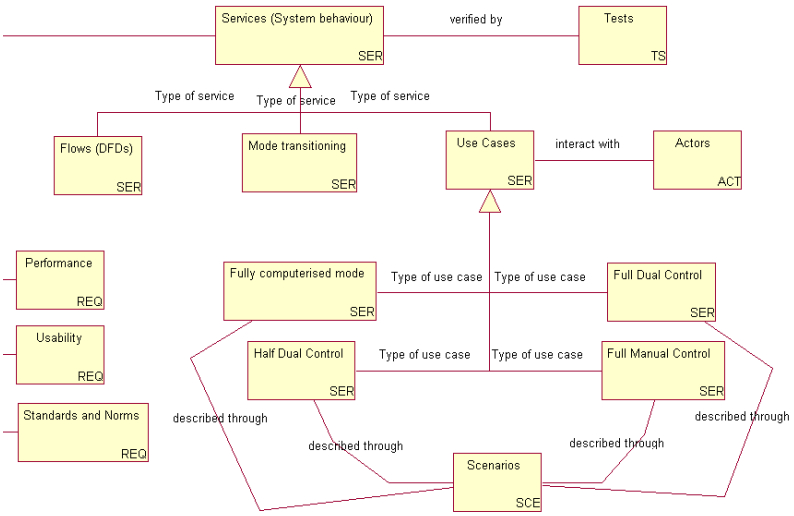
Generalization relationship creates relations for expressing subtypes or subparts, like subtypes of Non-functional Requirements in example underneath. These kinds of relationships are mostly expressed by the use of “Type of” or “Part of” or “Is”. The sub Blocks will also inherit all Attributes from the parent Block. So defining attributes for the parent Block will make these attributes also available for the underlying child Blocks, making the use of Attributes even simpler.
This kind of relationship structure makes it possible to activate a subpart to view, let’s say Performance Requirements. And only the Performance Requirements will be shown. But you can also activate a higher level like Non Functional requirements. Then the content of the Non functional Block plus the contents of all underlying child Blocks will be shown. So you can decide on what level you want to see the Contents.



Association relationship

Association relationship expresses all other kinds of relationships that can exist between Blocks, like “derived from” between System Requirements and Stakeholder Requirements. In the figure underneath the association relation between Services Block and Tests Block is expressed as “verified by”, expressing that the Services will be verified by corresponding Tests.

Also by designing the Block Diagram this way you create the possibility to associate the individual elements in those Blocks with each other by the means of the expressed relationship or motive. This way Traceability between elements in the Blocks is made possible (See Traceability). In the IRQA® Admin Centre the Project administrator can set an option that will make IRQA® only allow to associate elements with each other according to the designed relationships in the Block Diagram, restricting the freedom for making relations between elements.



Block Types

As you can see in above figure are there several types of Blocks which contain their corresponding types of elements. Available Block types are: Requirements, Services, Actors, Concepts, Scenarios and Test scenarios. Generalization relationship is only possible between Blocks of the same type. Association relationships are possible between every type of Block.

Hierarchies, like Parent child

Besides the already mentioned methods for structuring you can also structure the content by the use of Hierarchy like Parent child relations. This will create a hierarchical structure, with different levels of Requirements.
Another way to create a Hierarchy is by associating Requirements to their corresponding Services (See Views, See Relations and Traceability)

!TIP
Although you can make different Block diagrams for each Domain, it will probably be handier to make one Block diagram, that classifies all the different types of Requirements and other element types, and to associate this Block diagram to all Domains (Requirements model). Then the structure will be the same in all Domains, but still holds different contents.

Navigation

Navigation in IRQA® is done by using the Project structures as described in Domains and Blocks before in this chapter. The navigation itself can be done graphically or by use of dropdown menus.

