Requirements Quality Analyzer for DOORS

Requirements Authoring Tool

User Guide

Margarita Salas, 16 2nd Floor
Innovation Center
LEGATEC Technology Park
28919 Leganés – Madrid
SPAIN – EU

http://www.reusecompany.com

Tel: (+34) 91 146 00 30
Fax: (+34) 91 680 98 26

contact@reusecompany.com

Table of revisions

Version	Date	Author	Changes
1.0	13th December 2012	TRC	Initial version
1.1	17th April 2013	TRC	Some corrections

Table of Contents

1. Introduction 5
2. Glossary 6
3. Installation 10
4. Initial Configuration 11
   - 4.1 Accessing the server by the first time 11
   - 4.2 Providing licenses to RQA Clients 13
   - 4.3 Configuring the Synchronizer 16
   - 4.4 Configuring email notifications 17
5. More Server functionality 18
   - 5.1 Managing RQA Objects in the Quality Database 18
6. Accessing the RQA for DOORS Client 19
   - 6.1 Providing a valid client license 20
   - 6.2 Logging in 22
7. Supported individual metrics 24
8. User's Roles 27
9. Creating the improvement cycle 29
   - 9.1 Managing the ontology 32
   - 9.2 Managing special sentences 33
10. RQA for Project Managers 35
    - 10.1 Assigning a set of Quality Functions 35
    - 10.2 Enabling the inconsistent units global metric 40
    - 10.3 Reviewing the quality of a module or project 42
11. RQA for Business Analysts 48
    - 11.1 Analyzing the quality of a requirement 48
    - 11.2 Analyzing the quality of an individual requirement 48
    - 11.3 Aggregating the quality of all of your requirements 54
    - 11.4 Looking for inconsistent units 55
    - 11.5 Looking for overlapping requirements 56

11.6 Assessing completeness for your requirements 63
12. Exiting the RQA Client 65
13. Accessing the RAT Client 66
13.1 Providing a valid client license 66
13.2 DOORS Log in 67
13.3 Adding a new requirement in DOORS 70
13.4 Editing a requirement in DOORS 74

1. Introduction

The Requirements Quality Analyzer (RQA) for DOORS is a product that enables defining, measuring, improving and managing the quality of requirements specifications in systems and software projects based on DOORS.

RQA for DOORS provides:

- Metrics based models for measuring and improving quality
- A formal semantic requirements meta-model
- Text based and *non-text* based measures
- Metrics for individual requirements, but also for whole specifications
- Calculations of quality that can be performed on-line
- Quality assessment of requirements on a regular basis (every night) or user defined
- Multi roles operability (Engineer, Project Manager, QA Manager)
- Customizable measures calculation engine
- Full support for the customer's domain representation
- Incorporation for Domain Specific Languages to the application

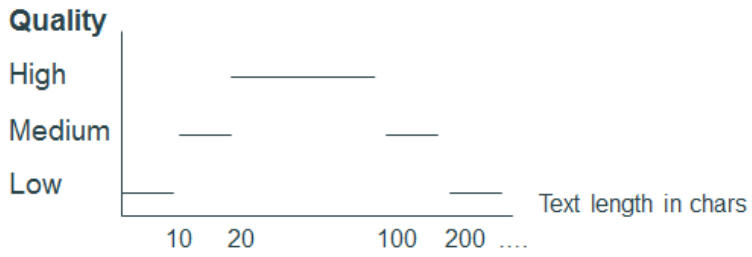
How does it work?

- Quality Assurance (QA) people define a set of quality functions for every metric
- For every requirement and every metric, a numerical value is generated
- Using a set of quality functions, every metric is qualified as of high, medium or low quality
- An aggregated quality value is generated for every requirement

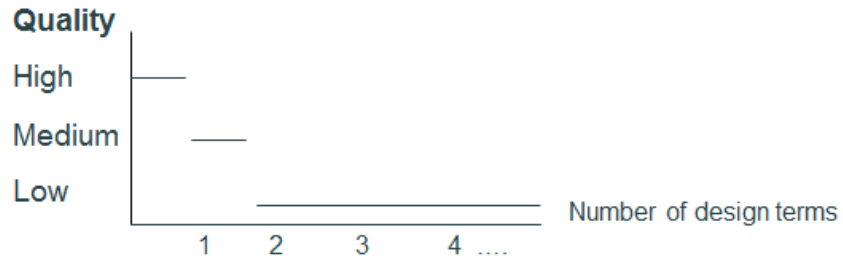
2. Glossary

This section presents a list of terms that are central to RQA for DOORS, along with their definitions and examples.

- **Metric:** every of the features analyzed for every requirement
  - Examples: number of ambiguous sentences, text length…
- **Individual metrics:** those metrics are analyzed requirement by requirement
- **Global metrics:** are analyzed for the whole set of requirements within a project
  - Examples: inconsistent units and overlapping tasks
- **Quality function:** represents how a quantitative quality value can be translated into a qualitative value.
  - Every individual metric is represented by means of a quality function
  - Quality functions for a given metric are customizable from a project to another
  - Quality functions for a given project/metric can evolve over the time
- Quality function examples:
  - Convex quality function:

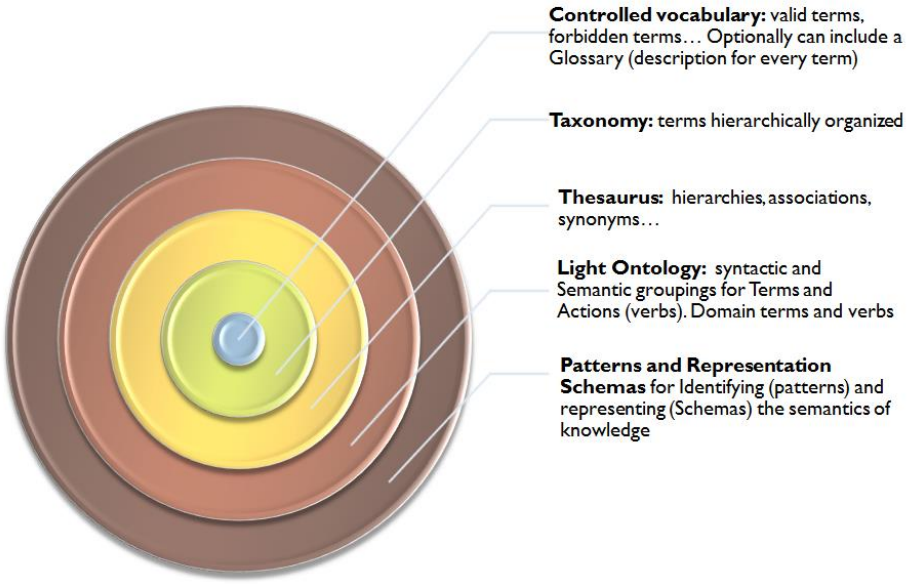


  - Decreasing quality function:



- Increasing quality function:

- **Maturity Level:** requirements quality improvement is a cyclic process. Quality functions could be fitted as maturity level improves (more and more metrics and more tighten quality functions)
- **Ontology:** stores all the information needed for the quality analysis:
  - Stores several sets of forbidden sentences (a.k.a. special terms): ambiguous sentences, flow sentences…
  - Stores a set of domain concepts and relationships
  - Stores a set of domain verbs and semantic clusters
  - Stores some Natural Language Processing (NLP) inference rules that allows RQA to 'understand' the content of every single requirement



- **Quality database:** stores all the quality information
  - Stores quality function configurations for every project/metric
  - Stores the results of the quality analysis
  - Stores an ontology for a given business area. The ontology is stored used a proprietary schema into this relational database
  - DOES NOT store the requirements themselves
- **DOORS Database:** stores
  - The requirements themselves
  - The privileges granted for DOORS and RQA users
  - Optionally stores the results of the quality analysis
- **RQA Server:**
  - Allows the initial configuration of RQA (see details)
  - Stores the license information for a set of RQA Clients (see details)
  - Allows RQA Admins to configure the RQA Synchronizer process (see details)

- A single server may analyze requirements quality from many DOORS databases

- **RQA Client:**
  - Allows Quality Assurance Team (QA) to customize how quality should be analyzed (see details)
  - Allows Business Analysts (BA) or Systems Engineers (SE) to check the results of the quality analysis (see details)
  - Allows Project Managers (PM) to check the quality configuration as well as the results of the quality analysis for a given project (see details)
- **RQA Synchronizer (a.k.a. Batch process):**
  - Allows all the time consuming tasks to be executed in a batch mode
  - Can be executed manually in the RQA Server or as a programmed task
  - See configuration details
- **Batch state:** represents the way the Synchronizer is going to work with every module
  - No batch process (without metrics): there're no quality functions defined for the module
  - No batch process (with metrics): there're metrics and quality functions defined for the module, but you can only analyze quality manually using RQA Client
  - Batch process: there're metrics and quality functions defined for the module; furthermore, you can analyze quality both, with RQA Client and with Synchronizer

3. Installation

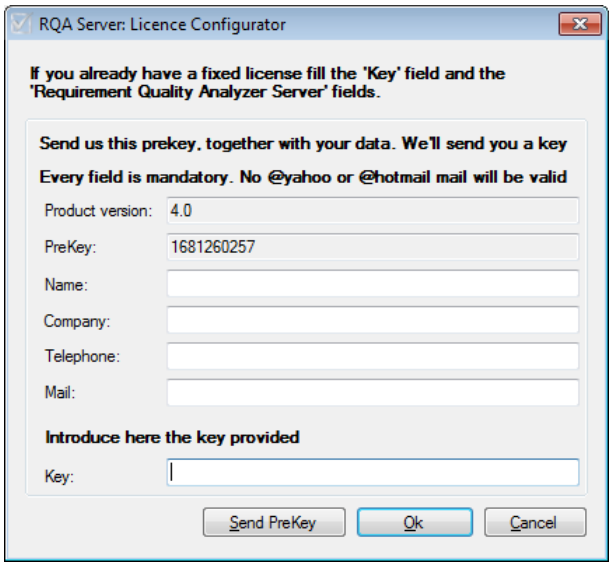
The installation procedure for both the RQA Server and the RQA Clients is described in detailed in the ***RQA for DOORS Installation Guide***, provided with this product.

4. Initial Configuration

4.1 Accessing the server by the first time

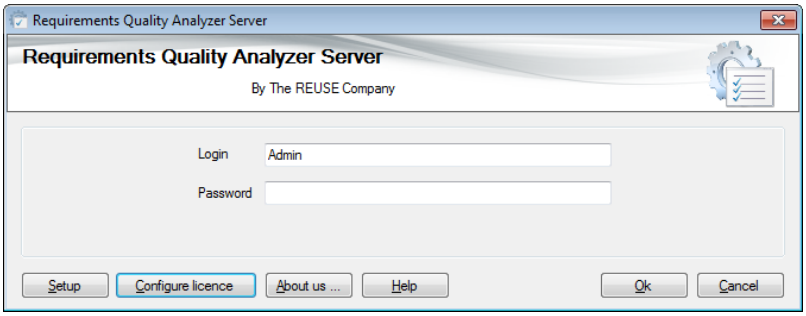
Once the server is installed and configured, it is configured to start the RQA service when Windows starts. It is also executed automatically when the installation process ends. If for any reason the RQA Server has not started or has been stopped, the RQA Server can also be started using the Windows Start Menu.

During the first execution, a valid **key** will be required

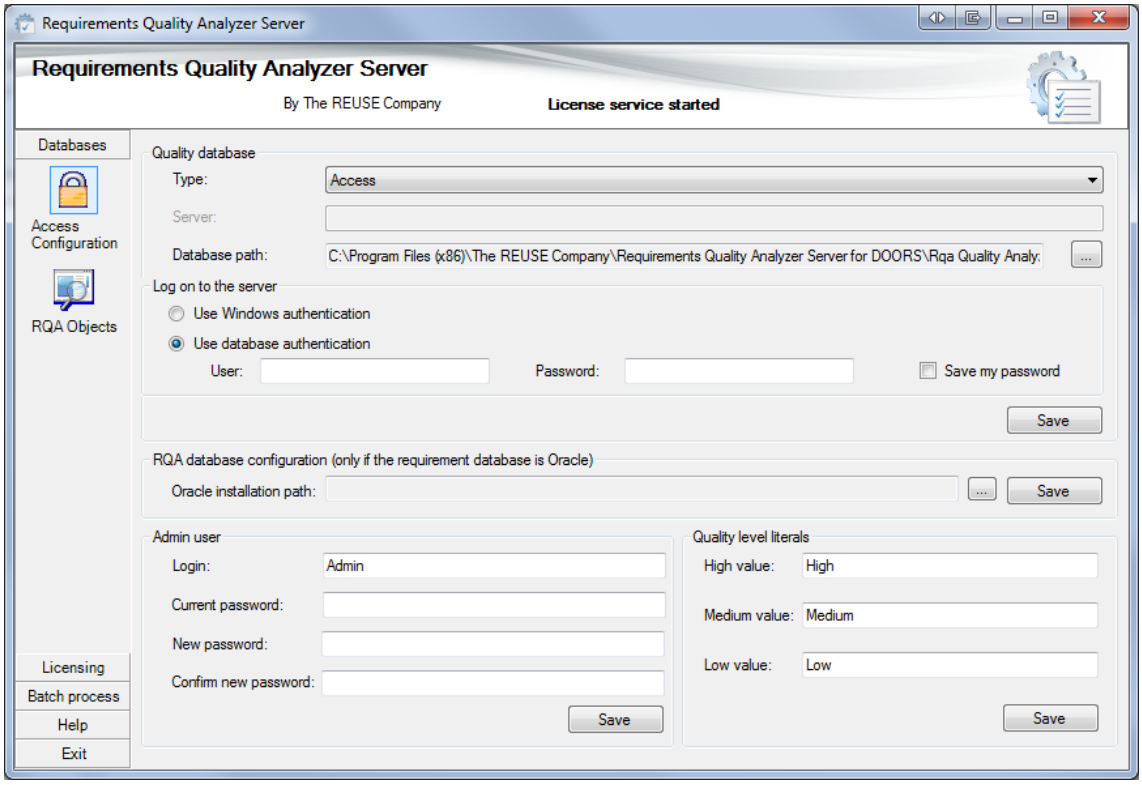


Credentials for the first access to the tool are the following:

- MS Access Quality Databases:
  - Login: Admin
  - Password: <empty> (no chars)
- SQL Server Quality Databases:
  - Login: Admin
  - Password : <empty> (no chars)
  - Click *Setup* to select the Quality Database for the first usage of the Server



Please click on the **Setup** button to configure. For more help, see the RQA for DOORS Installation Guide



- RQA database configuration (only if the requirement database is Oracle): this option does not make sense with DOORS, only if you want to connect RQA with other requirements management products.
- Admin User: configure the admin user credentials.
- Quality level literals: configure the three quality level literals.

4.2 Providing licenses to RQA Clients

In the main RQA Server window:

1. Select the **Licensing** tab on the left

2. RQA Clients licenses are:

   - Nominal licenses based on the Visure Requirements licensing system.

   - They are configured on the server in this **Licensing** tab:

     a. Select your hard disk

     b. Click on *Get Prekey* and send this number to your product representative

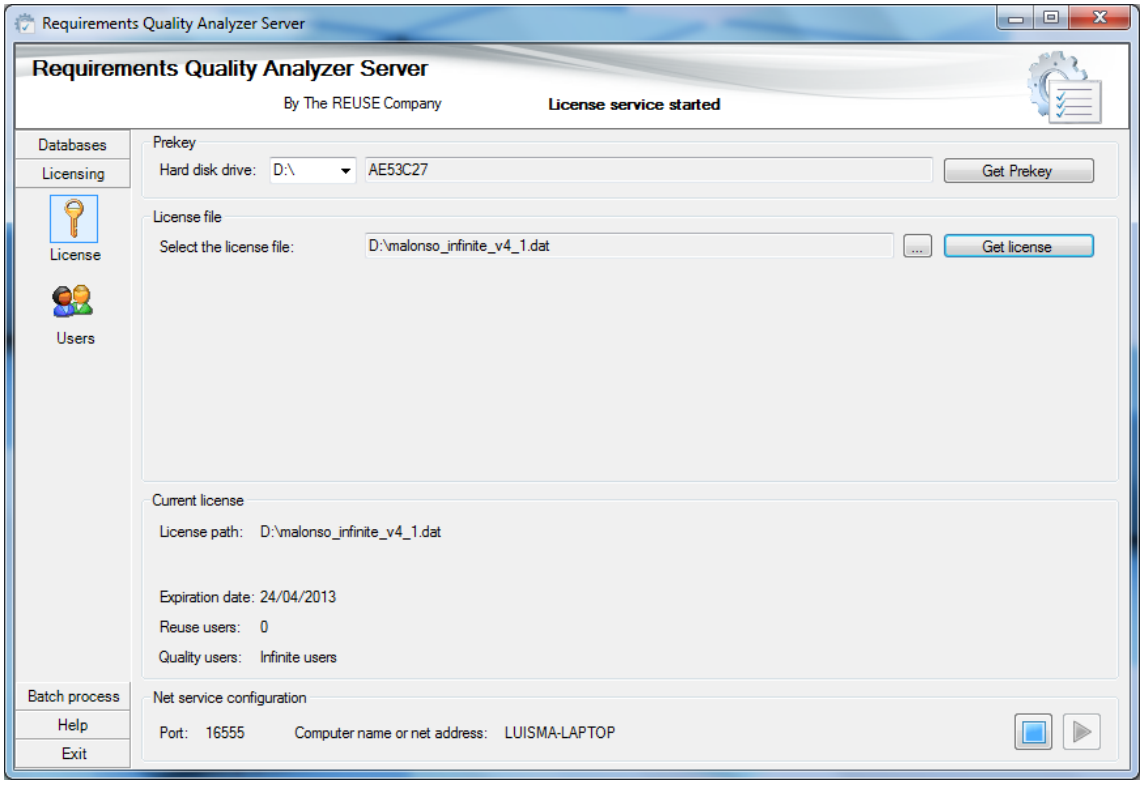
     c. Store the license file provided by the representative in the selected hard disk

     d. Select the license file and click on *Get License*

     e. License information is shown in the *Current license* group box

     f. Provide a set of valid user accounts for the clients (see next slide)

     g. Stop and re-start the process if needed

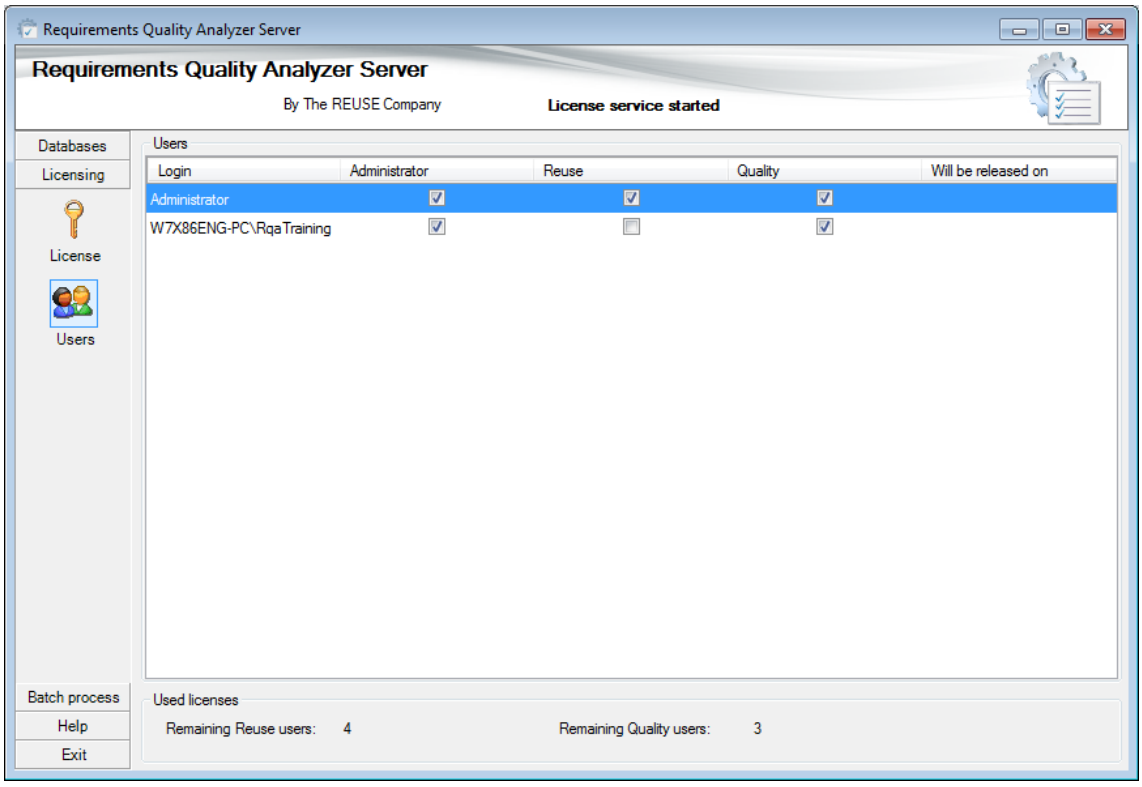


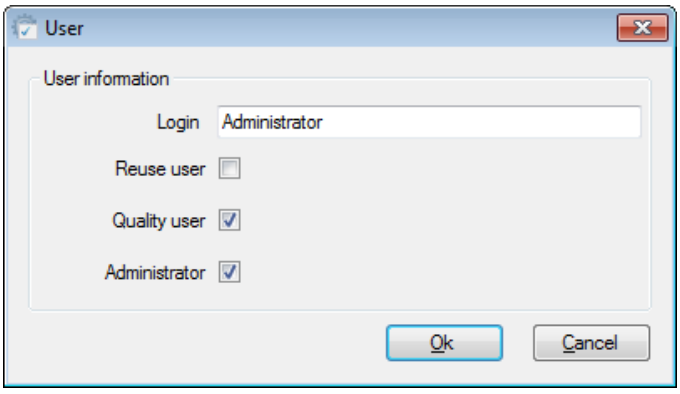
- Clients licensing is based on DOORS accounts
- Go to *Users* tab and provide a set of quality granted accounts

1. Right click on the list and then *Add User*
2. Type the name of the DOORS account (case sensitive)
3. Check *Quality user*
4. Check *Administrator* if you want this DOORS user to be able to:
    i. Modify quality configuration
    ii. See and aggregate the quality analysis for all the requirements, even those not created/modified by himself
5. Click *Ok*

6. Now you can grant as Admin or remove a user (to leave a license available) using the contextual menu

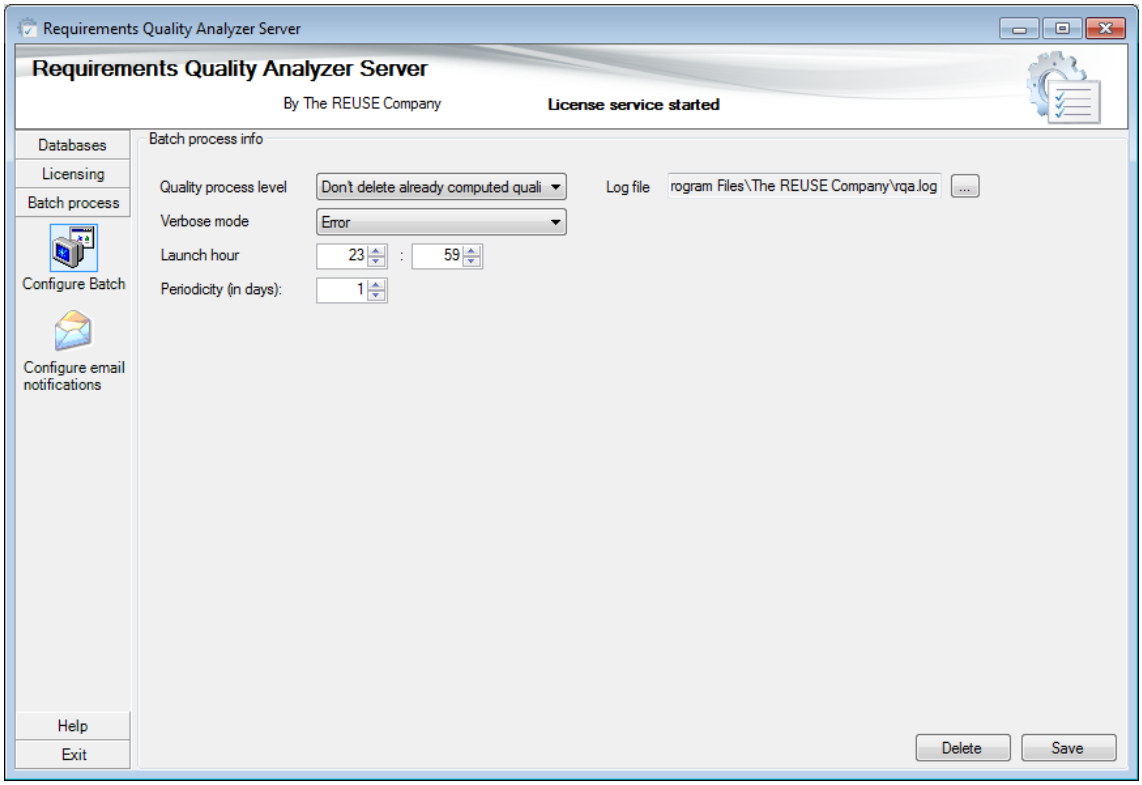
- Once a user has been removed, *it cannot be added again for 1 week*, the next time the user will be available to be added to the valid user is the date under the "Will be released" column.





