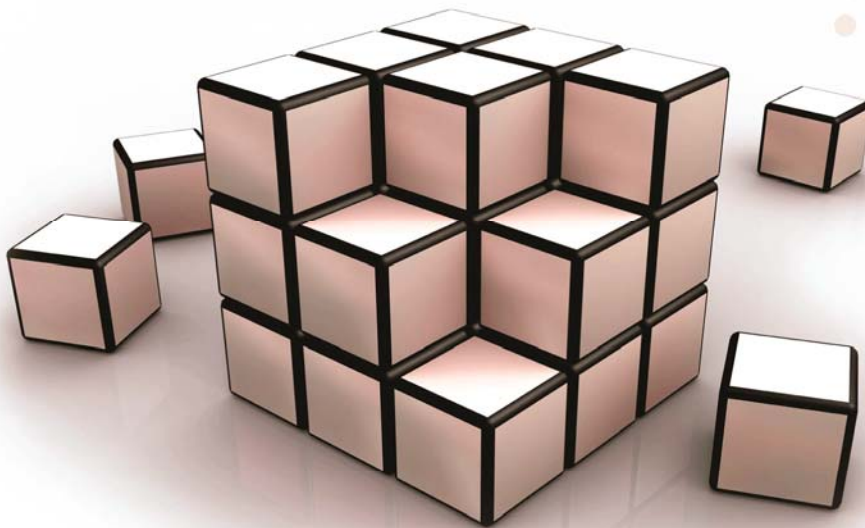




IRQA  **DOORS Integration**



**Requirements Management
and Engineering**





Copyright© 2007 Visure Solutions, S.L.
All rights reserved.

This document may not be reproduced or transmitted in any way or by any electronic or mechanical means, including photocopying, recording, or any other means, without the express permission of VISURE SOLUTIONS, S.L.

All of the material contained in this document has been developed by VISURE SOLUTIONS, S.L. and is the property of VISURE SOLUTIONS, S.L.

IRQA® is a registered trademark of VISURE SOLUTIONS, S.L.

The other products mentioned are the property of their respective companies.

Visure Solutions, S.L.

C.I.F.: B-85080554

Headquarters:

Av. de los labradores, 1, 4th Floor

28760 – Tres Cantos (Madrid)

Spain



Contents

1. INTRODUCTION	4
2. HOW TO RUN THE DOORS INTEGRATION	5
3. HOW TO USE DOORS INTEGRATION (IMPORT).....	7
3.1. Getting started.....	7
3.2. Importing Modules	9
3.3. Importing Objects (requirements).....	11
4. HOW TO USE DOORS INTEGRATION (EXPORT).....	14
5. MAPPING DOORS/ERS – IRQA	17
6. ADITONAL NOTES	21



Introduction

This document describes how to use the DXL files needed to import and export XRI files from and to DOORS. With these DXL files the user can import XRI files to DOORS and integrate it with his current projects and also export his requirements to his IRQA projects.



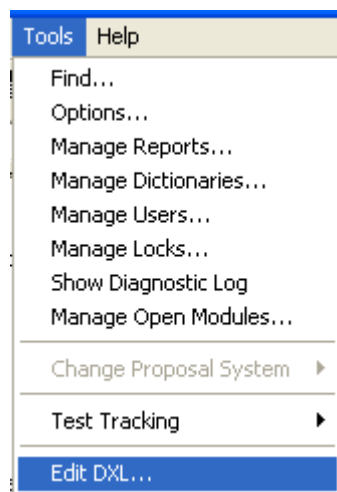
How to run the DOORS Integration

To Run the integration there is no need of any previous installation. You just need to register a dll on your system. To do this, simply go to the command line and execute the following line:

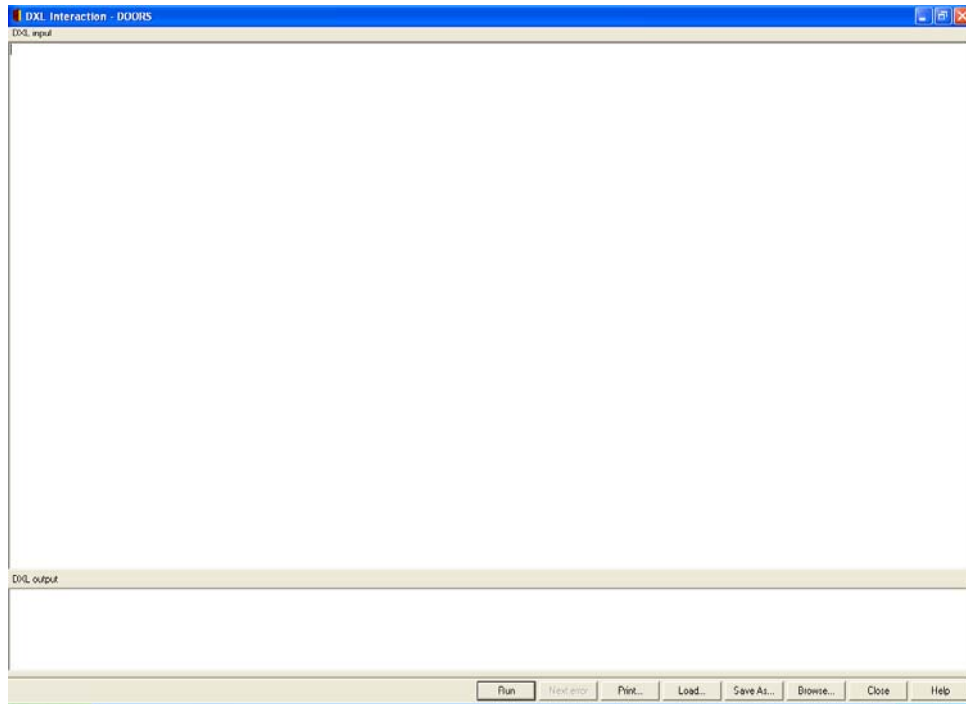
```
regsvr32 <Full_Path>\XRIData.dll
```

Take into account that XRIData.dll is distributed with the integration and that you should replace <Full_Path> with the full path to the folder where the file XRIData.dll is contained.

The integration between DOORS and XRI is based on DXL which is a programming language within DOORS. To run the import tool, click on the "Edit DXL..." option from the Tools menu in DOORS



This will lead to the following window



From this window click the “Load” button and select the corresponding XRI file, depending on the desired action (import or export). Once the script has been loaded in the screen, click on the “Run” button. This will start the integration. Note that the four files included in the .zip must be in the same folder.



How to use DOORS Integration (import)

Getting started

Open the “Edit DXL...” window and select “XRI2DOORS.L.dxl” file provided by Visure Solutions, S.L. as explained in the previous chapter.

There is a variable called `tracelvl` in the file:

```
Int tracelvl=1
```

You can use this variable to customize the trace level in the console. Just change the number for one of the following:

- 1: Error traces
- 2: Info traces
- 3: Warning traces
- 4: Debug traces

It is not recommended to use levels different from 1.

There are other variables called `CreateLog` and `CompareChanges` in the file:

```
bool CreateLog= true
```

You can use this variable to customize the log level in the console. Just change the value for one of the following:

True: A log will be generated out of the import process

False: A log will not be generated

WARNING: Generating a log file will consume memory.

```
bool CompareChanges= true
```

You can use this variable to select whether to perform a change comparison or not. Just change the value for one of the following:

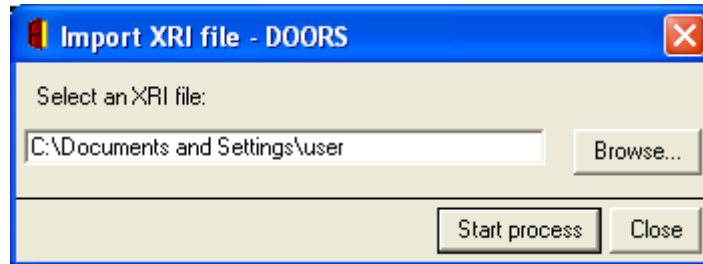
True: A change comparison between elements will be performed.



False: All existing elements will be detected as *Changed*, and will be updated.

WARNING: Establishing the *CompareChanges* and *GenerateLog* variables to *False* is recommended when managing extraordinary big xri files.