Graphical Navigation

Using the Domain- and Block diagrams gives you the possibility to graphically navigate through the different parts of the Projects content. By activating a Domain in a Domain diagram you select first which parts of the Project you are interested in, like part of the system or part of the process or something like that. By activating a Domain you now will only see the contents within that Domain. By then also activating a certain Block in a Block diagram you will get a View with the contents of that Block, -stakeholder requirements, for instance. So the result is that you will see the stakeholder requirements that belong to a certain Domain. Choosing another Domain would give other stakeholder requirements.

If you then want something else to see, like system requirements, then you activate another corresponding Block with system requirements in the Block diagram.
So by using Domains and Blocks in the diagrams you can graphically navigate to the contents of your interest.
The activated Domain or Block is recognizable by the bold edges.

The red square will activate your chosen Domain or Block

Toolbars

Another way to navigate within IRQA® Projects is to use the Toolbars for Domains and Blocks. The Toolbars have a dropdown menu with the Domains or Blocks that are available. You activate a Domain or Block simply by selecting one from the list.
Also this is a very handy way to quickly navigate through the contents of the Project. Being in a Requirement view (See Views) you can quickly change the content of the view by selecting a different Block in the Block toolbar.

!TIP
Place both toolbars for Domains and Blocks next to each other on the desktop of IRQA. This will be your place to navigate.

Left the Domain toolbar and right the Blocks toolbar, also do they contain the buttons to go to the diagrams or the list overview.

Views

In IRQA® there are different ways to look at the material within the tool, these are called Views. The different possibilities to look at the content in IRQA® will be shown to you in this chapter.

Requirement views

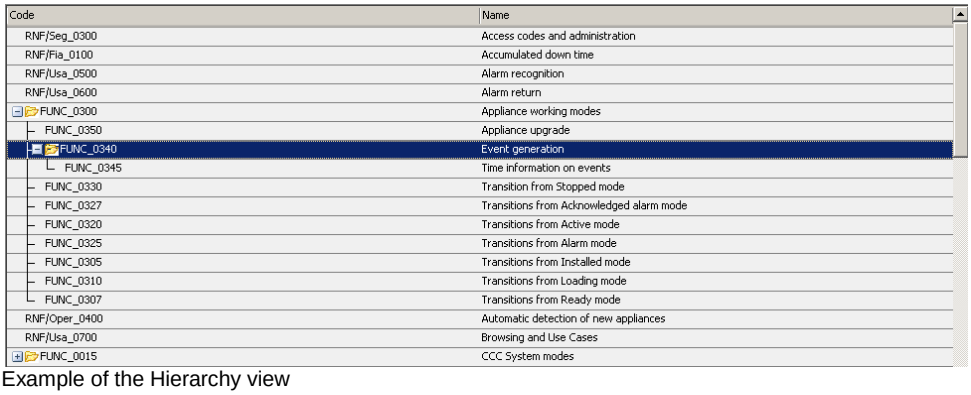
For viewing Requirements there are following types of Views to look at the requirements.

List view

This View is just a plain list of requirements ordered on the Coding number (See identification of elements). The contents of the List view depends on the selected part of the Project by means of activated Domains and / or Blocks and filters (See Structuring and Navigation).

Hierarchy view

The Hierarchy view shows Requirements in a hierarchical structure, so child requirements are shown a hierarchical level deeper than the parent. This structure can go as deep as the user creates them, -although it's wise not to make too many levels.