4.3 Configuring the Synchronizer

For the configuration of the synchronizer, please select the *Batch process* tab on the left and click on the Configure Batch icon. Then choose the appropriate values for the process parameters:

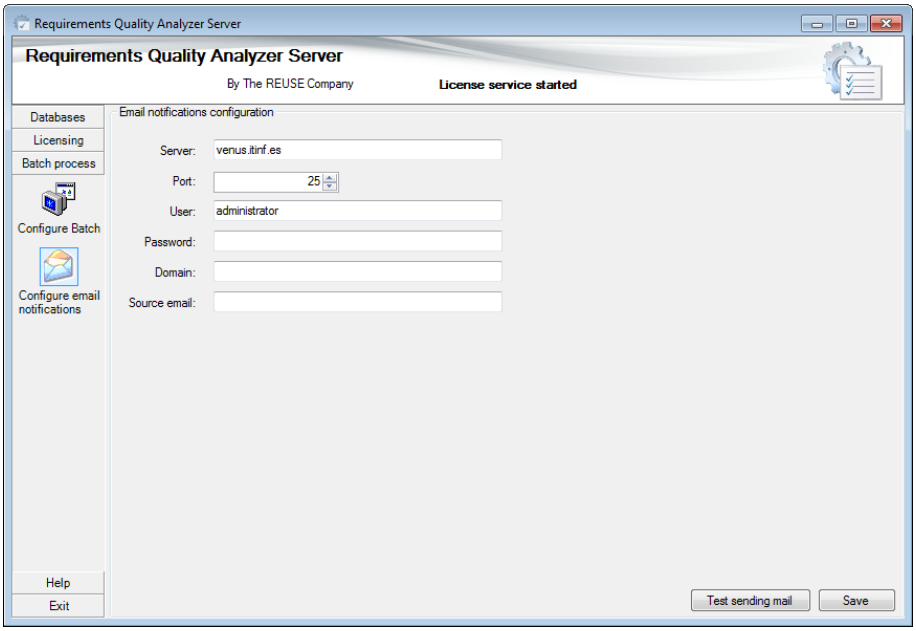


1. Quality process level:
    a. Don't delete already computed quality: should be the default value in order to avoid analyzing the same requirement over and over even if it has not been modified
    b. Delete already computed quality: forces the quality process to be executed for every requirement even if the text of the requirement has not been updated. Use this configuration if Quality Functions have been updated and you want to force the quality to be analyzed again
2. Verbose mode: represents the amount of information stored in the log file. Error level should be enough almost in every case
3. Launch hour: from 0 to 23 hours and 0 to 59 minutes, representing the exact time when the task is going to be executed automatically
4. Periodicity: this task could be executed every day (type 1), every week (type 7)…
5. Log file: the path where the RQA log is stored by the Synchronizer

6. Click on *Delete* if you want to delete the Windows Scheduled task

4.4 Configuring email notifications

For the configuration of email notifications, please select the *Batch process* tab on the left and click on the Configure email notifications icon. Then choose the appropriate values for the process parameters:



1. Server: The IP address or name of the mail server
2. Port: Port number of the mail server
3. User: username for the mail server
4. Password: password for the above username
5. Domain: the domain where the username belongs (e.g. Windows domain)
6. Source email: the email address of the sender

User can send a test mail to check that values for the process parameters are correct.

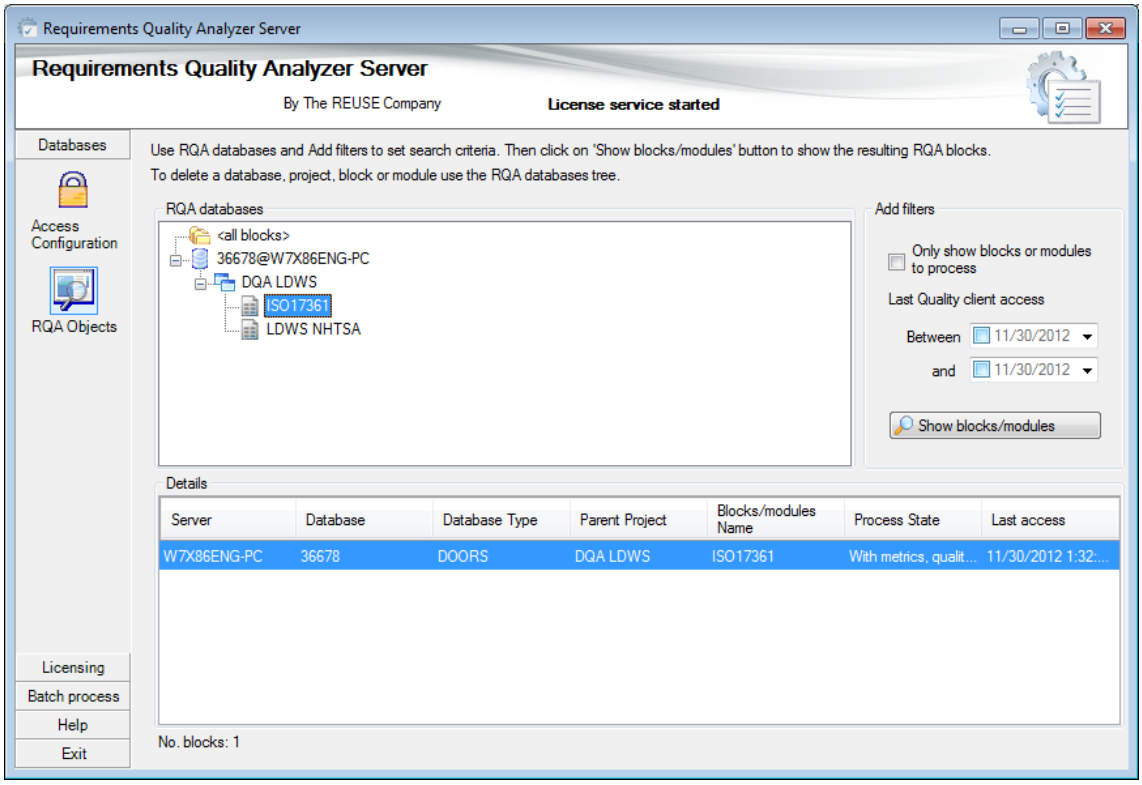
5. More Server functionality

5.1 Managing RQA Objects in the Quality Database

There are several cases where older quality information could remain in a Quality Database and must be removed. These cases may occur:

- when a Quality Database is reused from one business area to another
- when a project/module has been removed from the DOORS database

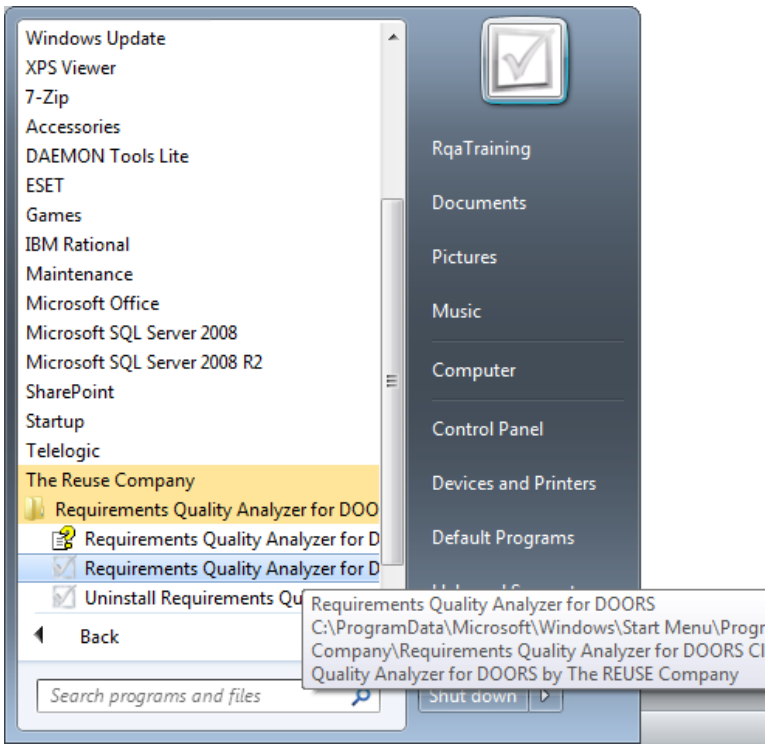
The *RQA Objects* tab shows all the information stored in the current Quality Database



1. Click on *Show blocks/ modules* to check which modules have associated Quality Functions
2. Right click on the *RQA Databases* list and click on *Delete project* if you want to remove a project from the Quality Database
3. Even if you remove a project from the Quality Database, you could analyze it again with RQA and the quality information will be generated and stored again

6. Accessing the RQA for DOORS Client

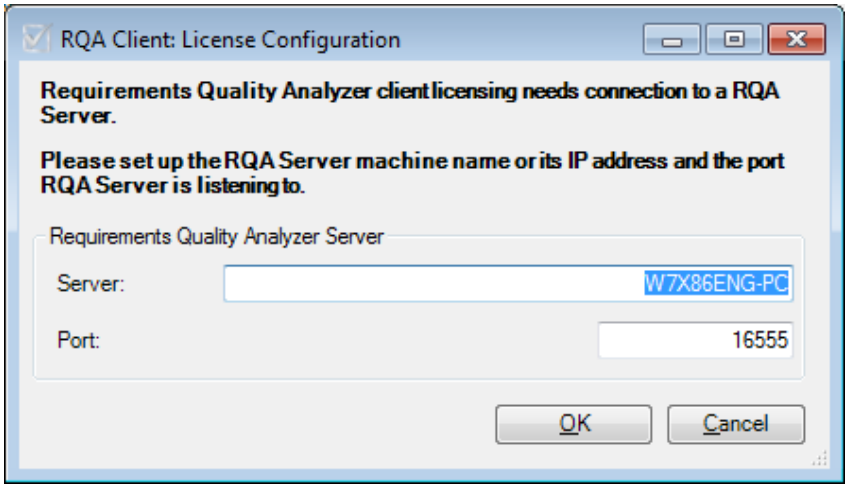
The RQA Client can be launched using the Windows Start Menu



6.1 Providing a valid client license

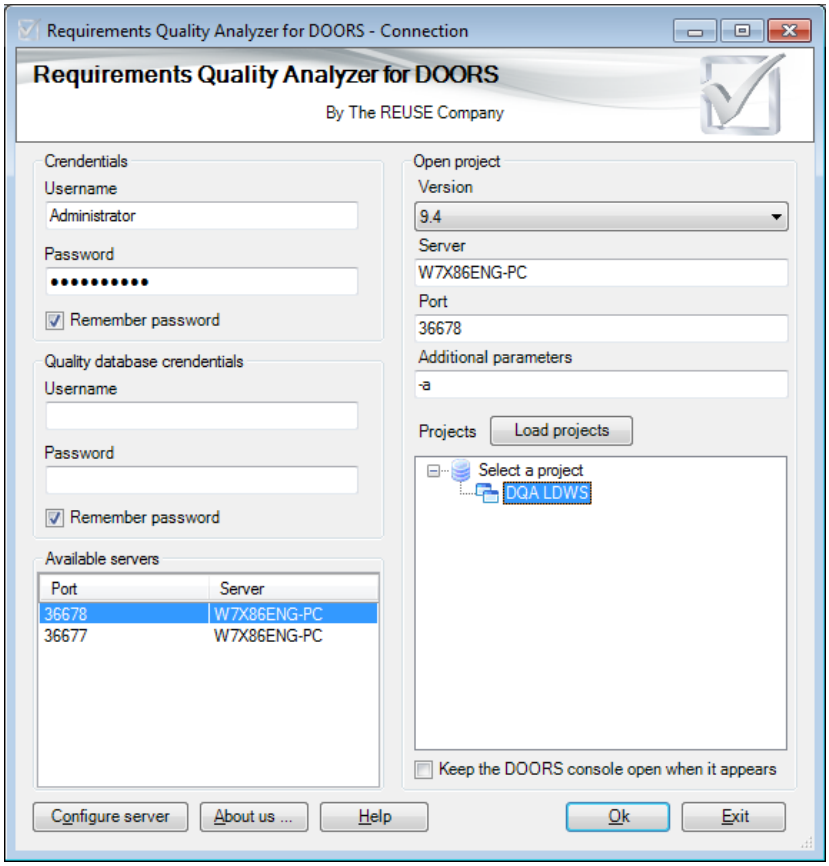
RQA Client licenses are:

- Nominal license, it must have been setup on the server before. So in the clients the only thing to do is indicating the RQA Server name or IP address and the port:



1. Click on *Configure server*
2. Type the server name and the service port in the RQA server
    a. Default port value: 16555
    b. Start the RQA server and click on *Licensing* to see this values
3. Click *OK* in the License Configuration dialog

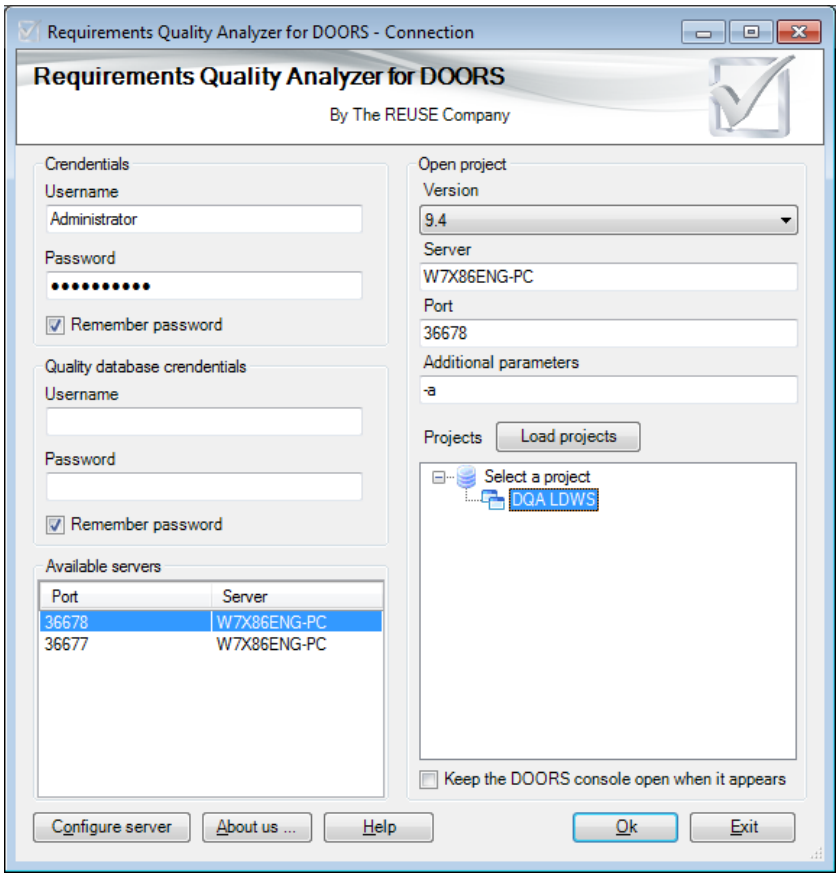
- Fixed (local) license:



1. Click on *Configure server*
2. Check *Fixed License*
3. Fill all the contact information and send the PreKey to a tool representative
4. When you receive a valid key, paste the provided key
5. Click on *Validate key*

6.2 Logging in

RQA accounts are based on currently existing DOORS accounts. In the log-in window of the RQA Client:



1. Type your credentials: Username and Password
2. Type you quality database credentials: Username and Password.
    a. Only necessary if the RQA Server administrator has not saved the Quality Database credentials to avoid the users to write them live every time they connect to a DOORS project from RQA Client
3. Select the version of the DOORS database you want to connect to
    a. Only available DOORS clients versions are shown
4. Type the DOORS Server, Port and Additional parameters (see your DOORS shortcut if needed)

5. Select the DOORS project you want to work with
6. Click on *OK* to see/configure the quality of the selected project and its modules
7. The list of Available servers represents the set of Version + Server + Port + Additional parameters you have logged in in the past

7. Supported individual metrics

The Requirements Quality Analyzer for DOORS assesses the quality of a DOORS repository by involving Natural Language Processing techniques that extract, out of every individual object, information about the metrics listed in the table below.

In the second column you can find the justification for each metric, while in the third column the default sentences included in each of the metric, as well as the way to customize the metric according to customer needs.

Metric	Description	Default content
Acronyms	Avoid using acronyms which are not declared into the ontology. In order for all stakeholders to fully understand the acronyms used in your requirements, they must be declared in the ontology. Please avoid this acronym or contact your domain architect in order to insert this acronym into the ontology	List of acronyms in the ontology (included in the ontology terms)
Ambiguity	Ambiguous sentences make the requirement difficult to understand, and can provoke other stakeholders to understand something different than the idea initially planned by the author of the requirement. Ambiguous sentences are difficult to understand, as two stakeholders may understand different needs in the same requirement	Ambiguous words/expressions to be avoided: adequate, approximately, approximate, as a maximum, as a minimum, maximum, minimum, minimal, as appropriate, appropriate, reasonable, as possible, as required, bad, be able to, be capable of, best practices, best possible, better, capability of, capability to, close quickly, easy, easy to, easy to use, effective, efficient, fast, flexible, good, high performance, high speed, if practical, improved, maximize, minimize, medium-sized, optimize, optimal, optimum, nominal, normal, typical, typically, useable, suitable, not limited to, provide for, prompt, quick, quickly, rapid, reliable, routine, safe, slow, sufficient, sufficiently, timely, too, user friendly, user-friendly, versatile, worst, at least, enough, clearly, based on, some, any, several, many, many of, a lot of, a few, few, about, very nearly, manage, about, easily, close to, small, significant, vague, flexible, ancillary, relevant, routine, common, generic, customary, so far as is possible, as far as possible, as little as possible, as much as possible, if it should prove necessary, as necessary, necessary, all, any, both
Boilerplates matching	The structure (grammar) of your requirements must fulfill one of the agreed grammars (boilerplates). Doing so will increase the readability of the requirement and you will ensure that the automatic tool will perfectly understand the requirement.	Number of boilerplates matched
Chars between punctuation	If you write long sentences without punctuation marks, the requirement will be difficult to understand. You must introduce more punctuation marks in order to get more readable requirements.	Number of counted characters between punctuation marks.

Conditional	A requirement must be written in an assertive way. Whether a requirement is mandatory or not must be indicated as an attribute, and not by using conditional expressions.	Examples of conditional words/expressions: May be, May be not, Can, Can't, Cannot, Could, Couldn't, Could not, Should, Shouldn't, Should not, Ought to, Oughtn't, Ought not, Would
Connectors	Using too many connectors, in most of the cases, may mean that either your requirement is over specified or you are mixing two or more different needs into the same requirement.	Examples of connectors: and, and / or, and/or, or, as well as, but also, however, whether, meanwhile, whereas, on the other hand, otherwise, /
Dependencies	An object with too many dependences could be difficult to understand	Number of in and out links
Design Sentences	Top level requirements or user requirements must be focused on a necessity (asking "what") instead of focusing on the solution to that necessity (asking "how"). By involving the concepts included in this list, you are clearly including the solution of the need (design).	Examples of terms to be avoided: .NET, ADSL, Array, Bit, C, C#, cache, cgi, check box, checkbox, COM, combo box, combobox, CORBA, Cursor, design, diagram, DLL, exception, FTP, Hash, HTTP, installation, IP, JAVA, kb, Kilobit, Kilobyte, Loop, mb, megabit, megabyte, object, OCX, parameter, PHP, source code, testing, thread, visual basic
Domain Concepts	Involving too many domain concepts means that the requirement is either over-specified or even different needs are involved in the same requirement. Try to reduce the number of domain concepts involved in the requirement.	List of concepts defined in the ontology
Domain Verbs	A requirement may involve a set of different concepts and verbs. Nevertheless, some of them, the most relevant, must be included into the ontology of your business unit. In case you consider that the verbs you are using in your requirement should be included into the ontology, please contact your domain architect.	List of verbs defined in the ontology
Flow Sentences	A requirement must avoid flow sentences or pseudocode in order to be more focused on the problem (the need) and not on the solution.	Examples of flow sentences: although, as well as, but, Else, except, if, if not, then, unless, when, While
Imperatives	Every requirement must have, at least, one verb in imperative mode. This is the way the assertiveness of the requirement is expressed.	Shall, Will, Must, Won't, Shan't, Mustn't
Implicitness	Implicit sentences, i.e. those that are not 100% explicit due to the use of pronouns, can make the requirement difficult to understand under certain circumstances. Please try to avoid pronouns in order for all stakeholders not to have problems identifying the meaning of those pronouns and enhancing readability.	Examples of pronouns: he, her, him, his, I, it, my, our, she, their, them, they, us, we, you, your, this, that
Incompleteness	Some sentences such as "etc.", "not limited to"… clearly demonstrate that the requirement has not a clear scope. Try to avoid this kind of expressions, since they make an object non-atomic	Examples of incomplete sentences: among others, and so on, as a minimum, etc., etcetera, further, not defined, not determined, not limited to, shall be included but not limited to, tbc, tbd, tbs, to be determined, but not limited to, e.g., eg, example, such as, (,)
In-links	In-links must be used according to the guidelines and policies of your organization.	Number of in-links
Negative Sentences	Including more than one negative expression in the same requirement can make the requirement difficult to understand. This metric is not activated when just one negative sentence is used, but when more than one is used at the same time.	Examples of negative terms: Cannot, Does Not, Doesn't, Never, No, Nobody, Non, None, Nor, Not, Nothing, Won't, Shan't, Mustn't, Couldn't, Could Not, Shouldn't, Should Not, Oughtn't, Ought Not, Can't, Cannot, May Be Not

Nested levels	The hierarchical nesting structure of your specification includes too many levels. This can lead to complex and difficult-to-read specifications and must be avoided.	Hierarchy of Requirements
Out-links	Out-links must be used according to the guidelines and policies of your organization.	Number of out-links
Paragraphs	An object should not be written using too many paragraphs	Criterion used: Number of paragraphs in an object
Passive voice	Passive voice could be, in some times, difficult to understand and it is less assertive than active voice since the object which is performing an action is located at the beginning of the sentence.	Number of verbs in passive voice
Rationale	Any kind of justification within the requirement makes it difficult to understand. These justifications should be out of the requirement text but must reference it with a link or a trace.	Examples of expressions that imply rationale: in order to, so that, thus allowing
Readability	A low readability mark for a requirement means that its structure could be somehow simplified in order to ensure that all the stakeholders understand exactly the same out of the text of the requirement.	Criterion used: Paragraph readability = 0,39 * (words / sentences) + 11.8 * (syllables / words) – 15.59 [Flesch Grade Level Readability Formula]
Speculative Sentences	Using speculative sentences suggests that the real necessity of the requirement is not clear for the reader. Please, try to avoid using this kind of sentences.	Examples of speculative expressions: almost always, commonly, frequently, generally, normally, often, optionally, perhaps, probably, rarely, typically, usually, eventually, at last, almost, always
Subjectivity	The point of view of the author must not be expressed in the requirement. Instead of this, the author must focus on the objective need of the requirement.	Examples of expressions of subjectivity: I think, imho, imo, in my humble opinion, in my opinion
Text Length (chars)	The length of a requirement must be controlled in order to avoid over-specification, several needs into the same requirement or even redundant information.	Number of characters in a requirement.
Text Length (paragraphs)	The length of a requirement must be controlled in order to avoid over-specification, several needs into the same requirement or even redundant information.	Number of paragraphs in a requirement.
Unclassified concepts	In order to enhance the readability of your requirement, you must use only those terms already managed by the system. In case you consider that the concepts you are using in your requirement should be included into the ontology, please contact your domain architect.	Number of concepts not recognized by the system
Unclassified Verbs	In order to enhance the readability of your requirement, you must use only those verbs already managed by the system. In case you consider that the verbs you are using in your requirement should be included into the ontology, please contact your domain architect.	Number of concepts not recognized by the system
Volatility	Volatility in a requirement or in a requirements specification is not directly related to the quality of this requirement or specification. Nevertheless, the risk derived from a volatile requirement is higher, and must be controlled carefully.	