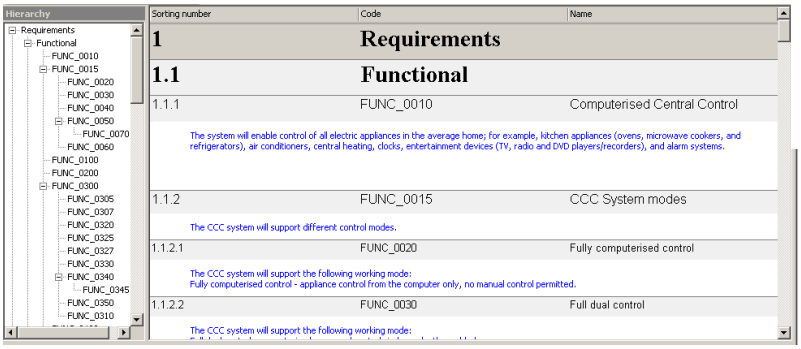
Example of the Hierarchy view

Document view

The Document view shows the requirements in a way that resembles to a structured MS Word document kind of presentation. The Requirements are shown in the structure of a Block diagram (See Structuring and navigation), where the Blocks are represented as headings of chapters and paragraphs. The chapters, paragraphs and requirements get additional paragraph numbers in the way like 1.2.3 etc. This numbering is independent from the Coding numbering (See Identification of elements). These numbers are just to provide a document structure and are automatically generated. Moving a Requirement to another chapter or paragraph will automatically reorder the document structure numbering. Also the Requirement will be associated to the corresponding Block of the new paragraph. With creating different Block diagrams it also possible to create different intersections or views.

The Document view has the advantage of having all contents of Blocks (See Blocks) presented in a structured way in one view. You can add elements into the right Blocks in this view just by adding them in the corresponding paragraph. Also can you drag and drop elements and put them in a different Block by dragging them to another paragraph. You can even change the order of the paragraphs, simply by drag and drop the paragraphs in the Document view. The setup of this Document view can be stored.
Another advantage is that the Descriptions are directly under the Code and Name of the corresponding element instead of being shown in a separate window on another place on the screen.

The document view can also be exported to MS-WORD or Excel and printed, so be used for reporting.



Example of the Document View

Services view

Like Requirements Hierarchy view also Requirements can be viewed in a list where they are shown as belonging to their corresponding Services. So the hierarchy shows the Services with their associated Requirements a hierarchical level deeper.
All configuring, storing, sharing and filtering capabilities for Requirements views are available here.

Configuration of Views

Change the shown element types

IRQA® provides the possibility to change the shown content in Views by changing the shown elements. For example one can choose which Attributes should be shown, or not. These View configurations can be stored separately and reused by the individual user or even be shared with other users (See Sharing of views).

Filters

Another way to manipulate the shown content is by the use of Filters. Filters make it possible to show certain parts of the content that fulfils a certain condition. For instance, a filter can be defined for showing only the Requirements with Attribute priority with value High, leaving the rest of the Requirements out of sight.
Also Filters can be shared among users.

Opening more Views at the same time

Normally IRQA® shows only one View with a certain type of elements, like Requirements. Choosing a different selection of Requirements will just simply replace the content of the opened View with the new selection. This is not always wanted behavior.

Sometimes it can be very handy to be able to work with two opened Views with same element type at the same time on the screen. When writing system requirements it is very handy to have also a View with the stakeholder requirements opened so you can see the stakeholder requirements where you write your system requirement against. Also for making relations between elements it is very handy to have two Views opened.
This can be done either by pressing the Control key and opening a View with, for instance requirements, while already a View with requirements is opened, or by checking the corresponding option in the Views management window, IRQA® will then open standard a second View of an element type instead of just replacing the contents of an already opened View.

Sharing of Views

Every user will be able to share views with other users belonging to the same user group. An administrator can define some views and make these available to the users by actively sharing these views.

Views have to be explicitly shared: that is, not all views created by each user are automatically shared; each user decides which views are to be shared.

SpeedSoft

Relations and Traceability

Requirement Engineering without support for impact analyses would not be complete. This chapter will show the different facilities of IRQA® for Relationships and Traceability to support you.

Relations

Most elements in IRQA® can be associated to each other by making a relation between the elements. So can Requirements be associated to each other, like stakeholder and system requirements. But also can Requirements be associated to Services (Use Cases) or Test scenario's to Services or/and Requirements (See Properties panel). Also parent child relations between Requirements of the same type are a type of relation.
Purpose is the capability to provide Traceability and the use of Suspect Links.