8. User's Roles

SE Expert:

- Author Requirements

Quality Assurance:

- Adjusts metrics and quality functions.
- How can metrics and quality functions be adjusted to carry on with the improvement process?

Quality Control:

- Executes assessments for individual requirements and global specifications.
- Is quality evolving as expected?

Project Manager / Quality Manager:

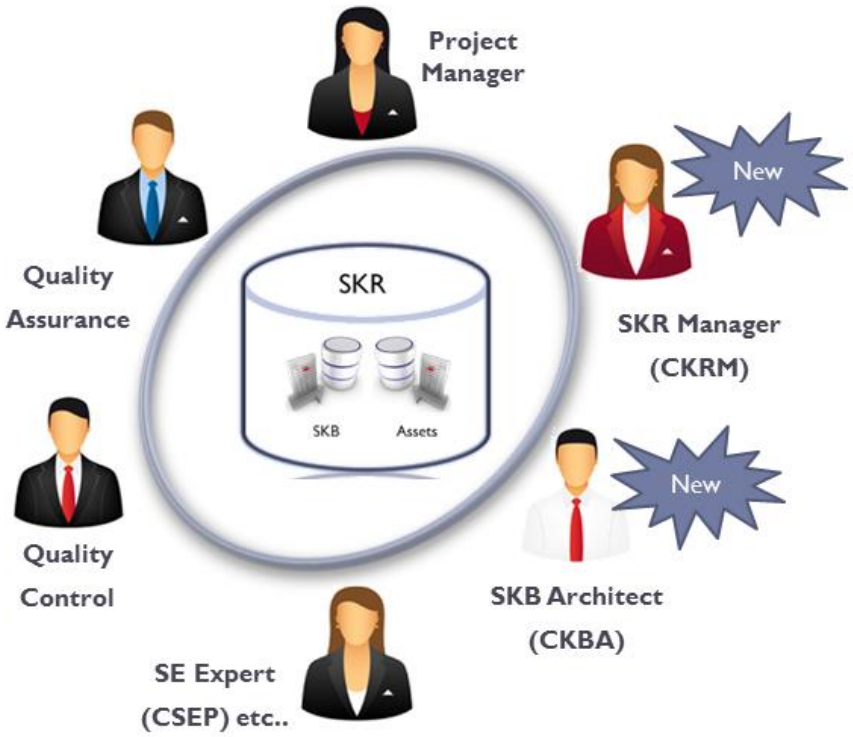
- Do my project requirements have the right quality?
- Do our teams need additional training?

SKB Architect:

- Evolves the requirements knowledge model.

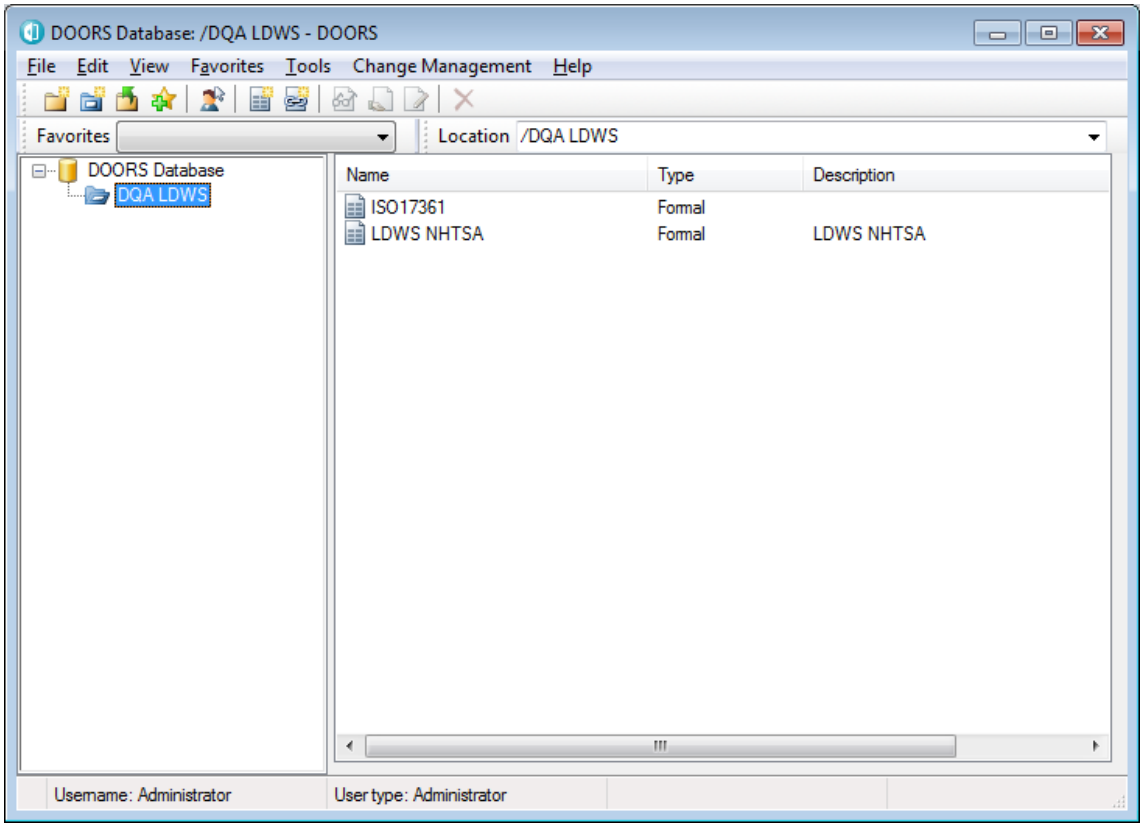
SKR Manager:

- Leader of the Knowledge Repository


Project
Manager
New
Quality
Assurance
SKR
SKR Manager
(CKRM)
SKB
Assets
New
Quality
Control
SKB Architect
(CKBA)
SE Expert
(CSEP) etc..

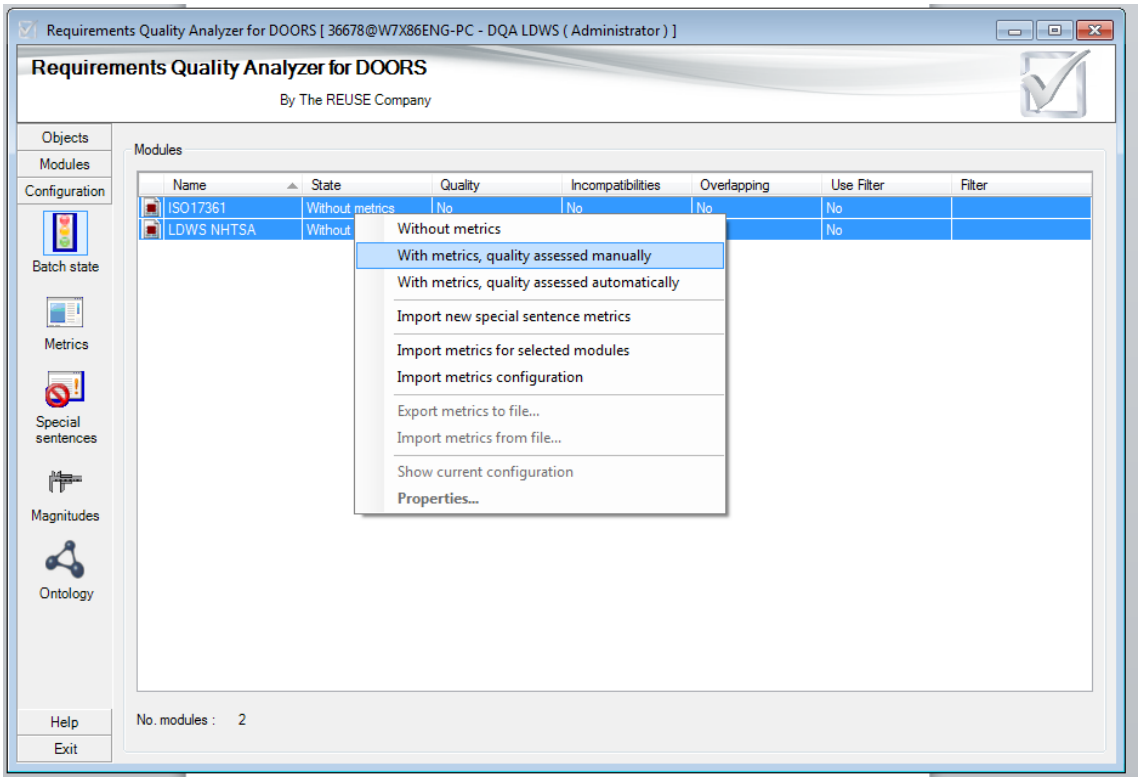
9. Creating the improvement cycle

Use DOORS to create several quality folders and modules. These will not contain real requirements, but the Quality Functions. Create one folder for every Maturity Level and create one module for every type of requirements.



Now use the RQA Client to provide quality functions for every Maturity Level and every different type of requirements, following the next steps:

1. Log-in into RQA Client
2. Move to *Configuration/Batch State*
3. Select a module, right click and click on *No batch process (with metrics)*

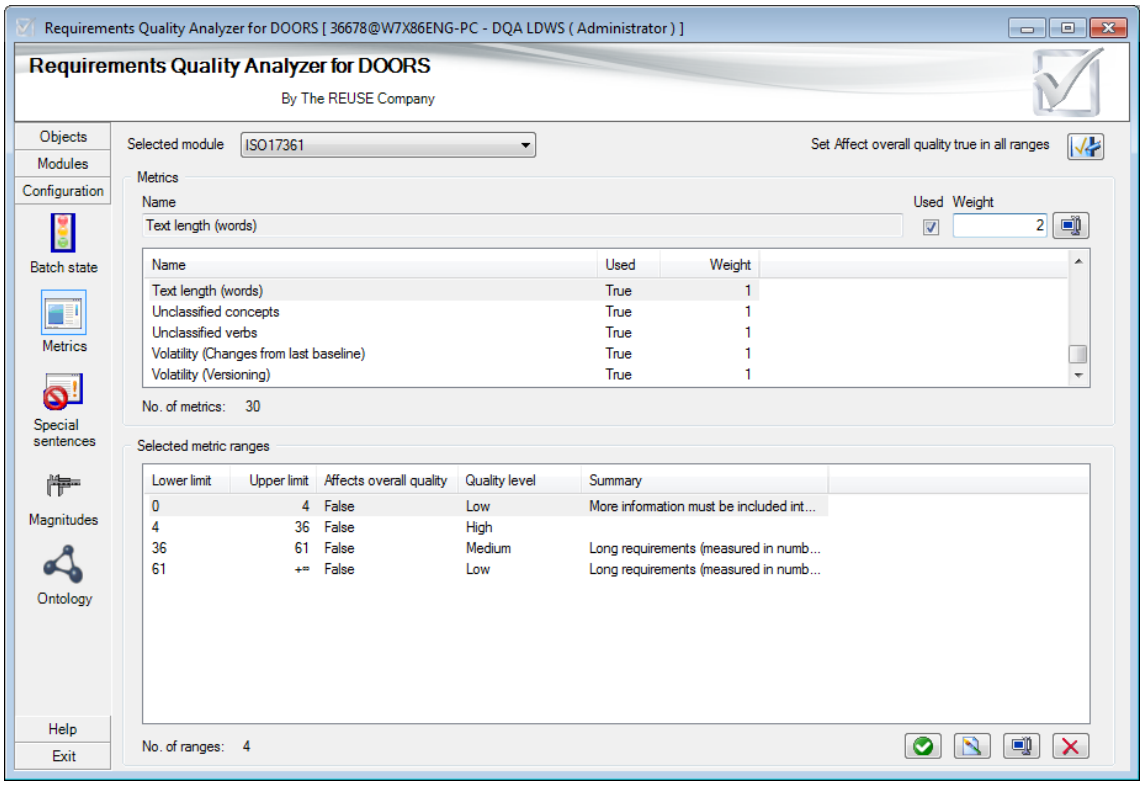


Click now to the *Metrics* icon, on the *Configuration* tab:

1. Select the type of requirements from the combo box on the top of the window
2. Select which metrics you want to involve in the current ML
    a. By checking the *Used* check box
3. Select the weight you want to assign to this metric
    a. By writing a number from 0 to ∞
4. Save every Metric configuration, one by one, with the *Update* button

5. Use the buttons below if you want to add, edit or remove a range for the given Metric

6. Use the *Check* button to look for overlapped intervals in the Quality Function



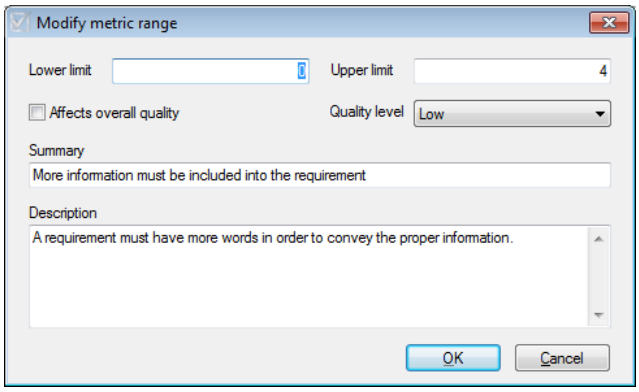
For every involved Metric, the Quality Function must be defined. Every range must match on low, medium or high quality. Provide ranges as

- [lower-limit , upper-limit) or
- [lower,-limit upper-limit [

where

- Lower-limit should include the endpoint
- Upper-limit should exclude the endpoint

Please check *Affect overall quality* if a low quality mark for this Metric involves low quality mark for the whole requirement.



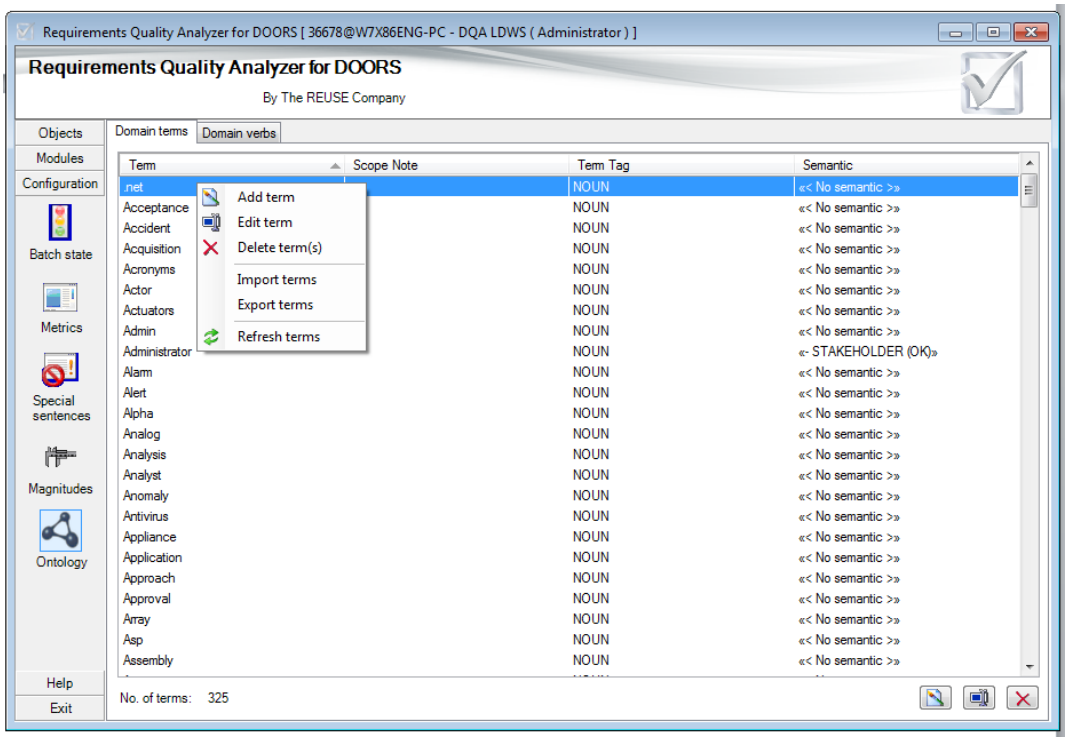
9.1 Managing the ontology

The proper way to manage the Ontology for the improvement process is by using the knowledgeMANAGER product. Nevertheless, the RQA Client also allows a shallow management of the ontology, including lists of terms, but not relationships or semantics. Please keep in mind that full ontologies are not mandatory, but help very much in the global metrics.

For basic ontology management, follow the next steps:

1. Move to *Configuration/Ontology*. Domain Terms and Domain Verbs can be displayed and managed through this window.
2. Use the buttons on the bottom part of the window to Add, Edit or Delete a Term or a verb

3. These lists may be exported/imported to/from *csv* files using the contextual menu.



9.2 Managing special sentences

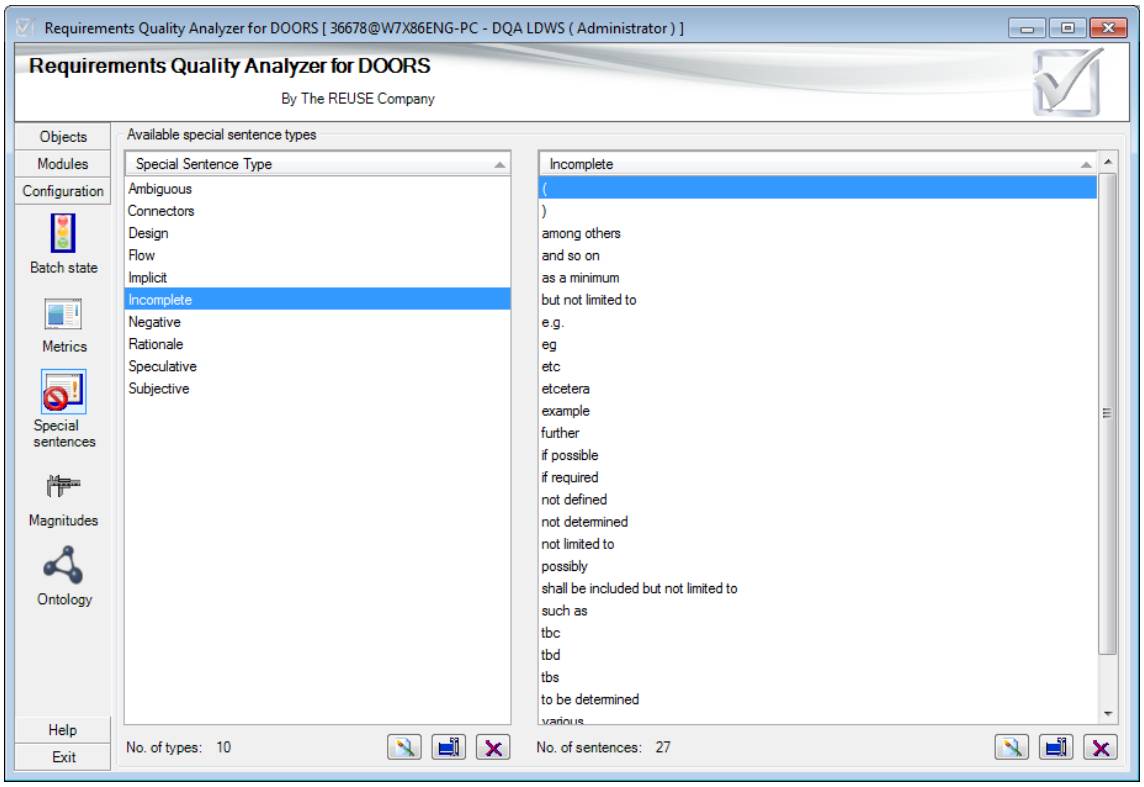
Special Sentences (Forbidden Sentences) represent a part of the ontology and cannot be managed using the knowledgeMANAGER. In order to manage special sentences, please:

1. Move to *Configuration/Special Sentences*
2. Click on one of the lists
3. Use the buttons on the bottom part of the window to Add, Edit or Delete a Sentence

4. These lists may be exported/imported to/from *csv* using the contextual menu

5. The lists can be extended with custom lists of sentences.

   - This additional sentences lists can be also edited or deleted
   - But the original sentence lists cannot be edited or deleted, the user is only allowed to manage the sentences that those original sentences lists hold.