Suspect Links

The term Suspect Link refers to the capability of IRQA® to track changes in a chain of linked elements. This means that if a stakeholder requirement is changed while being linked to one or more system requirement(s), the links to these system requirements will become suspect due to the change in the linked stakeholder requirement.

The idea behind this is that when something changes in the chain, the rest of the chain has to be examined. The change can have consequences for the other elements. Perhaps it will mean that also a system requirement has to be rewritten. Due to the Suspect Link the user becomes aware of the relation and has to check the chain.

The Suspect Link functionality has to be switched on by a Project administrator in the IRQA® Admin Centre. Mostly you want to use this functionality when having a pretty stable set of requirements. The consequence of using the Suspect Link functionality from scratch would mean that during the development of requirements constantly many links will become Suspect. This behavior can be unwanted in early phases of Requirements Engineering.

Traceability

Two types of Traceability are provided: direct and indirect Traceability.

Direct Relationships

Direct relationship means that two elements are directly connected to each other by the definition of a link between them. This relation can be made by selecting the other element in the properties panel of the element or with the option in the menu when clicking on the right mouse button in a view (See Views).
The existing Direct relationships are shown in the, so called, Traceability matrix. Also the direction of the relationship is made visible.
The Traceability matrix will be shown after selecting this function in the Tools menu or by activating a relationship in a Block diagram (See Navigation). When activating a relationship in a Block diagram the Traceability matrix will be automatically configured to show the relations between the elements in the two Blocks that are linked by the selected relationship between these Blocks.

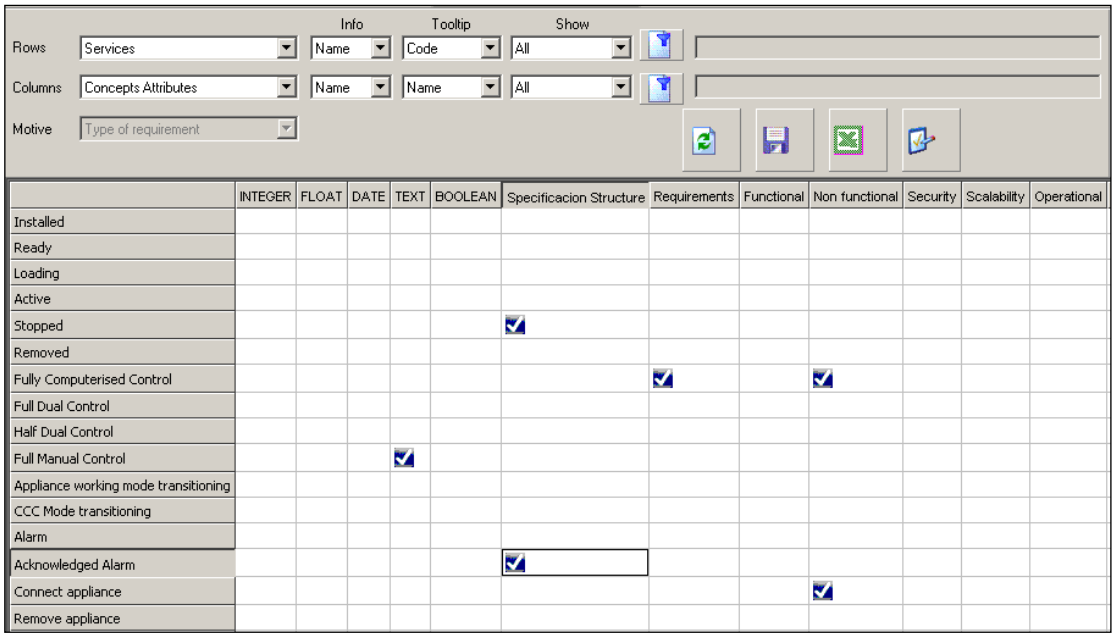
It is possible to manipulate the relations in the Traceability matrix by creating or deleting relationships. The direction of the relations can be chosen just by clicking on them.
The Traceability Matrix can be configured for what elements to be shown. Also filtering, storing and sharing of the Traceability Matrix configurations are available.