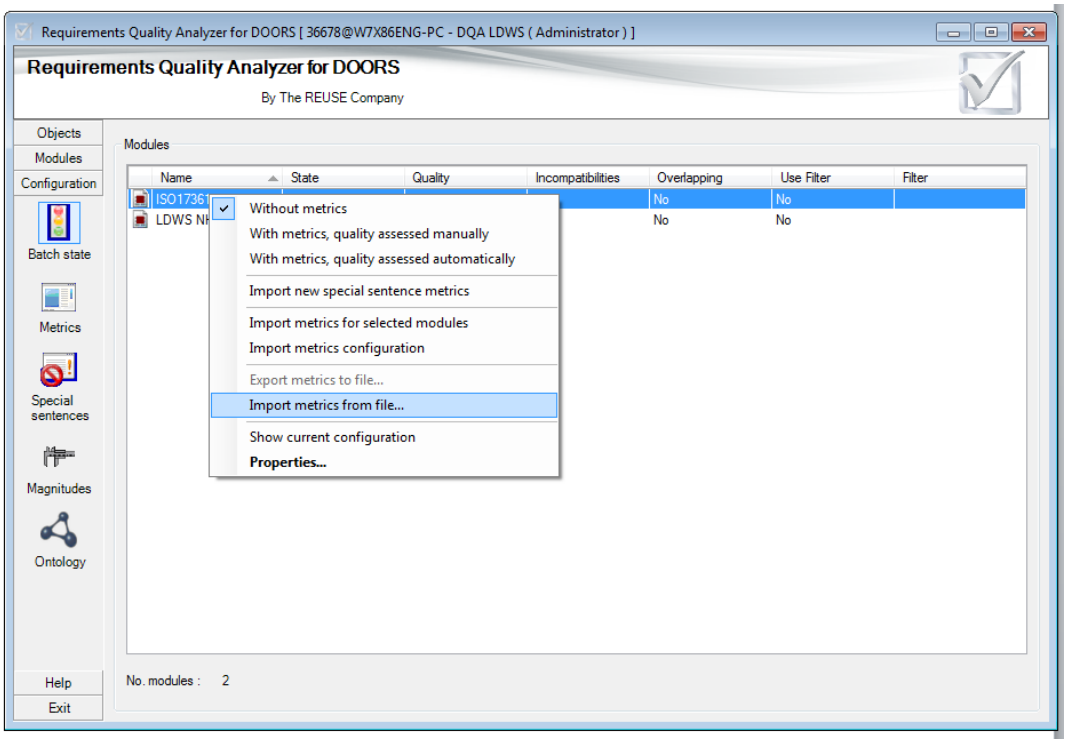


10. RQA for Project Managers

10.1 Assigning a set of Quality Functions

The Project Manager should be in charge of assigning a set of Quality Functions to each corresponding projects and then be able to visualize the quality of the project. In order to do so, the Project Manager should:

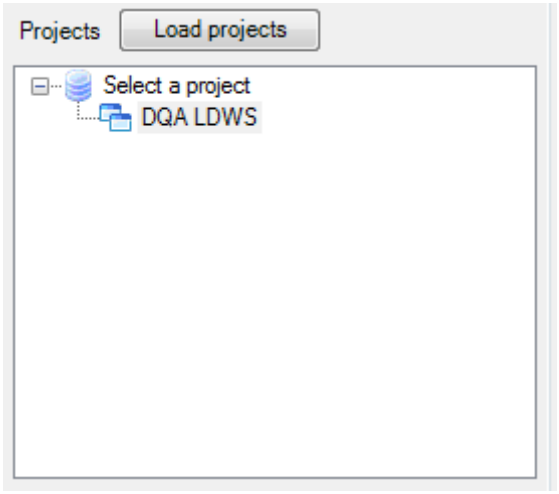
1. Open your project with RQA
2. Move to *Configuration/Batch state*
3. Import the Quality Functions from one of the defined by the QA Team
    - Right click and click on *Import metric configuration*
4. Select a Maturity Level (or any other module with metrics)
5. Select those module names whose Quality Functions you want to overwrite
6. Set *Batch State* if you want the *Synchronizer* to assess the quality of this modules when it is executed



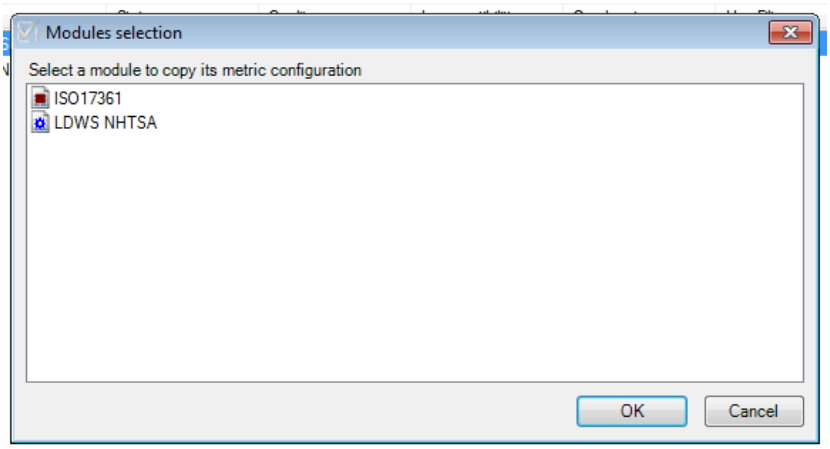
7. Other operations you can perform are:

   - Import new special sentence metrics
   - Import metrics for selected modules: you will be prompted to choose a different project and a concrete module

     i. Select the project



     ii. Select the appropriate module



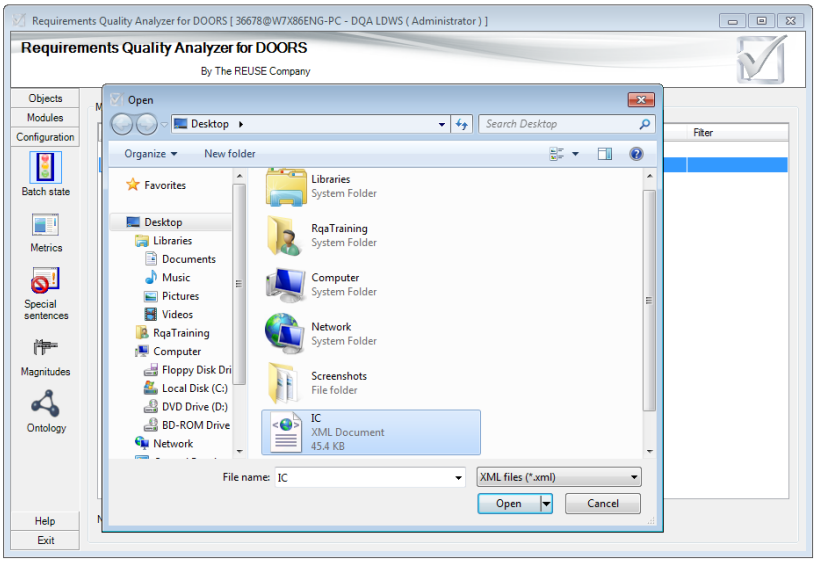
     iii. Click *OK*

   - Import metrics configuration: you will be prompted to choose a different project. All modules in the current project whose module name match with

another module in the target project will be overwritten their set of quality metrics

- Export metrics to file
- Import metrics from file

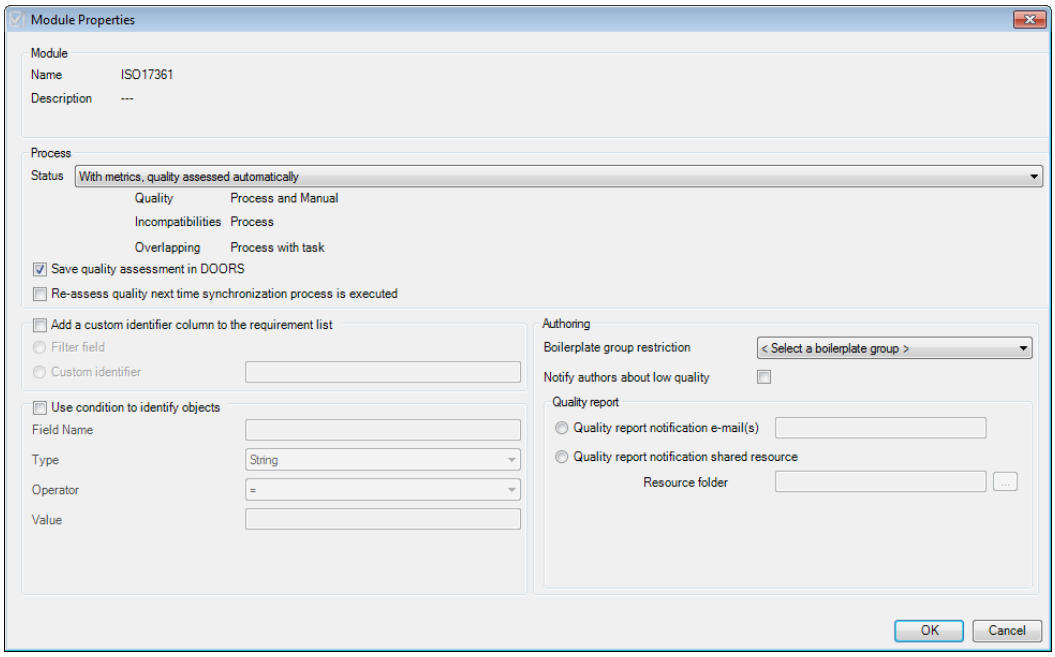
    i. Select the xml file



    ii. Click *Open*

- Show current configuration

- Module properties. The Module Properties window is separated in 5 sections, as shown in the following image:



i. Module.

Display the name and description of the selected module.

ii. Process.

Provides the following options:

- *Status* drop-down menu for defining the quality status for this module, as explained in Batch State.
- *Save Quality Assessment in DOORS.* Stores the results of the quality assessment for each metric back to DOORS. Quality information is added as new attributes to DOORS objects.
- *Re-assess quality next time synchronization process is executed.* By default, quality information is recalculated only if the Title, Description or content of an object is changed. By checking this option, quality will be recalculated in the next synchronization process, regardless of any change to the object.

Note: the re-assessment will be executed only once. After the next synchronization process is executed, this option will be automatically un-checked. You must explicitly check it again for the next time you want quality to be recalculated for all unmodified objects.

iii. Add a custom identifier column to the requirement list.

Defines which field of DOORS is used as an identifier for requirements. The user can choose the field defined in the filter (in Use condition to identify objects section) or a custom identifier.

iv. Use condition to identify objects.

Defines a filter according to which you can specify which objects of DOORS are actual requirements. Four fields are required for defining this filter:

- Field Name: The name of the DOORS field that is checked by the filter.
- Type: The data type of the condition to be checked. It can be String, Boolean or Number.
- Operator: A comparison operator for equality (=), non-equality (!=) and ordered comparison (>, >=, <, <=).
- Value: The result of the above comparison, as required to hold for the filter to apply.

In the example shown in the above window, the filter is defined on the PUID field, with a String data type. For the filter to apply correctly, we require that the string comparison is not equal (!=) to an empty Value. In other words, we consider requirements to be all objects that have some value defined in the PUID field.

v. Authoring.

In this section, we can define options related to authoring requirements:

- Select the group of Boilerplates over which the Boilerplates matching metric is calculated.

- Send a notification email to the author of every requirement that scores low quality.
  Note: if this check box is left unchecked, no notification email will be sent.
- Specify the quality report notification
  - Specify the email address of the person who will receive the quality report, after each execution of the synchronization process.
  - Specify the shared folder where will store the quality report, after each execution of the synchronization process.

  Note: if this field is left empty, no notification will be sent.

10.2 Enabling the inconsistent units global metric

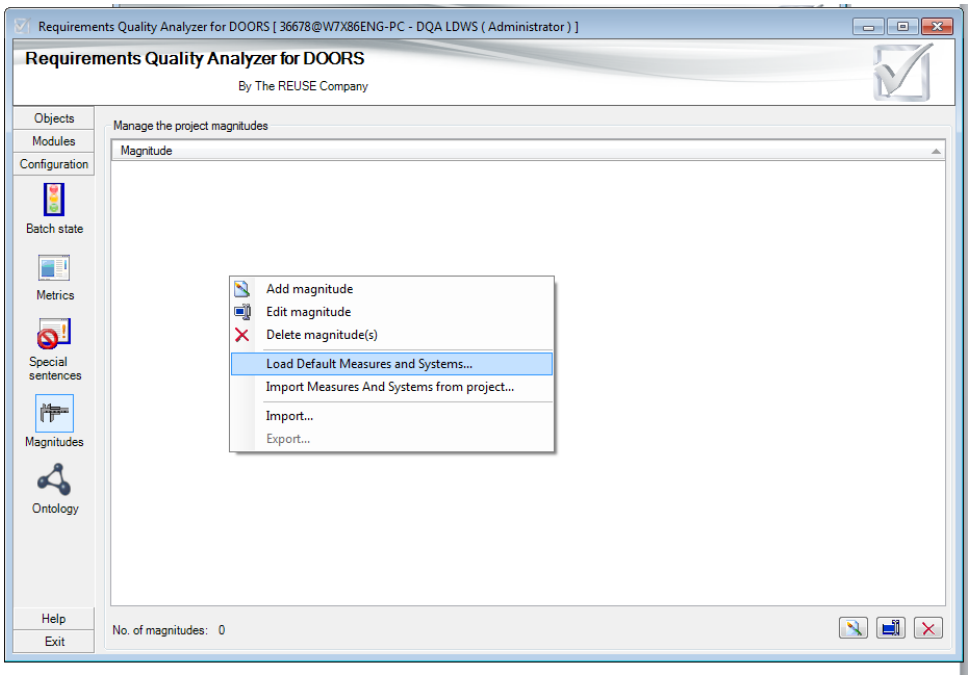
RQA for DOORS can identify projects where two inconsistent units are used at the same time, in two or more different requirements. For example:

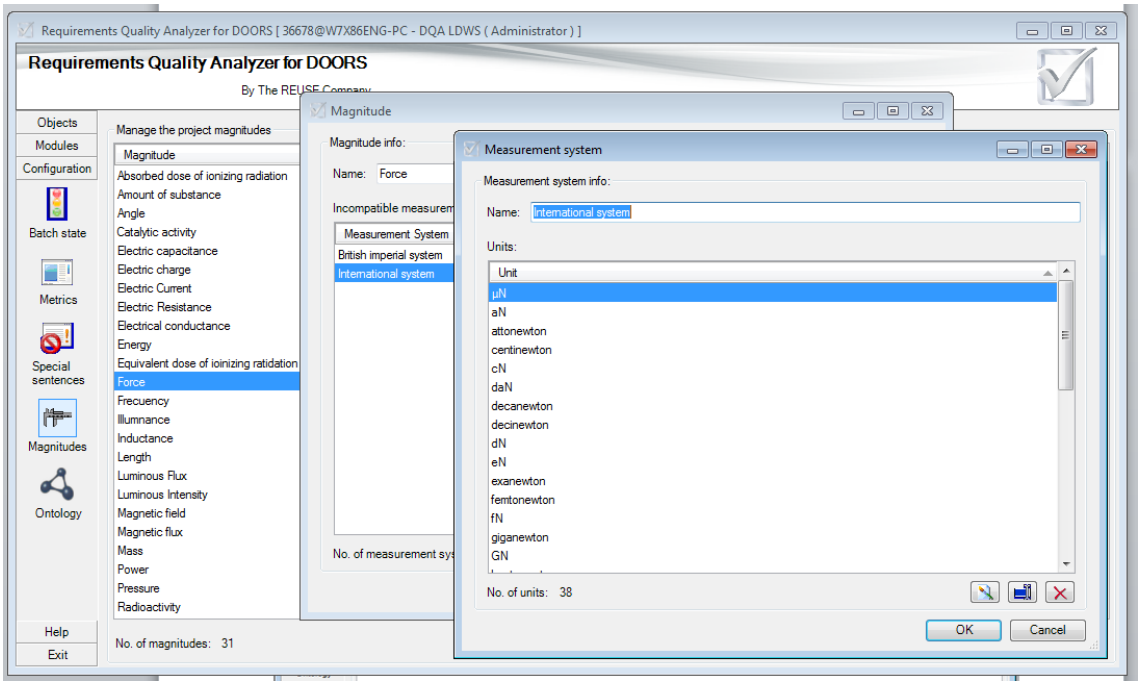
- UR001 The altimeter must calculate the altitude
- UR002 The altitude is measured in feet
- UR003 At 100 meters, the landing gear must be released automatically

A set of units is provided, but it must be enabled in the ontology.

1. Move to *Configuration/Magnitudes*. Unfold the contextual menu and click on *Load Default Measures and Systems*
2. Once the default configuration is loaded, you can add, edit or remove measures and systems
3. Double click on a magnitude, to add, edit or remove systems to it
4. Double click on a system to add, edit, or remove measures to it. Unit names and abbreviations are handled.

5. Import and export from/to a *csv* file





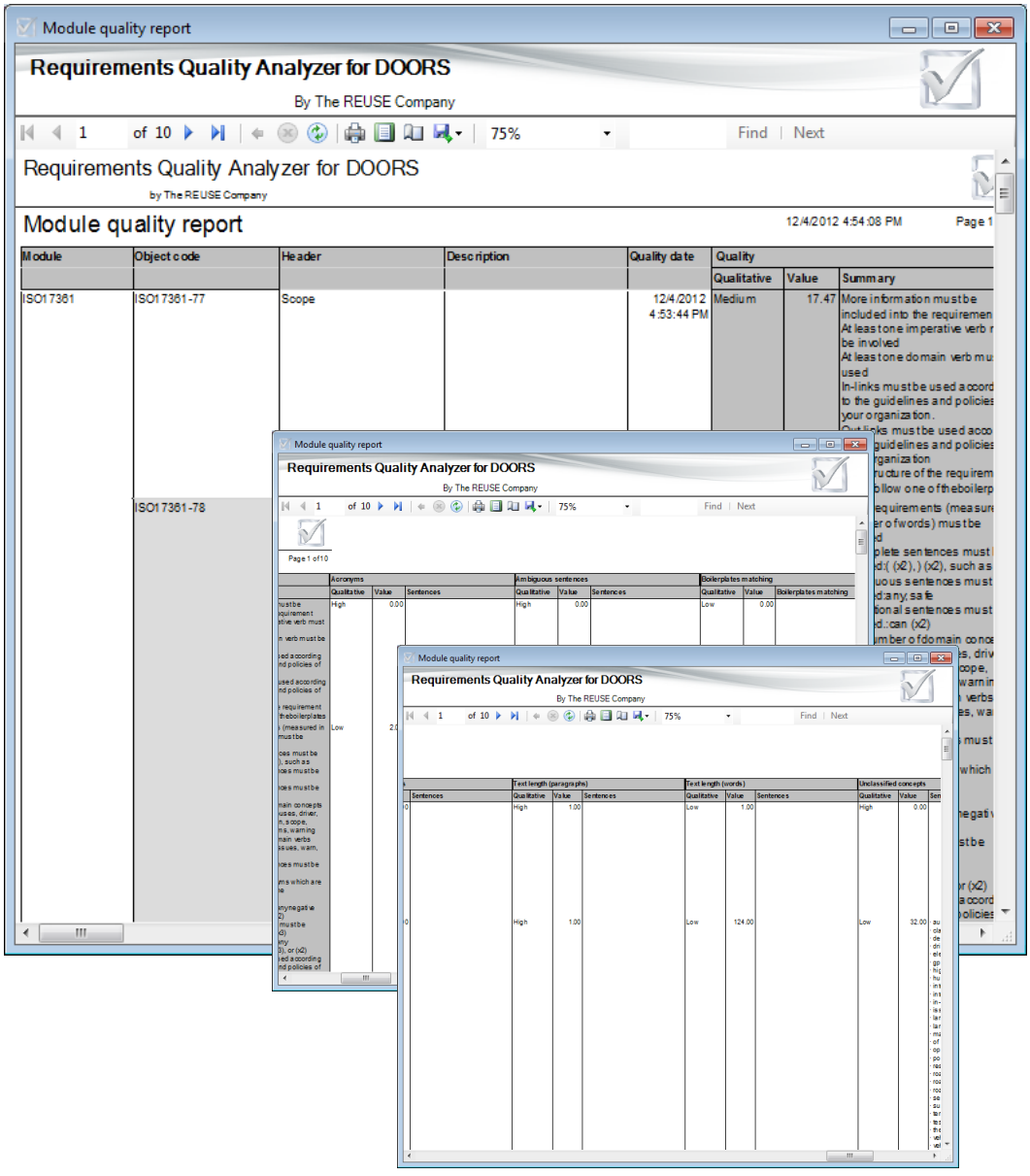
10.3 Reviewing the quality of a module or project

The Project Manager must be able to review the quality of the corresponding projects or modules. In order to do so, the Project Manager should:

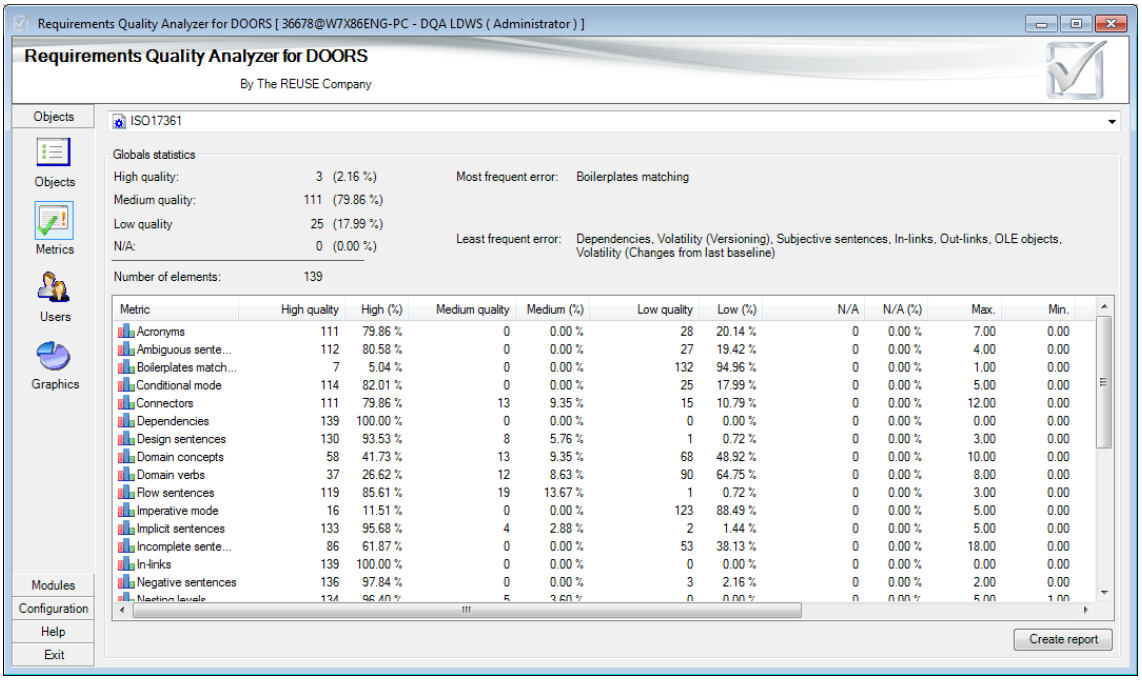
1. Select your project or module using the combo box in the upper part of the window
2. Once you have assigned a set of Quality Functions, quality can be calculated:
    a. Manually: Move to *Objects/Objects*, select your requirements, right click and click on *Recalculate* (could be a very time consuming task)
    b. Automatically: let the *Synchronizer* to calculate the quality on regular basis

Note: Remember to calculate the quality of the module in RQA and save it back in DOORS, the DOORS module must be closed, not locked and the DOORS user must have writing rights in that DOORS module.

3. A new window appears, containing the quality report for the selected Object. The report window is usually very wide, since it contains all calculated metrics and quality information organized in columns of a table. It is often the case that the BA must scroll horizontally to navigate through the report. The following three screens depict the generated report in its entire width. All calculated metrics are clearly visible in the columns of the report, while their values appear in the corresponding cells, for each object in the module.