Changes made in the Traceability matrix have to be saved explicitly (See Immediate storage of changes); otherwise they will be lost when closing the Traceability matrix.

Indirect Relationships

Indirect Relationships is the capability of IRQA® to show the relations between elements that are not directly connected to each other. It is available through the use of Blocks in a Block diagram (See Blocks).
When selecting two Blocks (use shift key) that have no direct relation, -probably will have one or more Blocks between them-, you can view the Indirect relationships by activating the Indirect Relationship button. The selection of a path will be asked; -more optional paths to come from one Block to another can be possible.

In this way, for instance, is visible how stakeholder requirements are related to system tests, although the corresponding Blocks have no direct relations between them. The relations exist by the relations between stakeholder requirements and system requirements and the relations between system requirements and system tests.
This Indirect relationship matrix isn't configurable and also can't be used to manipulate relations, -as they are not direct relations.
The matrix can although also be saved or exported to Excel simply by pressing the Excel button.



	INTEGER	FLOAT	DATE	TEXT	BOOLEAN	Specificacion Structure	Requirements	Functional	Non functional	Security	Scalability	Operational
Installed												
Ready												
Loading												
Active												
Stopped						☑						
Removed												
Fully Computerised Control							☑		☑			
Full Dual Control												
Half Dual Control												
Full Manual Control				☑								
Appliance working mode transitioning												
CCC Mode transitioning												
Alarm												
Acknowledged Alarm						☑						
Connect appliance									☑			
Remove appliance												

Example of a Traceability matrix.

Modeling capabilities

IRQA® can support the Requirement Engineering phase by the use of different types of models. A subset from the UML and functional modeling diagrams are provided. This chapter will show you briefly an overview of the provided Modeling capabilities within IRQA.

UML modeling

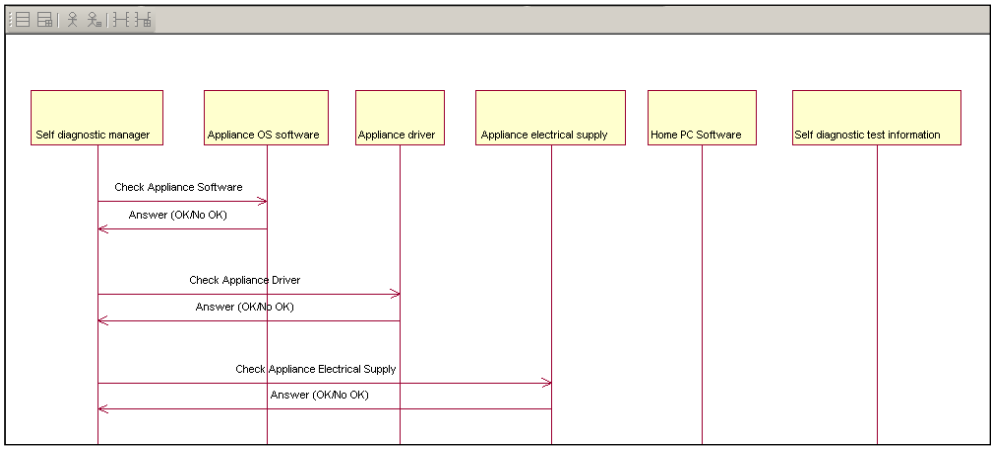
IRQA® provides the following UML models: Use Case Diagram, Sequence Diagram, State Diagram and Concept Diagram.

Use Case Diagram

The Use Case Diagram can be used in different ways. It can serve as Use Case Diagrams to analyze business processes and for communication with Stakeholders. But it can also be used for specifying the solution on high level by giving insight in the system services.

Sequence Diagram

Also this diagram can be used in different stages of the development process. It can be used for detailing business processes but also for specifying system behavior in Services (Use Cases).