4. Once the quality has been calculated, move to *Objects/Metrics* and see:

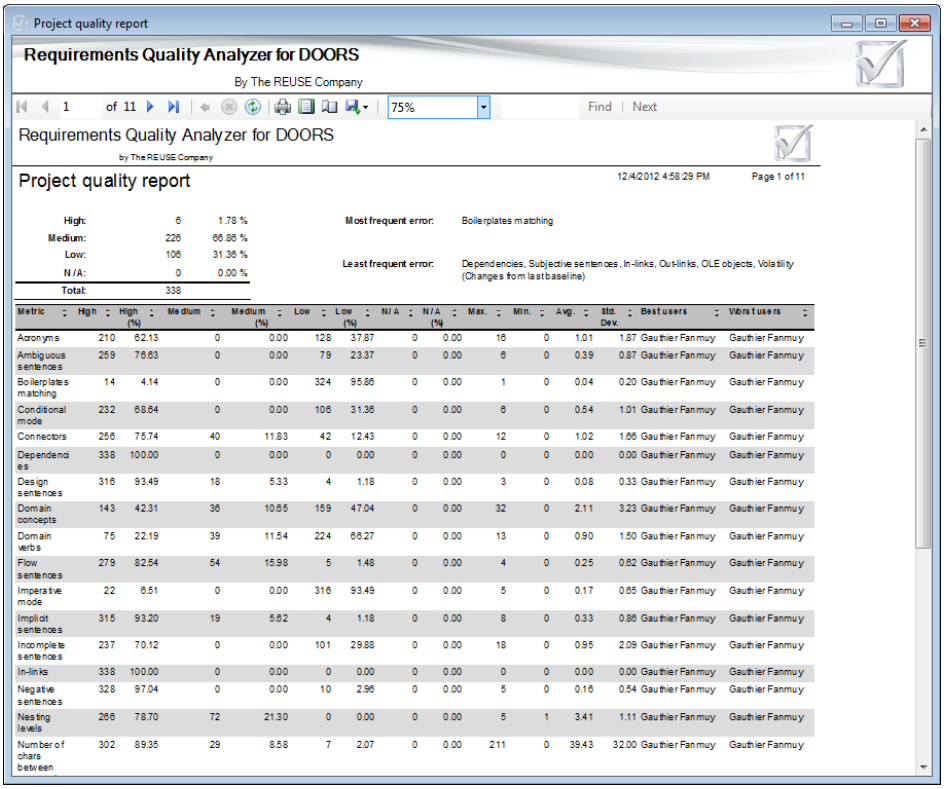


   a. How many requirements you have in high, medium or low quality

   b. Which are the most or least frequent mistakes

   c. Relevant statistical information for every metric

d. Metric quality information for each module can be presented as a report.



Project quality report

Requirements Quality Analyzer for DOORS

By The REUSE Company

1 of 11 | 75% | Find | Next

Requirements Quality Analyzer for DOORS

by The REUSE Company

Project quality report

12/4/2012 4:58:29 PM Page 1 of 11

High:	6	1.78 %
Medium:	226	66.86 %
Low:	106	31.36 %
N/A:	0	0.00 %
Total:	338	

Most frequent error: Boilerplates matching

Least frequent error: Dependencies, Subjective sentences, In-links, Out-links, OLE objects, Volatility (Changes from last baseline)

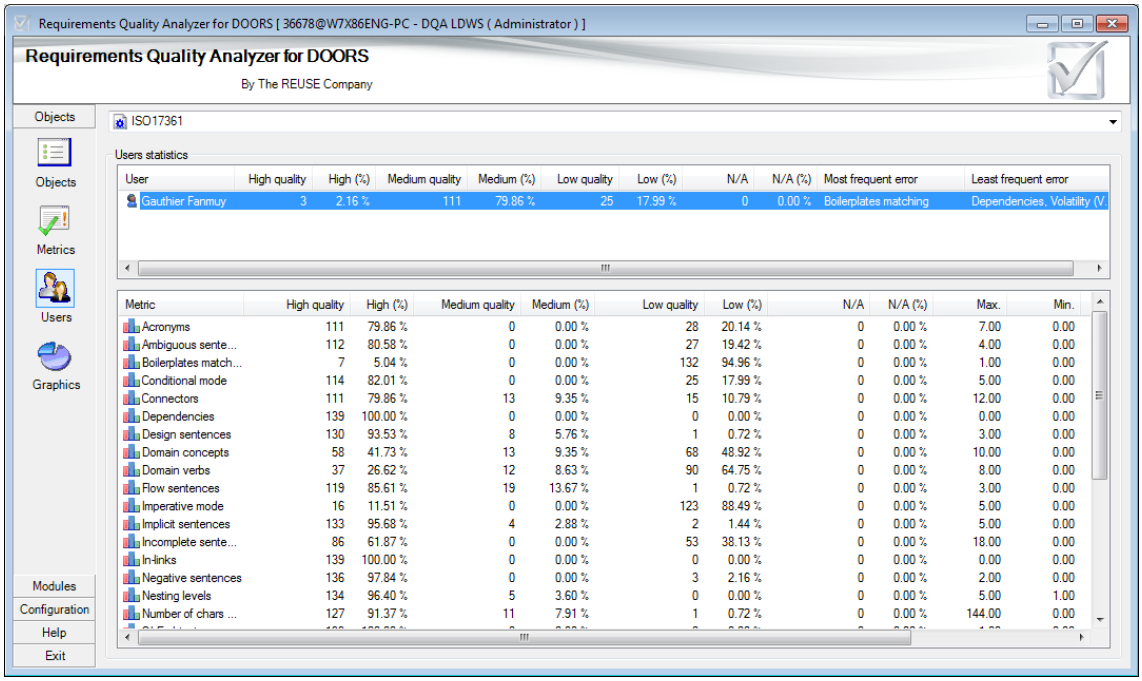
Metric	High	High (%)	Medium	Medium (%)	Low	Low (%)	N/A	N/A (%)	Max.	Min.	Avg.	Std. Dev.	Best users	Worst users
Acronyms	210	62.13	0	0.00	128	37.87	0	0.00	16	0	1.01	1.87	Gauthier Fanmuy	Gauthier Fanmuy
Ambiguous sentences	259	76.63	0	0.00	79	23.37	0	0.00	6	0	0.39	0.87	Gauthier Fanmuy	Gauthier Fanmuy
Boilerplates matching	14	4.14	0	0.00	324	95.86	0	0.00	1	0	0.04	0.20	Gauthier Fanmuy	Gauthier Fanmuy
Conditional mode	232	68.64	0	0.00	106	31.36	0	0.00	6	0	0.54	1.01	Gauthier Fanmuy	Gauthier Fanmuy
Connectors	256	75.74	40	11.83	42	12.43	0	0.00	12	0	1.02	1.66	Gauthier Fanmuy	Gauthier Fanmuy
Dependencies	338	100.00	0	0.00	0	0.00	0	0.00	0	0	0.00	0.00	Gauthier Fanmuy	Gauthier Fanmuy
Design sentences	316	93.49	18	5.33	4	1.18	0	0.00	3	0	0.08	0.33	Gauthier Fanmuy	Gauthier Fanmuy
Domain concepts	143	42.31	36	10.65	159	47.04	0	0.00	32	0	2.11	3.23	Gauthier Fanmuy	Gauthier Fanmuy
Domain verbs	75	22.19	39	11.54	224	66.27	0	0.00	13	0	0.90	1.50	Gauthier Fanmuy	Gauthier Fanmuy
Flow sentences	279	82.54	54	15.98	5	1.48	0	0.00	4	0	0.25	0.62	Gauthier Fanmuy	Gauthier Fanmuy
Imperative mode	22	6.51	0	0.00	316	93.49	0	0.00	5	0	0.17	0.65	Gauthier Fanmuy	Gauthier Fanmuy
Implicit sentences	315	93.20	19	5.62	4	1.18	0	0.00	8	0	0.33	0.86	Gauthier Fanmuy	Gauthier Fanmuy
Incomplete sentences	237	70.12	0	0.00	101	29.88	0	0.00	18	0	0.95	2.09	Gauthier Fanmuy	Gauthier Fanmuy
In-links	338	100.00	0	0.00	0	0.00	0	0.00	0	0	0.00	0.00	Gauthier Fanmuy	Gauthier Fanmuy
Negative sentences	328	97.04	0	0.00	10	2.96	0	0.00	5	0	0.16	0.54	Gauthier Fanmuy	Gauthier Fanmuy
Nesting levels	266	78.70	72	21.30	0	0.00	0	0.00	5	1	3.41	1.11	Gauthier Fanmuy	Gauthier Fanmuy
Number of chars between	302	89.35	29	8.58	7	2.07	0	0.00	211	0	39.43	32.00	Gauthier Fanmuy	Gauthier Fanmuy

5. Export this statistical information to a *csv* file using the contextual menu

All of this statistical information can be broken down by users:

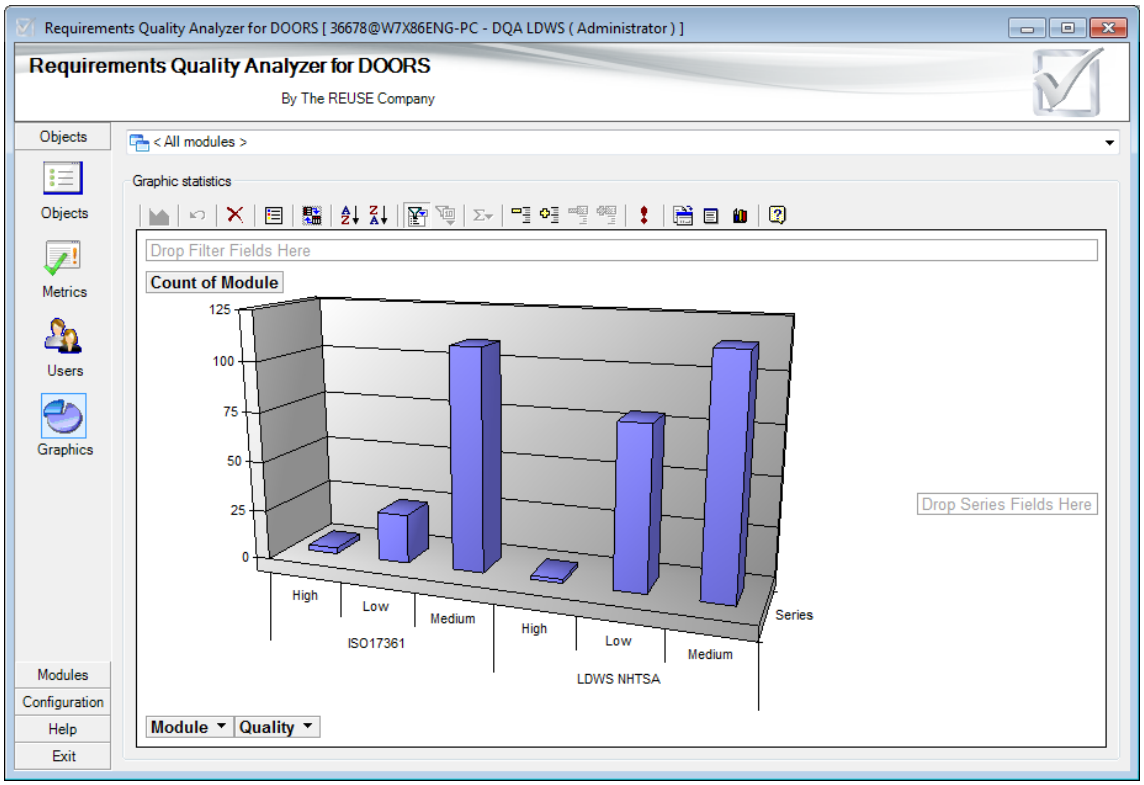
1. Move to *Objects/Users* and select a single user

2. All of the information should be related to this Business Analyst (BA)



The statistical information could be also depicted as dynamic graphs:

- Move to *Objects/Graphics* and create your dynamic graph using all of the data provided by RQA



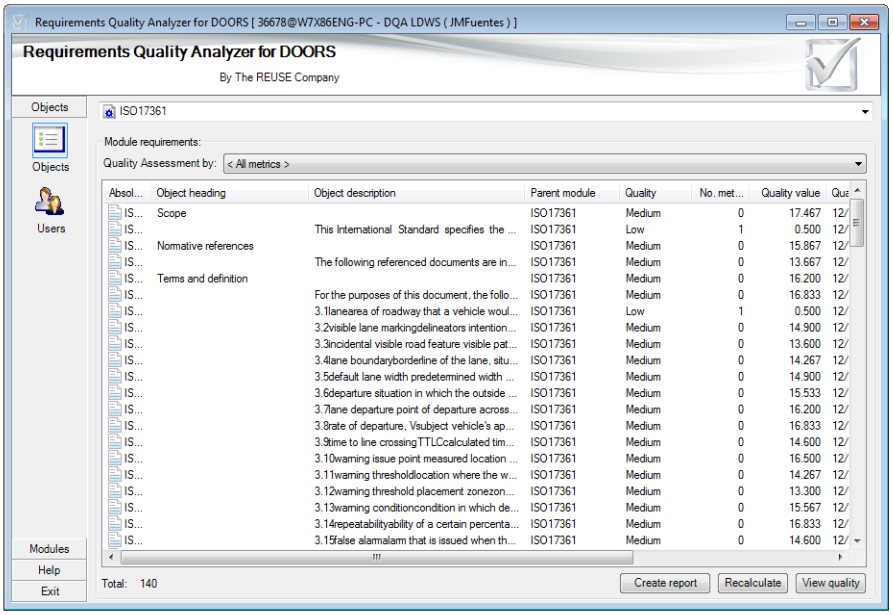
11. RQA for Business Analysts

11.1 Analyzing the quality of a requirement

The Business Analyst (BA) of the project must be able to check the quality of your requirements as they are being written (on-the-fly). The BA must also be able to see the aggregated quality of your requirements in the corresponding project.

11.2 Analyzing the quality of an individual requirement

1. Move to *Objects/Objects*
2. Select the module you want to analyze using the combo box in the upper part of the window
3. If needed, select your requirements and click on *Recalculate* in order to calculate again (or for the very first time) the quality of the selected requirements
4. A couple of columns help you deciding which requirements you should focus on:
    a. Quality: contains a label corresponding to high, medium or low quality
    b. Quality value: helps you prioritizing your requirements



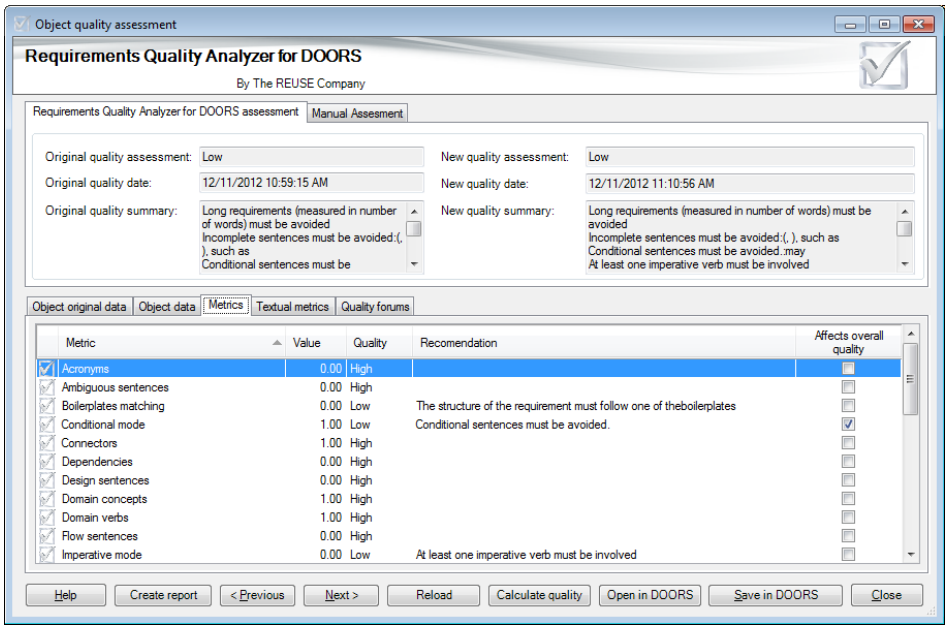
5. Now, double click (or use the contextual menu and *View Quality*) in order to get deeper into the quality of an individual requirement

6. You can see the result of the quality assessment, but you also see:

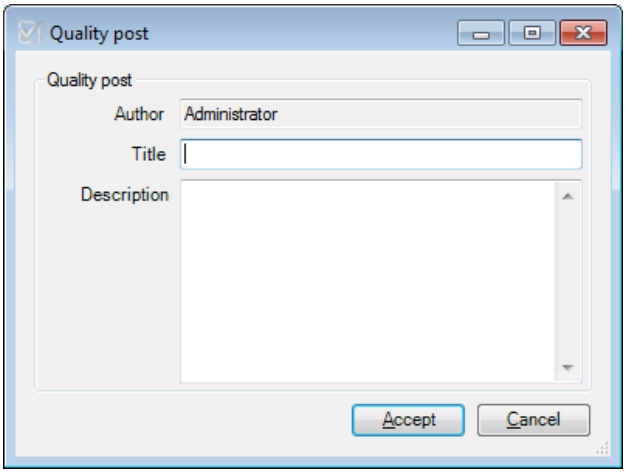
    a. A summary with all the recommendations made by RQA

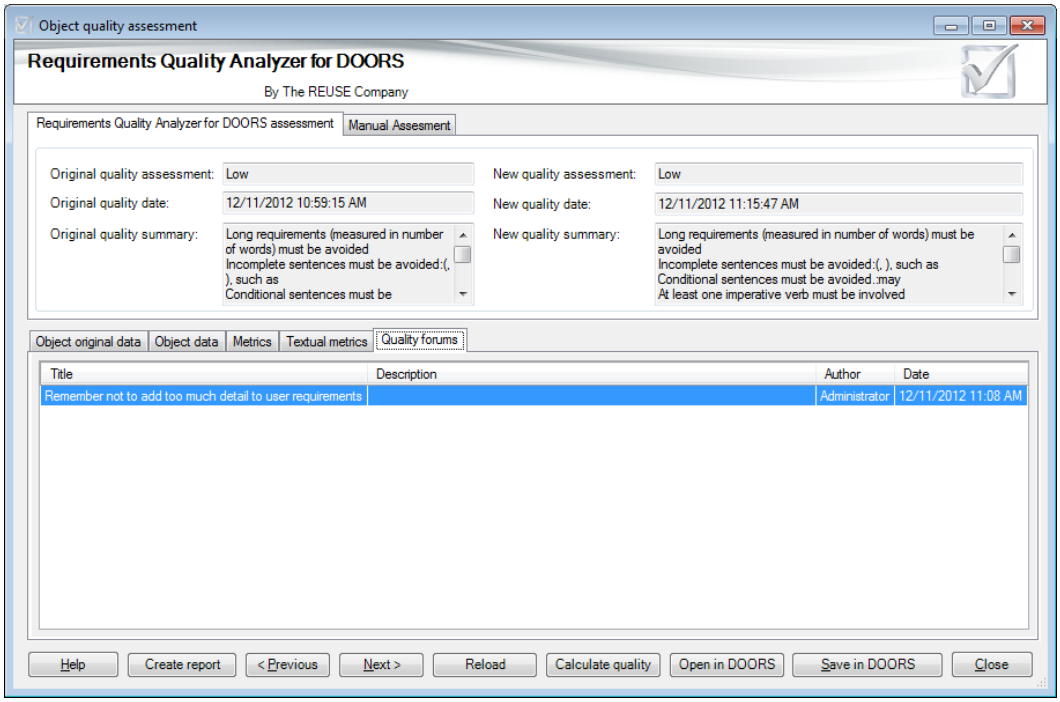
    b. The result for every individual Metric

    c. Which are the terms detected by RQA that appears in your requirement (domain terms, domain verbs, conditional sentences, imperative sentences, forbidden sentences…)

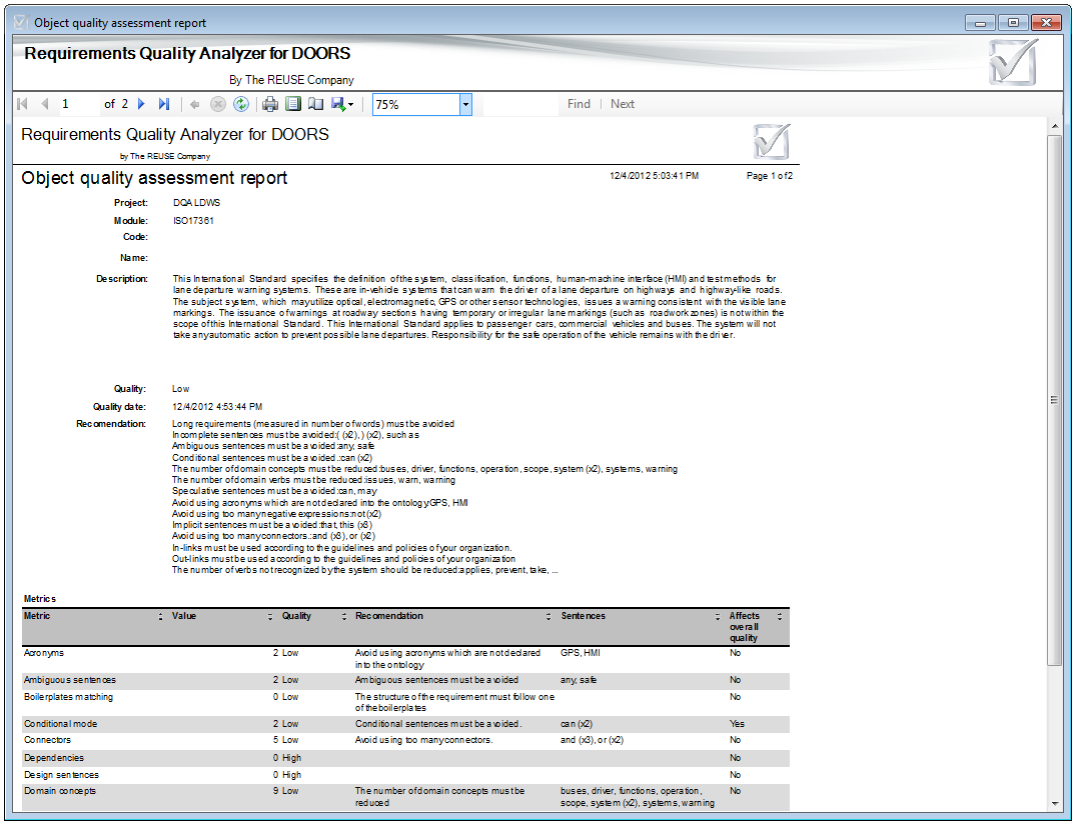


d. Read and write a quality forum system for every requirement





e. Quality information for each object can be presented as a report.



Object quality assessment report

Requirements Quality Analyzer for DOORS

By The REUSE Company

1 of 2 | 75% | Find | Next

Requirements Quality Analyzer for DOORS

by The REUSE Company

Object quality assessment report

12/4/2012 5:03:41 PM Page 1 of 2

Project: DQA LDWS
Module: ISO17361
Code:
Name:
Description: This International Standard specifies the definition of the system, classification, functions, human-machine interface (HMI) and test methods for lane departure warning systems. These are in-vehicle systems that can warn the driver of a lane departure on highways and highway-like roads. The subject system, which may utilize optical, electromagnetic, GPS or other sensor technologies, issues a warning consistent with the visible lane markings. The issuance of warnings at roadway sections having temporary or irregular lane markings (such as roadwork zones) is not within the scope of this International Standard. This International Standard applies to passenger cars, commercial vehicles and buses. The system will not take any automatic action to prevent possible lane departures. Responsibility for the safe operation of the vehicle remains with the driver.