Example of a Sequence diagram

State Diagram

The State Diagram is used as graphical detail for the Services (or Use Cases), but is also used for Workflow definition (See Workflow).

Functional modeling

IRQA® provides the following Functional models: Data Flow Diagram, Context Diagram and Entity relationship diagram.
Functional models in IRQA® are used to represent the interaction of the system to be built with the outside world (external entities) and to describe in detail the system functionalities.

Reusability

IRQA® supports the reusability of contents by the use of, so called, Components. Components can be defined and published by a Project and shared over different Projects.
Reusability is very helpful when certain parts of the Requirements or system specifications within an organization are valid to more Projects. Examples of this can be; legal Requirements, standards, or non-functional requirements.
Components are also very useful when working on variants of a product in different Projects where the common base can be shared. This chapter will show you briefly the Reusability capabilities of IRQA®.

Component modes

There are different modes to work with Components.

Share

When a Component is shared, its contents can not be changed by other Projects. The Component is managed by the publishing Project. Changes in the Publishing Project will be notified to Projects that are using the Component. These Projects can then decide if they want to use the new version of the Component or not.

Copy Link

In this mode a Component's contents can be changed by other Projects. New versions can be published by the Project who created the new version of the Component. Other Projects using the Component will receive a notice of the publishing of the new Component version.

Copy

In this mode a complete redundant copy of the published Component is made in another Project and will live its own life completely independent from the original published Component. So the Component and its copy in the different Projects have no interaction.

Component creation

A component can be created as part of the project management in IRQA® and selects contents based on the Structuring of the Project (See Structuring and navigation) by selecting Blocks, Domains or from already existing Components. From the list of, for example, Requirements can individual elements be selected to be part of the Component.
Components can contain several types of elements, besides Requirements it can also contain associated Services and/or Test scenario's, their relationships and their belonging Attributes.

Reporting

Besides information stored in a Solution like IRQA® you oft need documents that contain a certain part of the information stored. These documents are often used for communicating with, for instance, internal or external customers. The generation of these documents is called Reporting in IRQA. This chapter will show you briefly the reporting Capabilities within IRQA®.

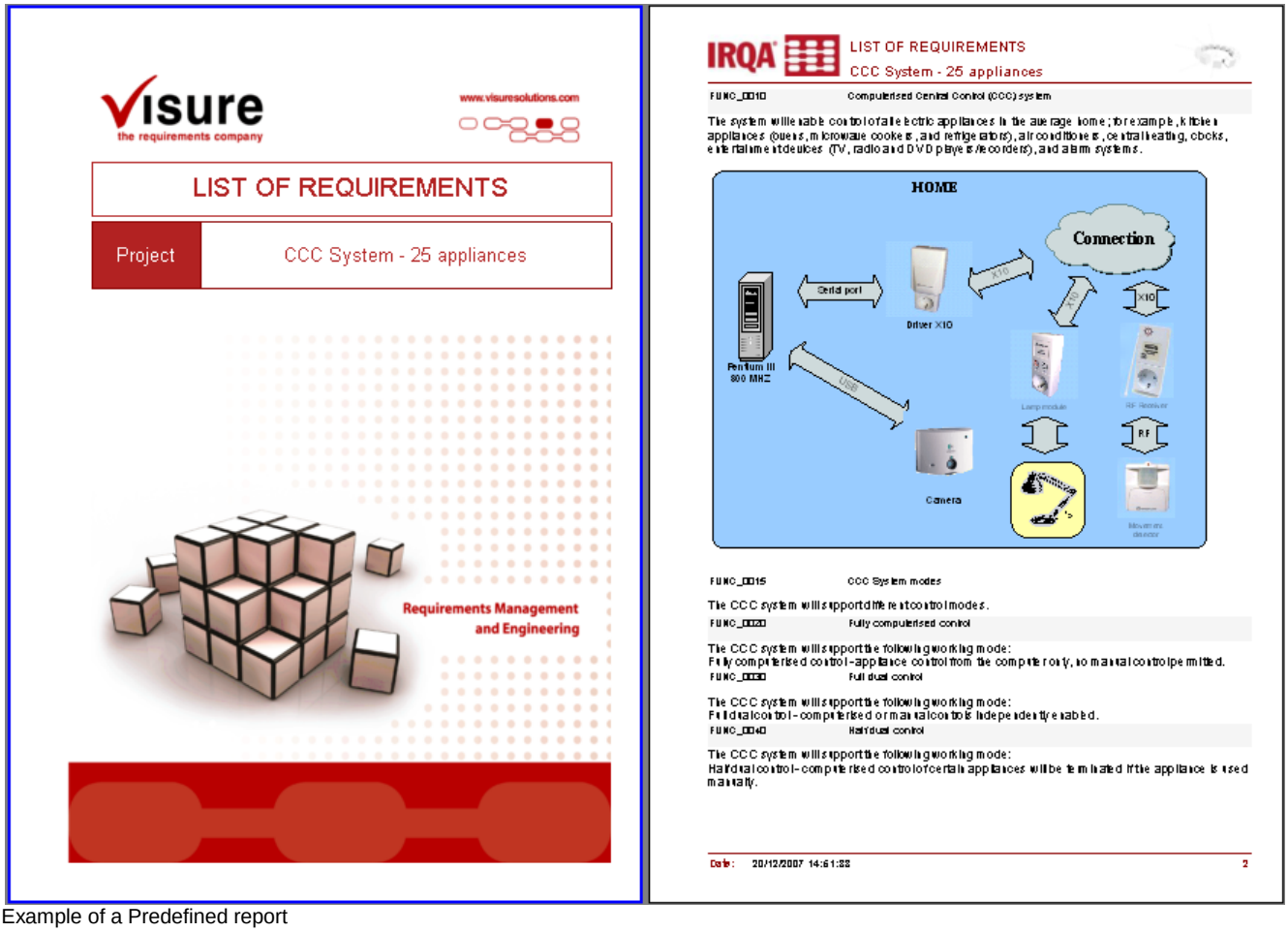
Predefined reports

These reports can be selected from a list and gives a report containing the contents according to the selected Predefined report option. These Predefined reports are standard within IRQA®.

Advanced reporting and Corporate reports

Advanced reporting means the use of the extension module Report Manager (See extended modules). With this module you can create your own Report templates with the looks and contents as you wish. The created Reports can be made available to others in the Corporate reports option within the IRQA® application. The Project administrator has to define a storage location for these Corporate Reports in the IRQA® Admin centre. Only when the storage location is defined will the option of Corporate Reports appear in the Reports menu within IRQA.

The Report Manager has to be purchased separately from the IRQA® Vendor (See extended modules) Also the IRQA® vendor can create reports for you to be used as Corporate reports within IRQA®.



Example of a Predefined report

Rights management

Important is the management of rights to access or read, write and deletion of contents. Rights are managed in the IRQA® Admin Centre by the Database administrator and the Project administrator. This chapter will show you a brief overview of the capabilities of IRQA® for managing these rights.

Database administrator

The Database administrator is the creator of Repositories and Projects in the IRQA® Admin centre. Within the Repositories he can create Users.

Project administrator

The Project administrator will manage the rights within the Project by defining first Access partitions, Groups and the addition of Repository users to the Groups in the IRQA® Admin centre. When allowed by the Database administrator he can also create new users in the Projects.
Then in the Rights management part the Project administrator will combine these elements to create the rights for users. By selecting which Group has Read or Write rights to what Access partition can the Project administrator decide for instance to let analysts have the rights to Read and Write system requirements and software developers only the rights to Read them. Groups that may not access certain parts of the content can also be forbidden to access these parts, so they even won't see the contents of these parts.