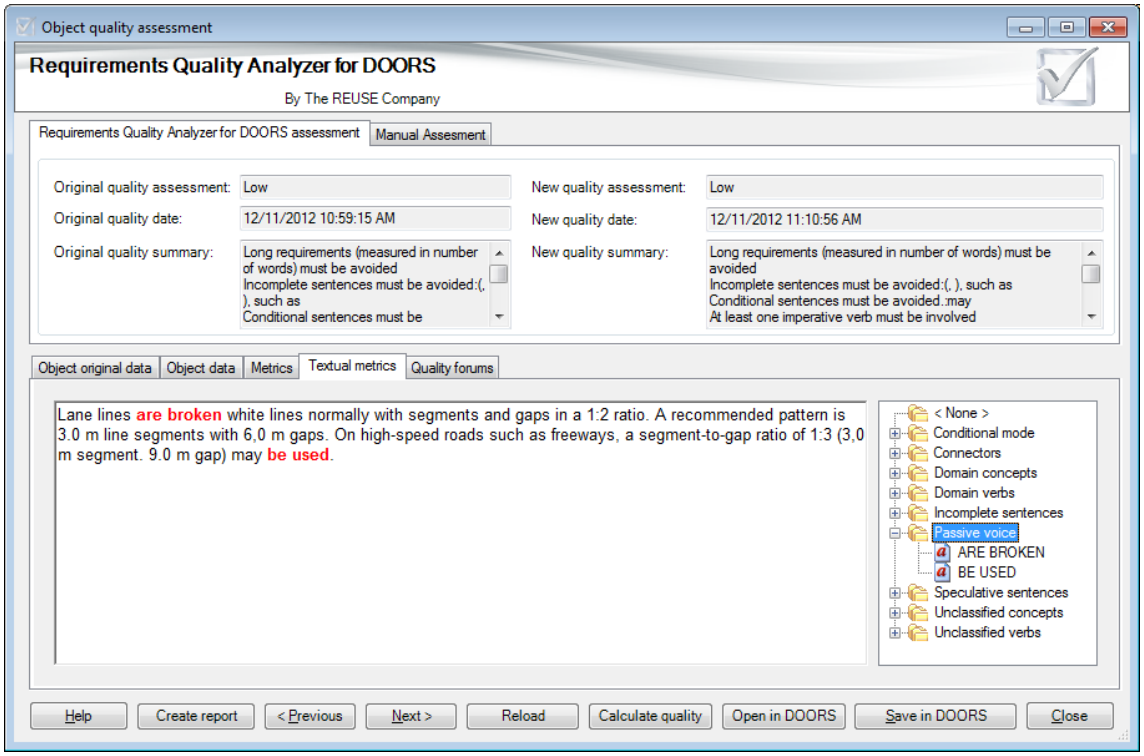
Quality: Low
Quality date: 12/4/2012 4:53:44 PM
Recomendation:
Long requirements (measured in number of words) must be avoided
Incomplete sentences must be avoided:((x2),) (x2), such as
Ambiguous sentences must be avoided:any, safe
Conditional sentences must be avoided.:can (x2)
The number of domain concepts must be reduced:buses, driver, functions, operation, scope, system (x2), systems, warning
The number of domain verbs must be reduced:issues, warn, warning
Speculative sentences must be avoided:can, may
Avoid using acronyms which are not declared into the ontology:GPS, HMI
Avoid using too many negative expressions:not (x2)
Implicit sentences must be avoided:that, this (x8)
Avoid using too many connectors.:and (x8), or (x2)
In-links must be used according to the guidelines and policies of your organization.
Out-links must be used according to the guidelines and policies of your organization
The number of verbs not recognized by the system should be reduced:applies, prevent, take, ...

Metrics

Metric	Value	Quality	Recomendation	Sentences	Affects overall quality
Acronyms	2	Low	Avoid using acronyms which are not declared into the ontology	GPS, HMI	No
Ambiguous sentences	2	Low	Ambiguous sentences must be avoided	any, safe	No
Boilerplates matching	0	Low	The structure of the requirement must follow one of the boilerplates		No
Conditional mode	2	Low	Conditional sentences must be avoided.	can (x2)	Yes
Connectors	5	Low	Avoid using too many connectors.	and (x3), or (x2)	No
Dependencies	0	High			No
Design sentences	0	High			No
Domain concepts	9	Low	The number of domain concepts must be reduced	buses, driver, functions, operation, scope, system (x2), systems, warning	No

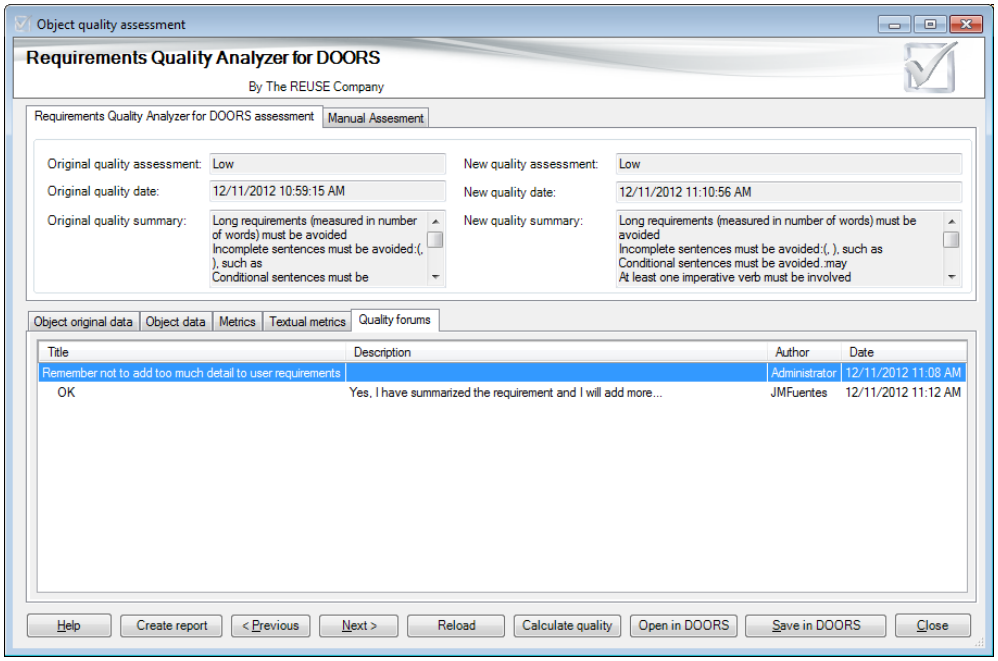
Using all the information provided by these dialogs, try to correct the problems of your requirements. Once a requirement has been re-written, you are able to:

- Calculate the quality of the new text as many times as needed
- Reload the original text from DOORS
- Save your new text to DOORS
- Open the current requirement in DOORS



Every requirement implements a forum system. The Project Manager and Business Analysts can interchange messages regarding quality metrics:

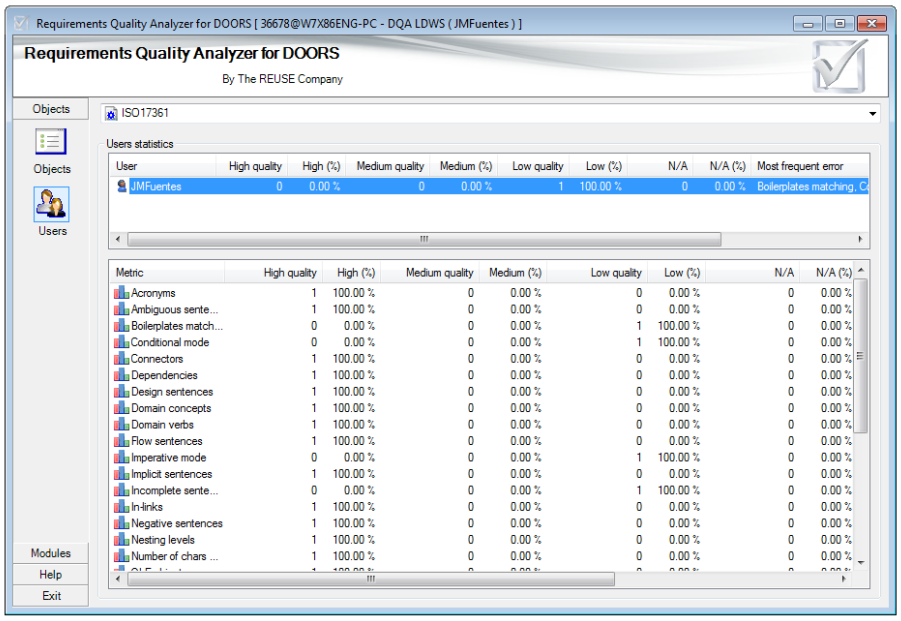
1. Go to *Quality Forum* tab
2. Use the contextual menu to add:
    a. New threads
    b. New comments to an existing thread



11.3 Aggregating the quality of all of your requirements

The BA of the project should not have Admin privileges in RQA. Therefore, a BA can only aggregate the quality of the requirements written by them. In order to do so, a BA should:

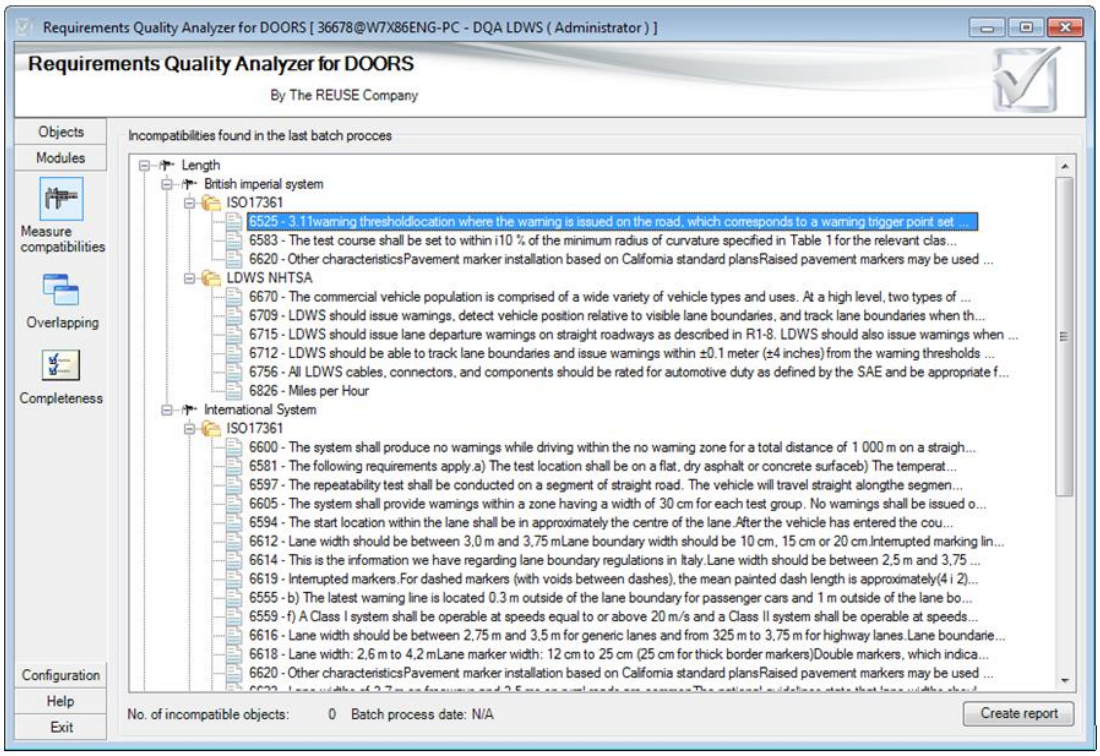
1. Move to *Objects/Users* and find out which are your most/least frequent mistakes, statistical information about every metric…
2. Export this statistical information into a *.csv* file if needed
    a. Use the contextual menu for this purpose



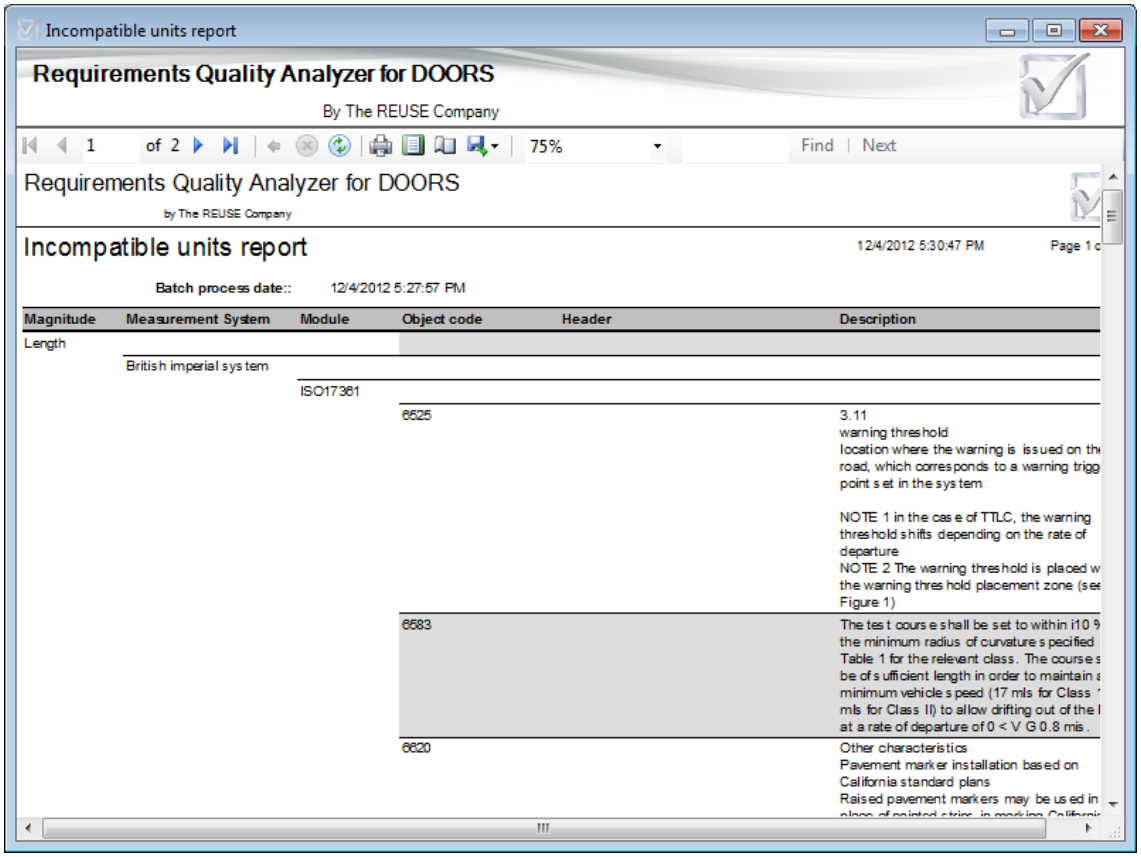
11.4 Looking for inconsistent units

RQA is able to detect the usage of inconsistent units within the same project. For instance, meters and yards should not be mixed together in a project. This task is not executed in the RQA Client; instead, it is executed by the Batch Process (Synchronizer). This task is executed for every module with metrics and batch state ON. But the results can be reviewed with the Client.

Click on *Modules/Measures compatibilities*



Compatibility quality information can be presented as a report.

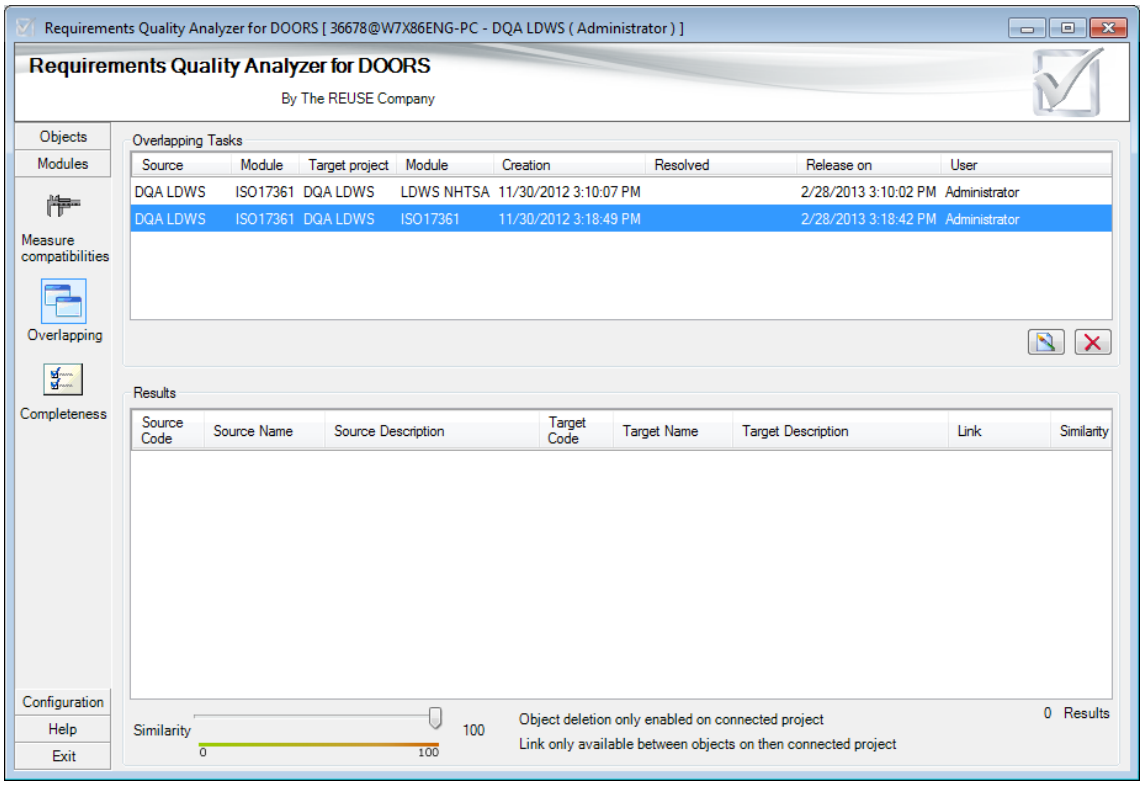


11.5 Looking for overlapping requirements

RQA is able to generate a matrix of overlapping requirements for different purposes:

- Overlapping requirements in the same module: redundancies may lead to incompatibilities and rework.
- Overlapping requirements between user and system requirements: it can be a helpful tool to elicit traceability.

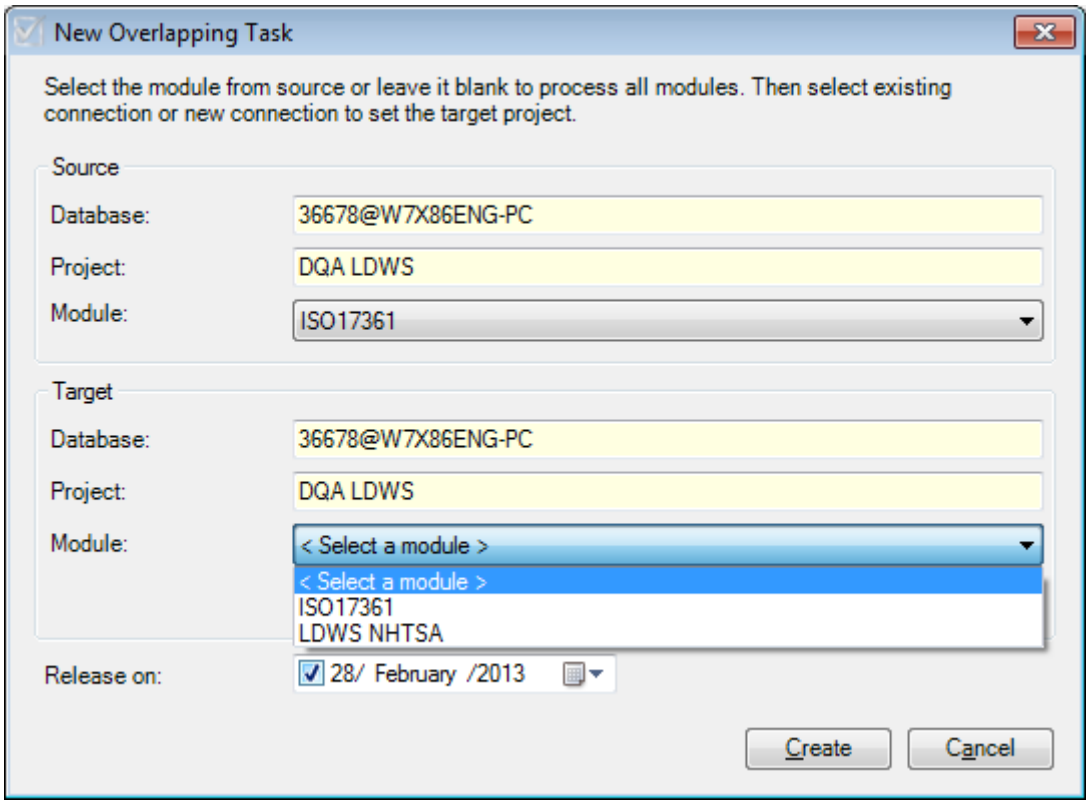
You must use the RQA Client to program the overlapping task and later on review the results. However, this process is always executed by the Synchronizer (it is a time consuming task).



Creating an overlapping task:

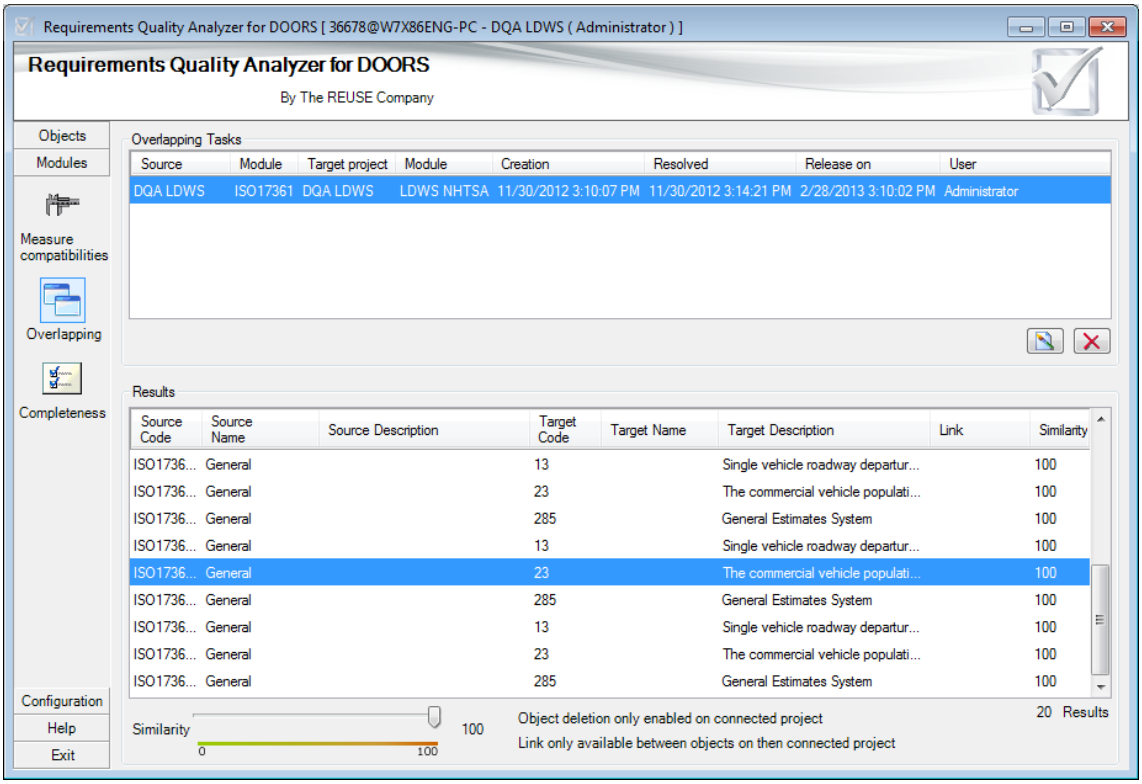
1. Move to *Modules/Overlapping*
2. Use the contextual menu or the buttons to create a new overlapping task
3. Select the source module you want to analyze
4. Target could be the same module, or another module in the same project: click on *Existing connection*
5. Target could also be placed in a different project: click on *New connection*
6. Unfold the *Module* combo box to select the target module

7. Overlapping results will be automatically removed from the database in the date selected in this dialog box: *Release on* data field



Reviewing overlapping results:

1. Move to *Modules/Overlapping*
2. Select the overlapping task you want to review



3. Identical requirements will arise with a similarity value of 100%
4. Other overlapped (not identical) requirements will be shown with a lower similarity value: move the *Similarity* slider to show more or less pairs
5. Use the contextual menu and *Show overlapped objects*

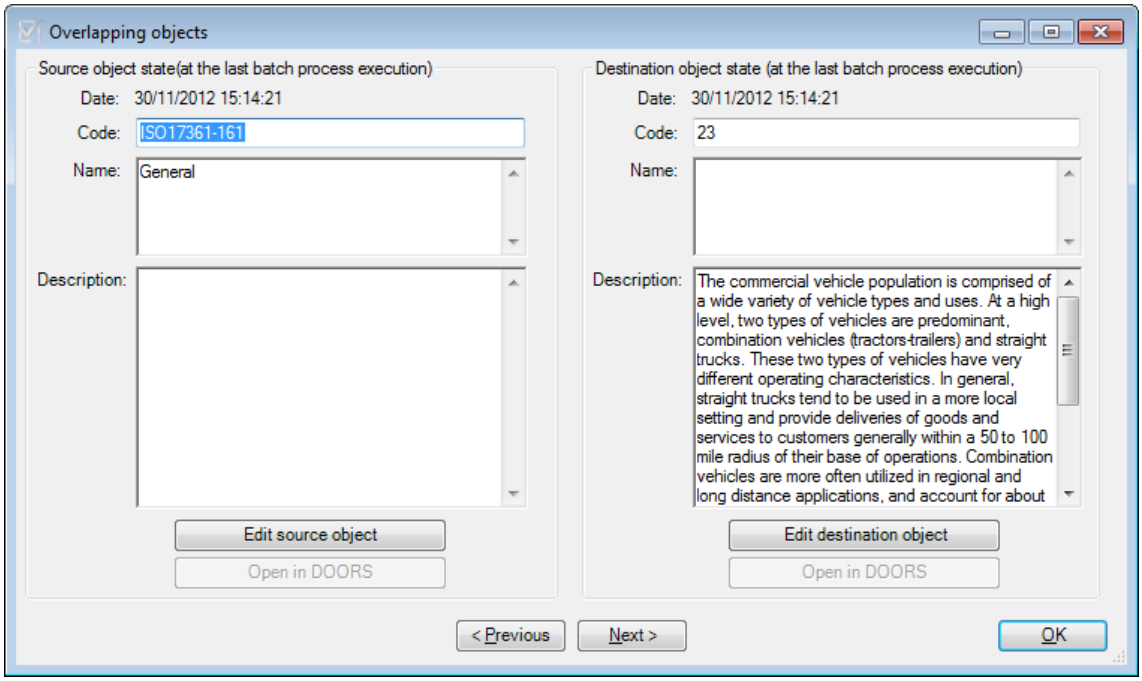
6. With this dialog, you'll be able to:

    a. Check the text of both requirements (one from the source module and another one from the target module)

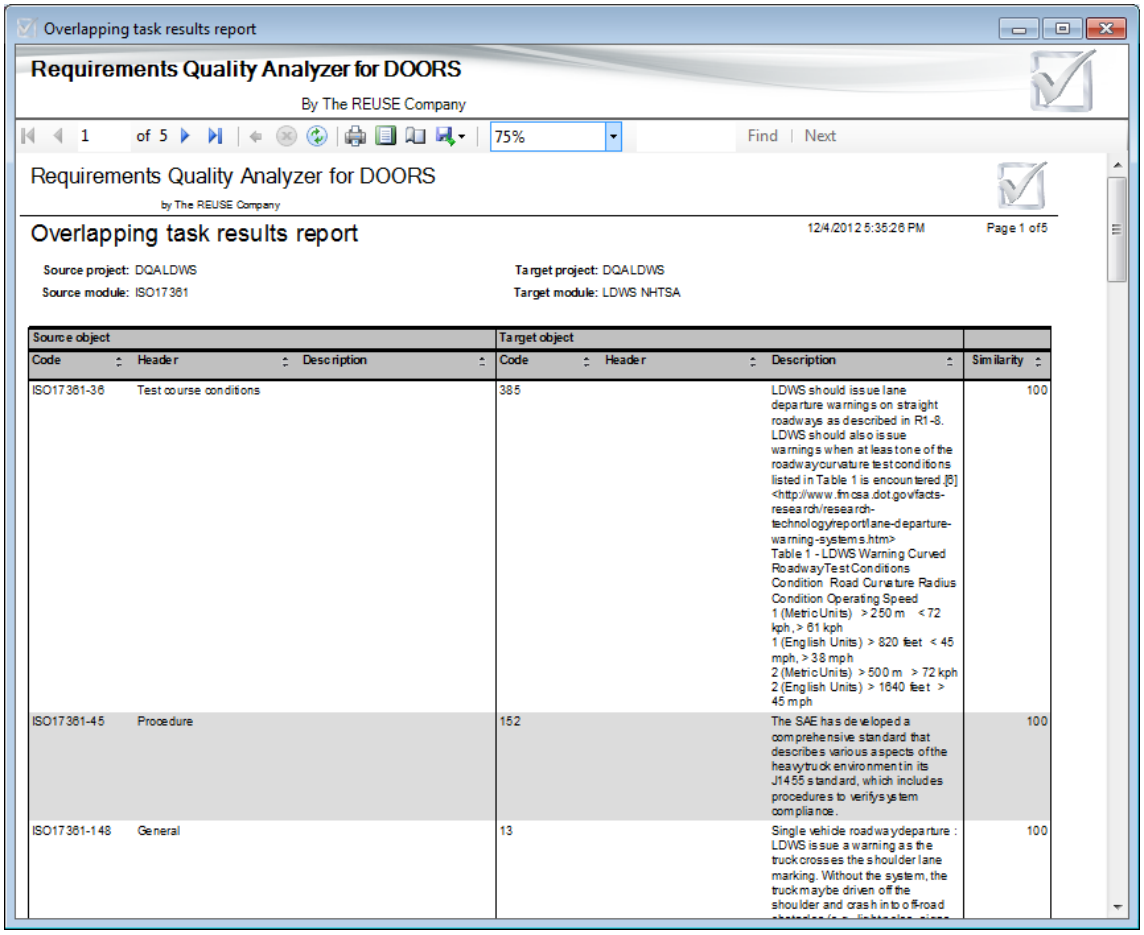
    b. Open one of the requirement (source or target) in DOORS

    c. Open one of the requirements in the RQA edit dialog

    d. Move to the next or previous couple of requirements



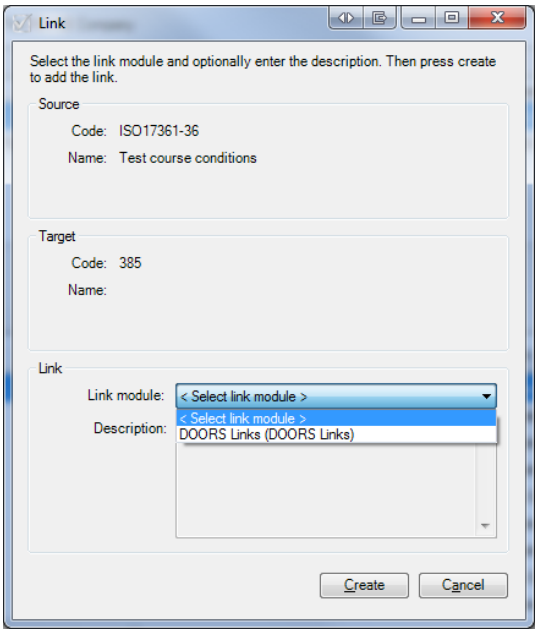
7. Overlapping quality information can be presented as a report



Acting with the overlapping results: the actions allowed by RQA Client, for every couple of overlapped requirements, are the following:

- Delete source: deletes the overlapped requirements in the source module
- Delete target: deletes the overlapped requirements in the target module

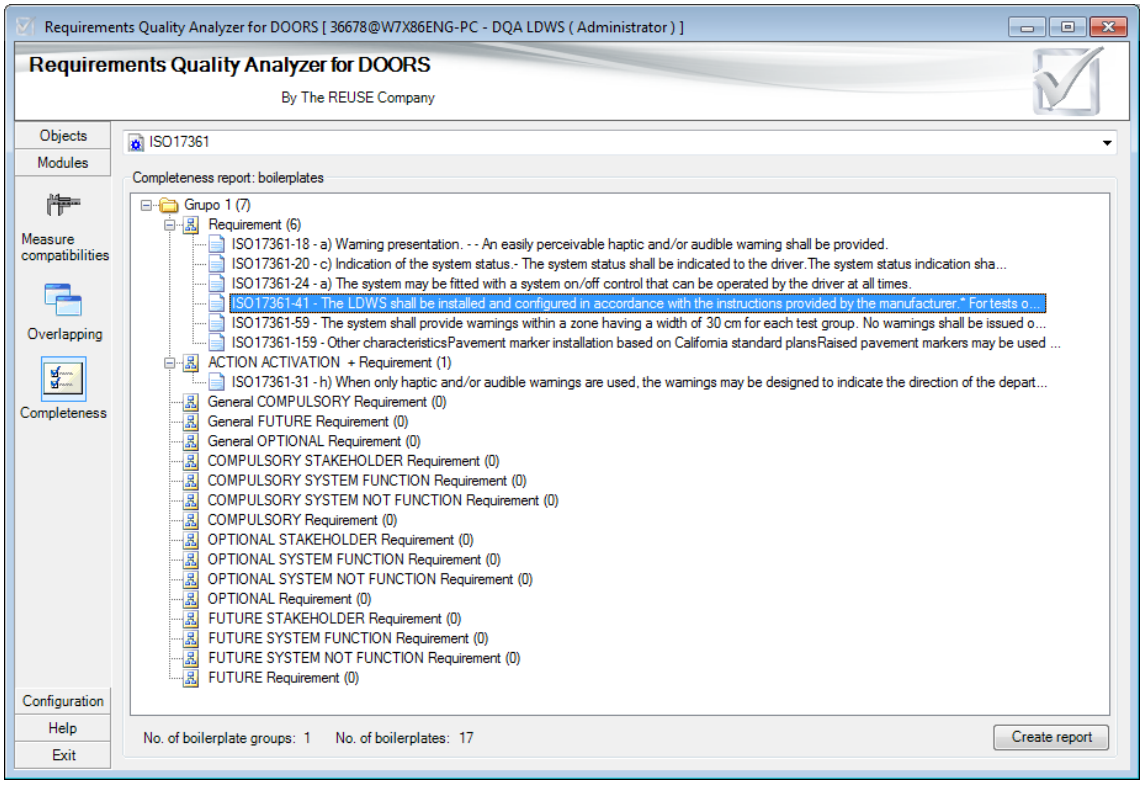
- Link: use this dialog box to easily create a link between source and target requirements
  - RQA links are stored in the DOORS database
  - RQA links are based on the link modules already existing in the DOORS database



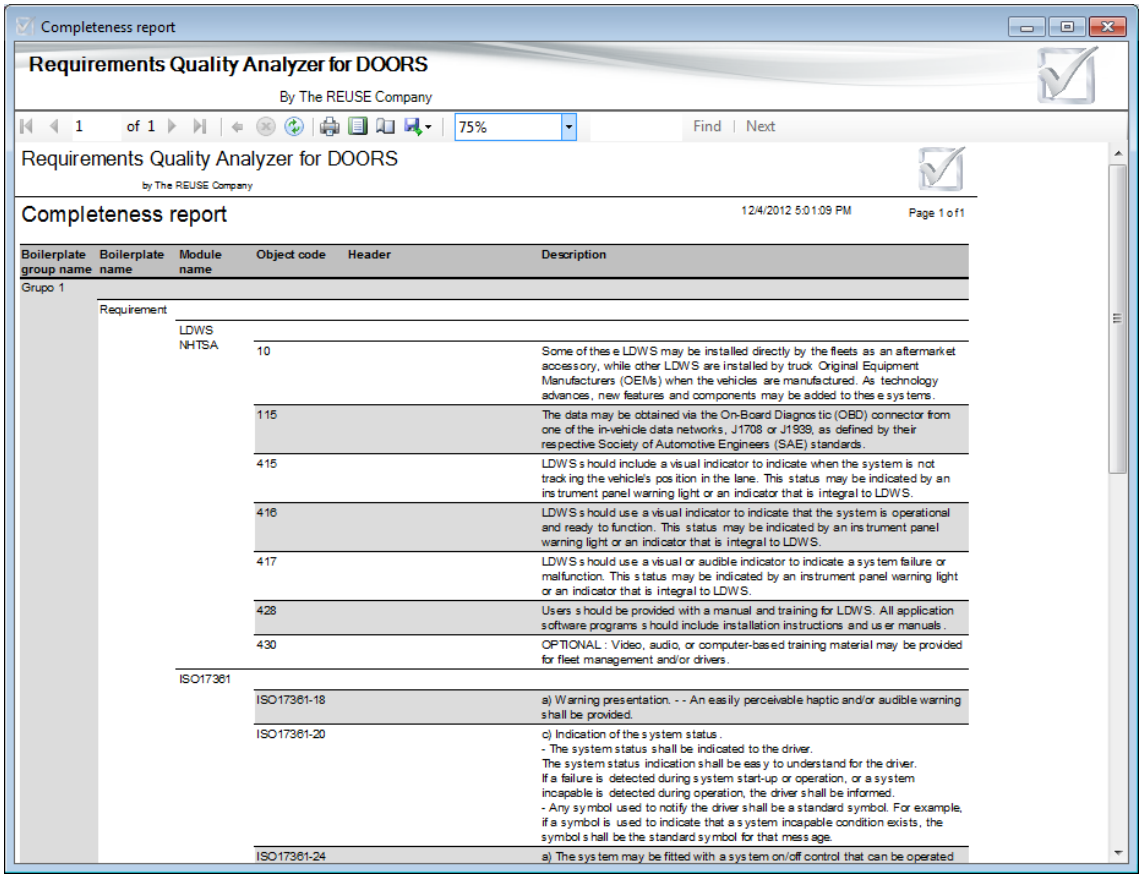
11.6 Assessing completeness for your requirements

RQA is able to group each requirement in each group and boilerplate which it belongs to. This task is not executed in the RQA Client; instead, it is executed by the Batch Process (Synchronizer). This task is executed for every module with metrics and batch state ON. But the results can be reviewed with the Client.

Click on *Modules/Completeness*



Completeness quality information can be presented as a report.



Completeness report

Requirements Quality Analyzer for DOORS

By The REUSE Company

1 of 1 | 75% | Find | Next

Requirements Quality Analyzer for DOORS

by The REUSE Company

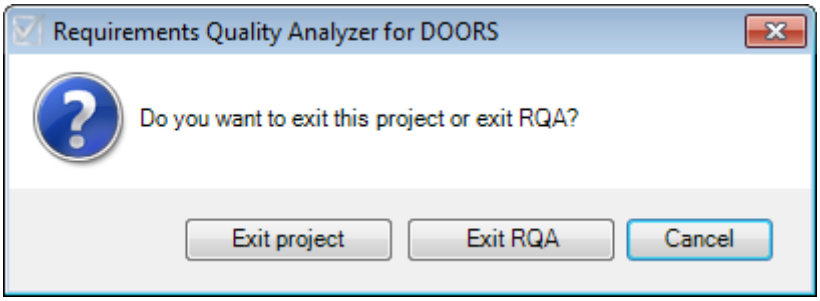
Completeness report

12/4/2012 5:01:09 PM Page 1 of 1

Boilerplate group name	Boilerplate name	Module name	Object code	Header	Description
Grupo 1					
	Requirement				
		LDWS NHTSA			
			10		Some of these LDWS may be installed directly by the fleets as an aftermarket accessory, while other LDWS are installed by truck Original Equipment Manufacturers (OEMs) when the vehicles are manufactured. As technology advances, new features and components may be added to these systems.
			115		The data may be obtained via the On-Board Diagnostic (OBD) connector from one of the in-vehicle data networks, J1708 or J1939, as defined by their respective Society of Automotive Engineers (SAE) standards.
			415		LDWS should include a visual indicator to indicate when the system is not tracking the vehicle's position in the lane. This status may be indicated by an instrument panel warning light or an indicator that is integral to LDWS.
			416		LDWS should use a visual indicator to indicate that the system is operational and ready to function. This status may be indicated by an instrument panel warning light or an indicator that is integral to LDWS.
			417		LDWS should use a visual or audible indicator to indicate a system failure or malfunction. This status may be indicated by an instrument panel warning light or an indicator that is integral to LDWS.
			428		Users should be provided with a manual and training for LDWS. All application software programs should include installation instructions and user manuals.
			430		OPTIONAL: Video, audio, or computer-based training material may be provided for fleet management and/or drivers.
		ISO17361			
			ISO17361-18		a) Warning presentation. - - An easily perceivable haptic and/or audible warning shall be provided.
			ISO17361-20		c) Indication of the system status. - The system status shall be indicated to the driver. The system status indication shall be easy to understand for the driver. If a failure is detected during system start-up or operation, or a system incapable is detected during operation, the driver shall be informed. - Any symbol used to notify the driver shall be a standard symbol. For example, if a symbol is used to indicate that a system incapable condition exists, the symbol shall be the standard symbol for that message.
			ISO17361-24		a) The system may be fitted with a system on/off control that can be operated

12. Exiting the RQA Client

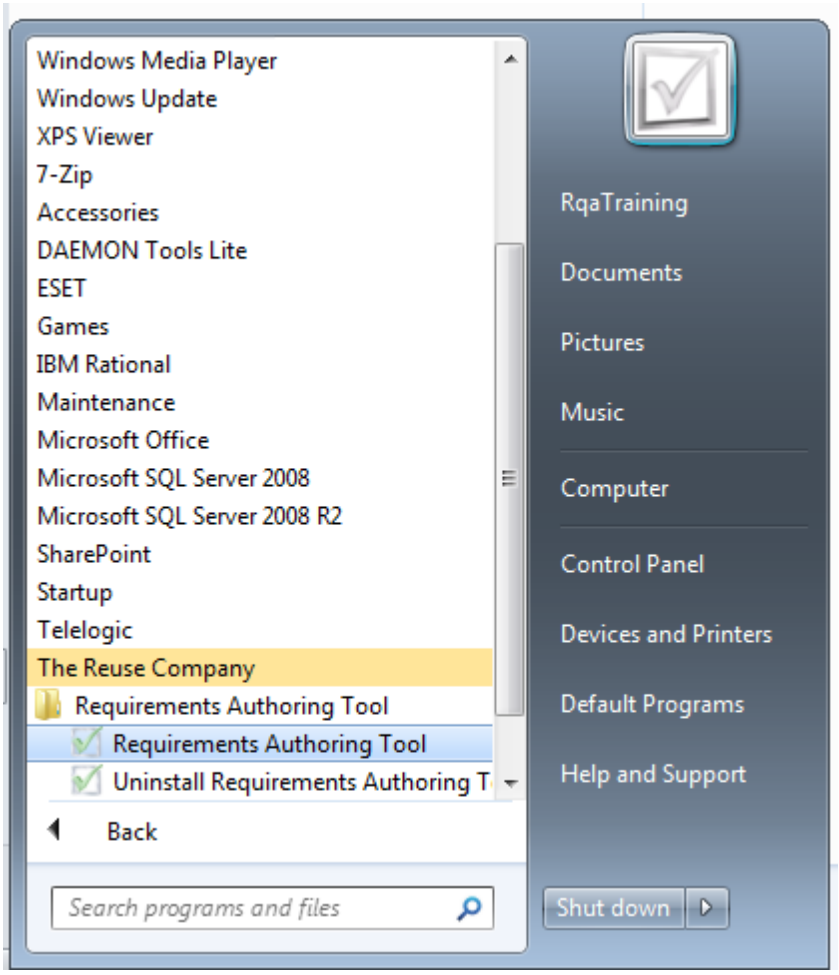
If you wish to exit the RQA Client at whichever time, please select the *Exit* tab on the left. The following dialog will appear:



- Click Exit Project if you wish to stay in the RQA Client and later open a new project.
- Click Exit RQA if you wish to exit the RQA Client.