Role Based (Groups)

As to be expected from a modern solution, IRQA defines Users and Groups. Where groups will most likely used to define Roles. Users will be added to Groups by the project administrator.

Users

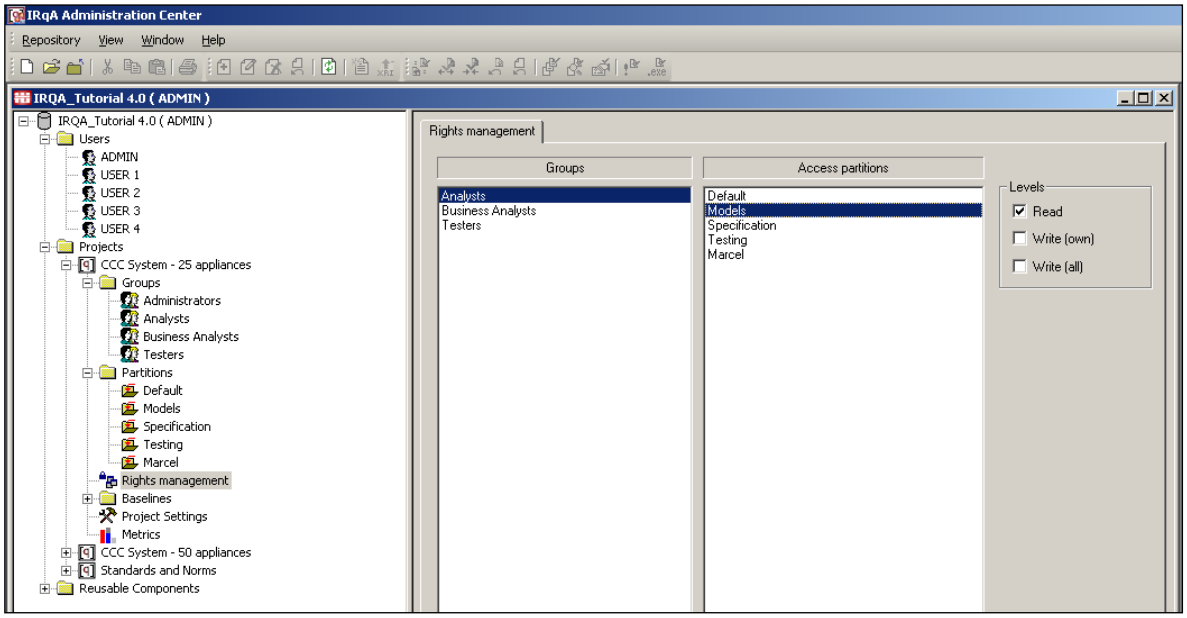
In IRQA Users can be defined within the tool itself, but is also possible to make use of existing Users in Active Directory, Windows domain or LDAP.

Access partitions

Access partitions are selections of the Projects contents and can be defined in different ways. The Project administrator (See Project administrator) can create Partitions based on element types, like Requirements, Services, Tests, Blocks etc..
So by defining an Access partition with, for example, Stakeholder Requirements it is only possible to access these Stakeholder Requirements for the Groups that are giving rights for access in the IRQA® Admin centre by the Project administrator.
Through the use of Access Partitions it is possible to fine-tune your rights management implementation as you can also make certain parts available to certain users and other parts not, even for the same type of elements. Also an often needed feature is the possibility to set different rights on the Attributes. All can Read the requirements, but only certain Groups will be allowed to change the Attribute values. These features are often lacking in other solutions.

Defining User rights

The rights of users will be defined by setting rights on the combination of Groups (Roles) and Access Partitions.



Example of the IRQA® Admin centre showing the rights management capabilities.

Setting available functionality

Sometimes it is undesirable that users can change structures or create, for example, their own attributes. By turning off or on toolbars or individual menu options, you can exactly define the available functionality per Role (Group). Also in a grow implementation scenario where is started with a limited functionality this will make the tool easier to learn as the number of options can be reduced.