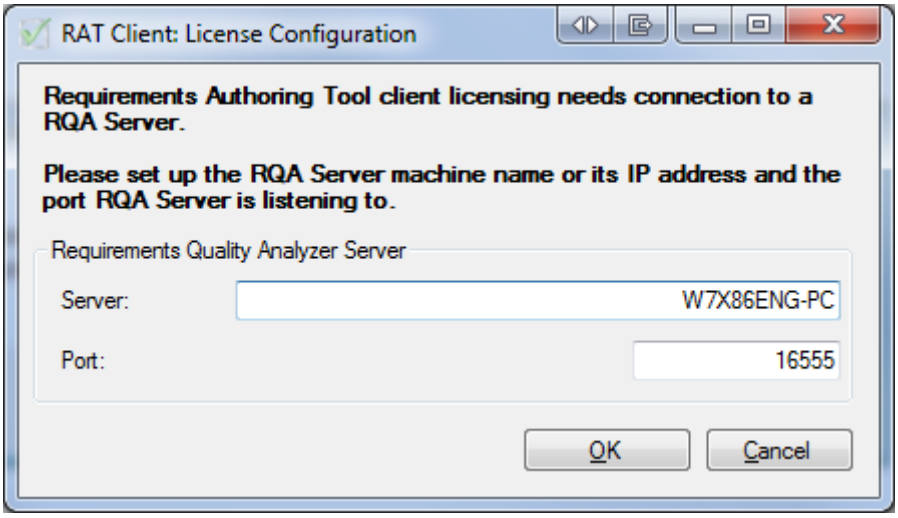
13. Accessing the RAT Client

The RAT Client can be launched using the Windows Start Menu



13.1 Providing a valid client license

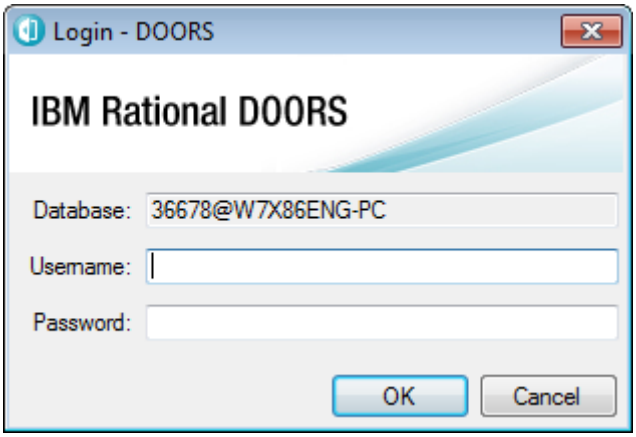
- Floating (server based) license:



1. Click on Configure *server*
2. Type the server name and the service port in the server
3. Default port value: 16555
4. Start the server and click on *Licensing* to see this values
5. Click *OK* in the License Configuration dialog

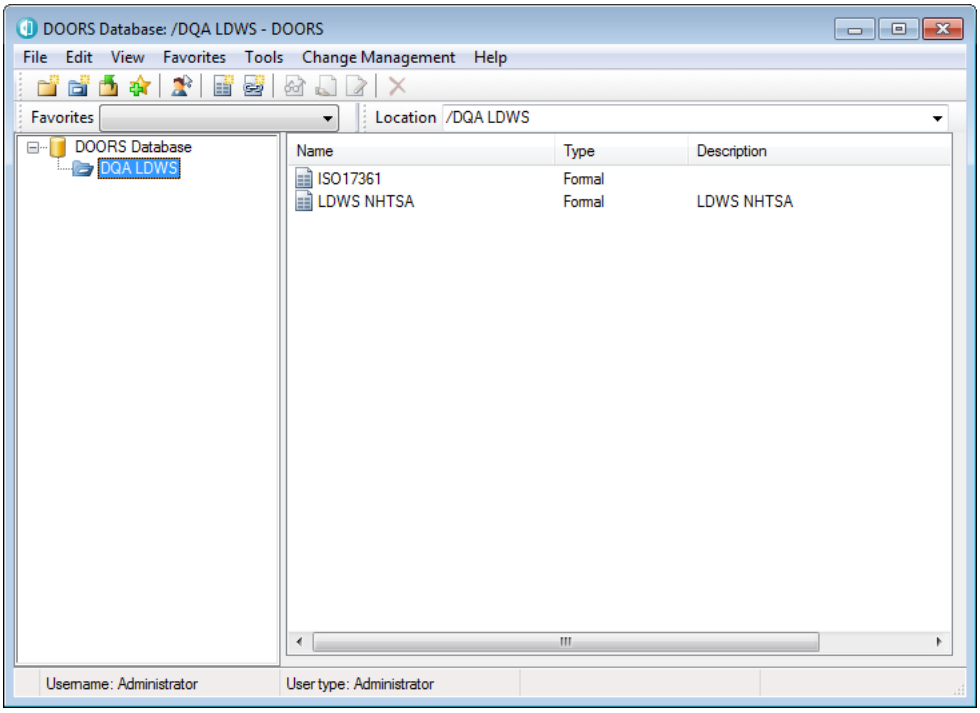
13.2 DOORS Log in

- Login in DOORS

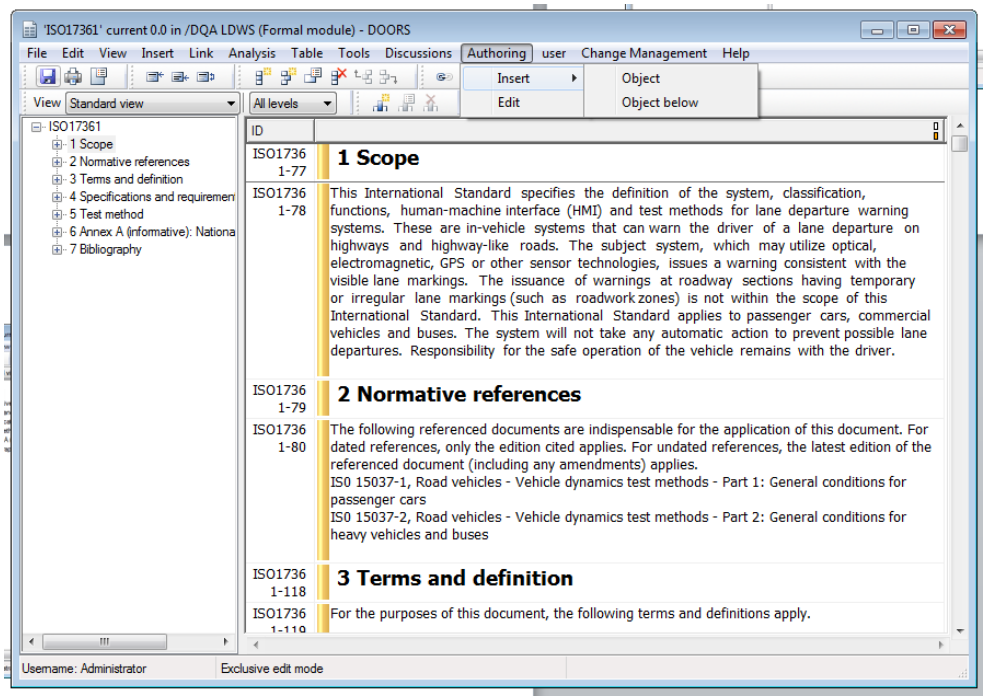


1. Insert username and password.
2. Click *OK*.

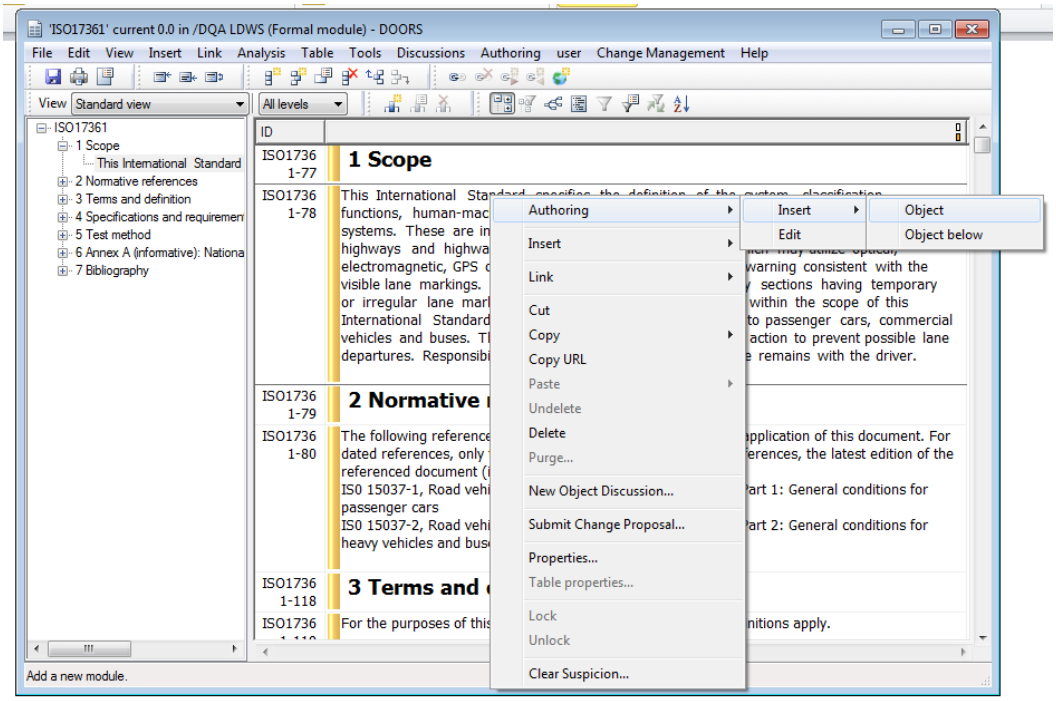
- Authoring option
    1. The user must navigate through your folders/subfolders and project structure to reach the module. Ensure that you have writing access to open that module with writing wrights.



2. Press Authoring option on the top menu

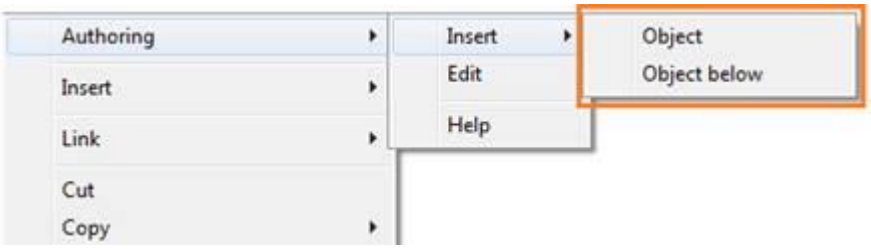


3. Another way to open the RAT tool is pressing the right mouse button on the requirement.

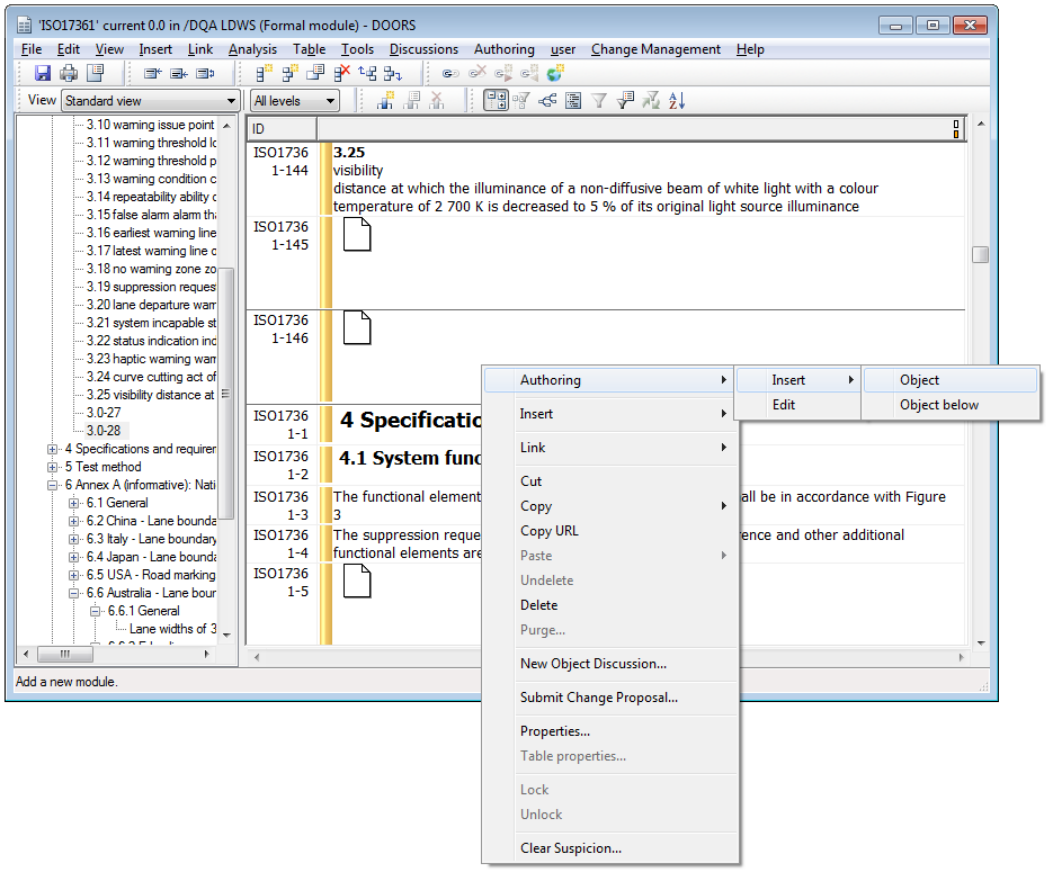


13.3 Adding a new requirement in DOORS

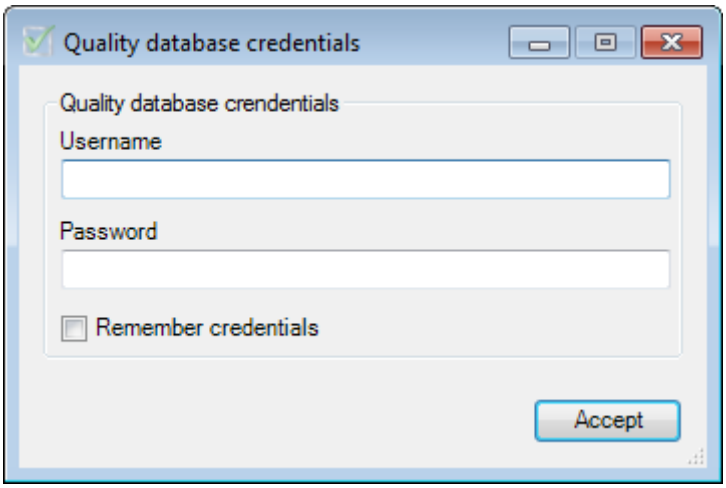
RAT allows the insertion of a new requirement either at the same hierarchical level with the selected one or one level below.



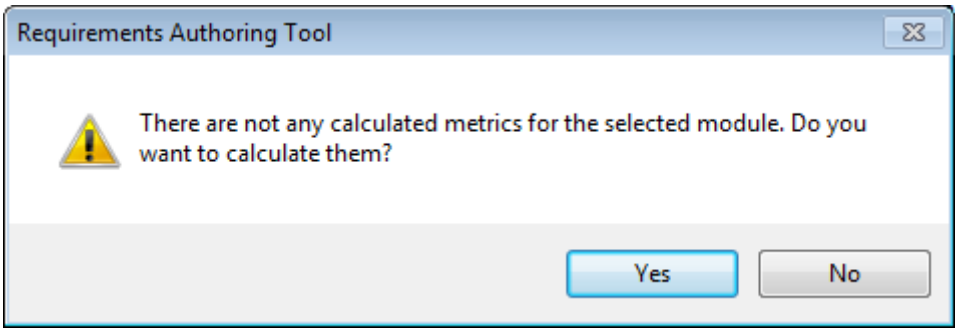
1. *Click Authoring > Insert > Object*



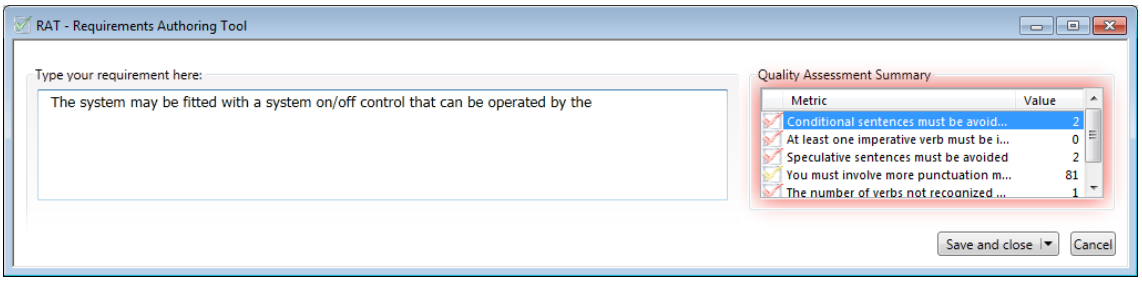
2. Type you quality database credentials: Username and Password. (Only necessary if the RQA Server has not stored the Quality database credentials before)



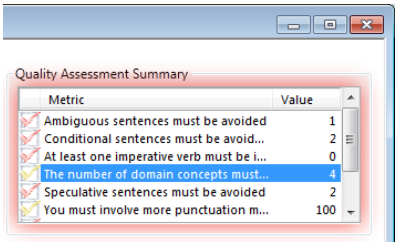
3. A confirmation window will appear if there is no quality information calculated yet, for the selected module



4. Type the text for your requirement. In the box on the right, you can see the quality summary

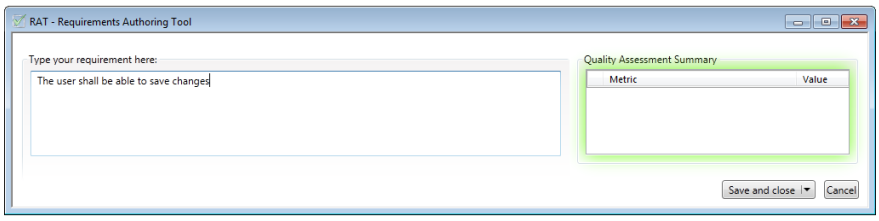


5. Icons show the quality level. Red icons show low quality and yellow icons show medium quality.

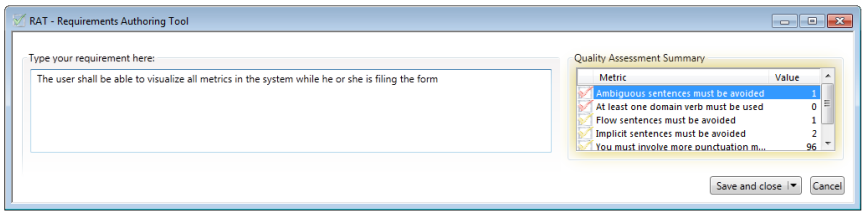


6. The Quality Frame is part of the right side of the RAT User Interface window that is showing the quality of the written requirement. The Quality Frame shows the requirement qualitative value as a coloured frame using the following colours:

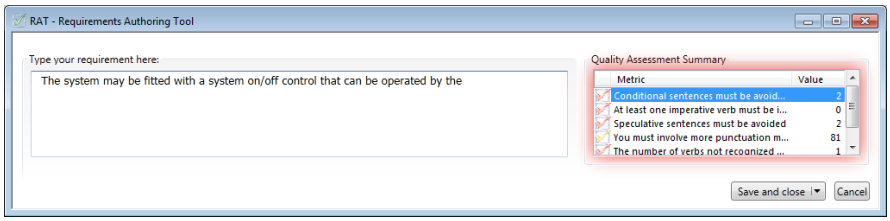
- **Green:** the quality of the requirement written in its text box has been assessed as high.



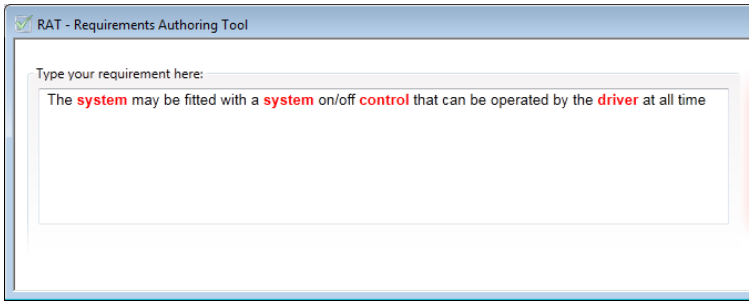
- **Yellow:** the quality of the requirement written in its text box has been assessed as medium



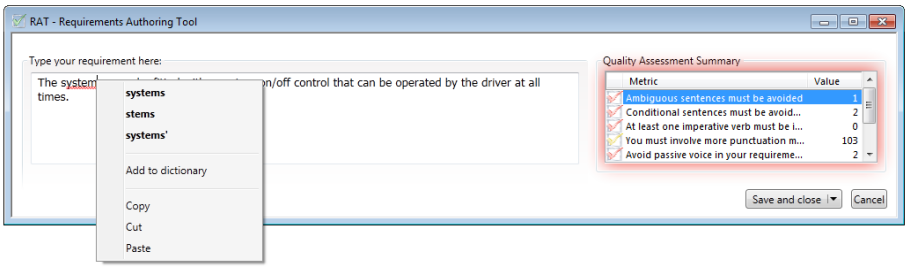
- **Red:** the quality of the requirement written in its text box has been assessed as low



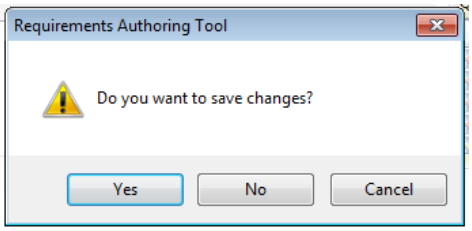
7. Based on the quality assessment, the text box shows in red color the words or terms that should be considered for re-authoring.



8. Based on a standard dictionary, the text box shows possible spelling errors, offering a solution similar to the one provided by Microsoft Word.



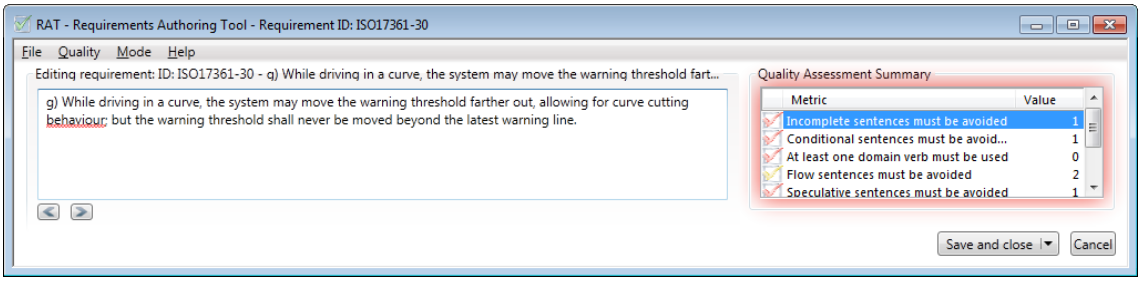
9. Saving your changes



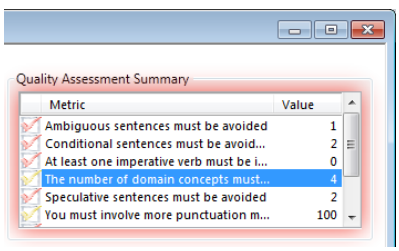
13.4 Editing a requirement in DOORS

1. *Click Authoring > Edit > Object*

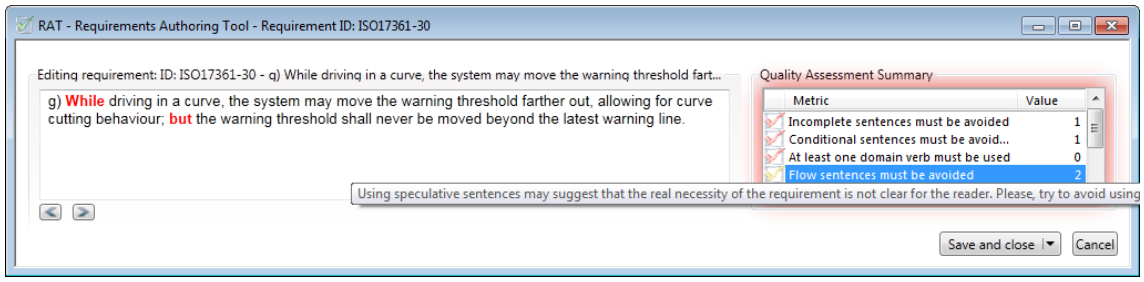
2. Edit your requirement



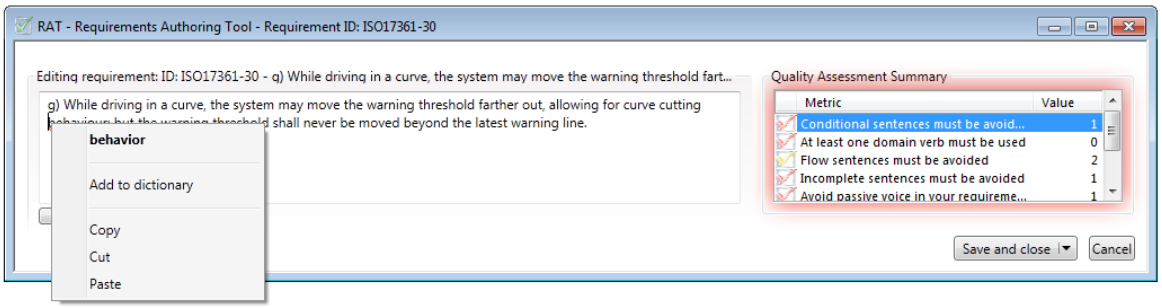
3. Icons show the quality level. Red icons show low quality and yellow icons show medium quality.



4. Based on the quality assessment, the text box shows in red color the words or terms that should be considered for re-authoring.



5. Based on a standard dictionary, the text box shows possible spelling errors, offering a solution similar to the one provided by Microsoft Word.



6. Buttons de *Previous* and *Next* allowed to move between the different requirements

7. Saving your changes