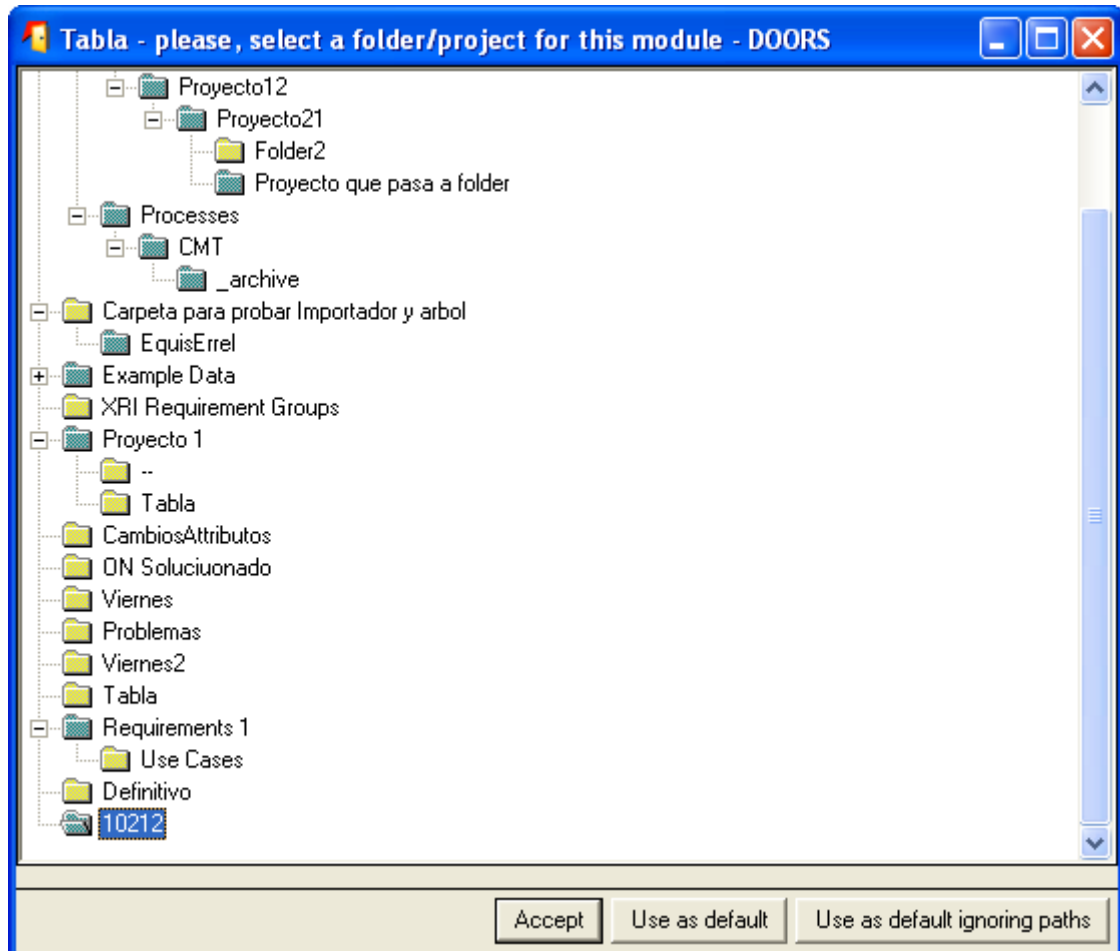
After running the file, the following dialog window will appear:



From this dialog box, select the desired .xri containing the information and click on the *Start process* button. If the modules in the file do not have a Path (which means that they haven't been in DOORS before), the following window will be shown:



Importing Modules



This dialog window allows the user to select the destination folder/project for the modules to be imported.

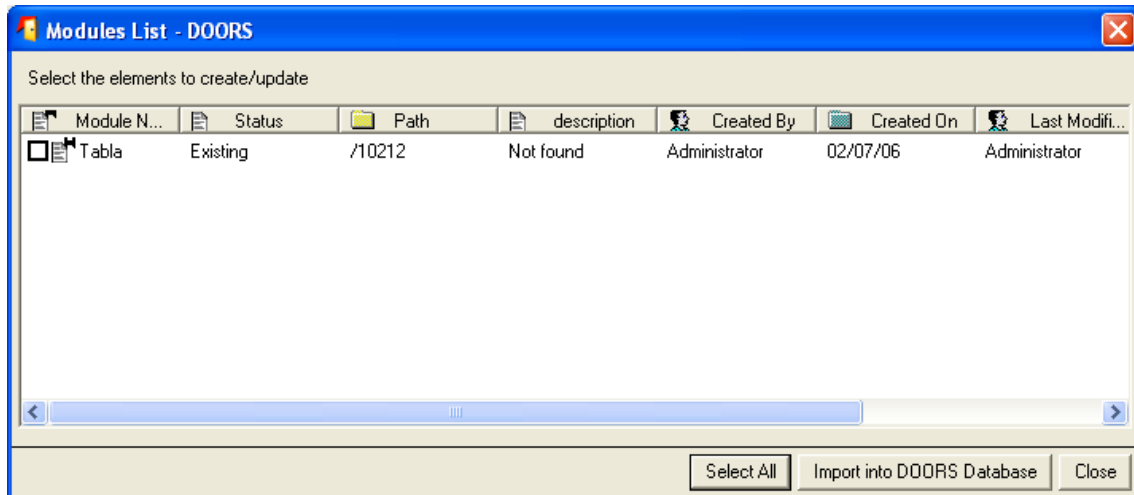
The following actions will be possible:

- Accept. The current module will be imported in the selected folder/project.
- Use as default. The current module will be imported in the selected folder/project, and the rest of the modules with no specified path will also be located in this folder/path.
- Use as default ignoring paths. All modules will be imported in the selected folder/project, ignoring their specified path, in case there is.



Note: A new folder (**XRI Requirement Groups**) will be created during the import process, so the users may use it as default.

After selecting a destination for the modules to be imported, the following window will appear.



From this window, the user will be able to select the desired modules to be imported as well as see associated information, like the current path and author of the modules.

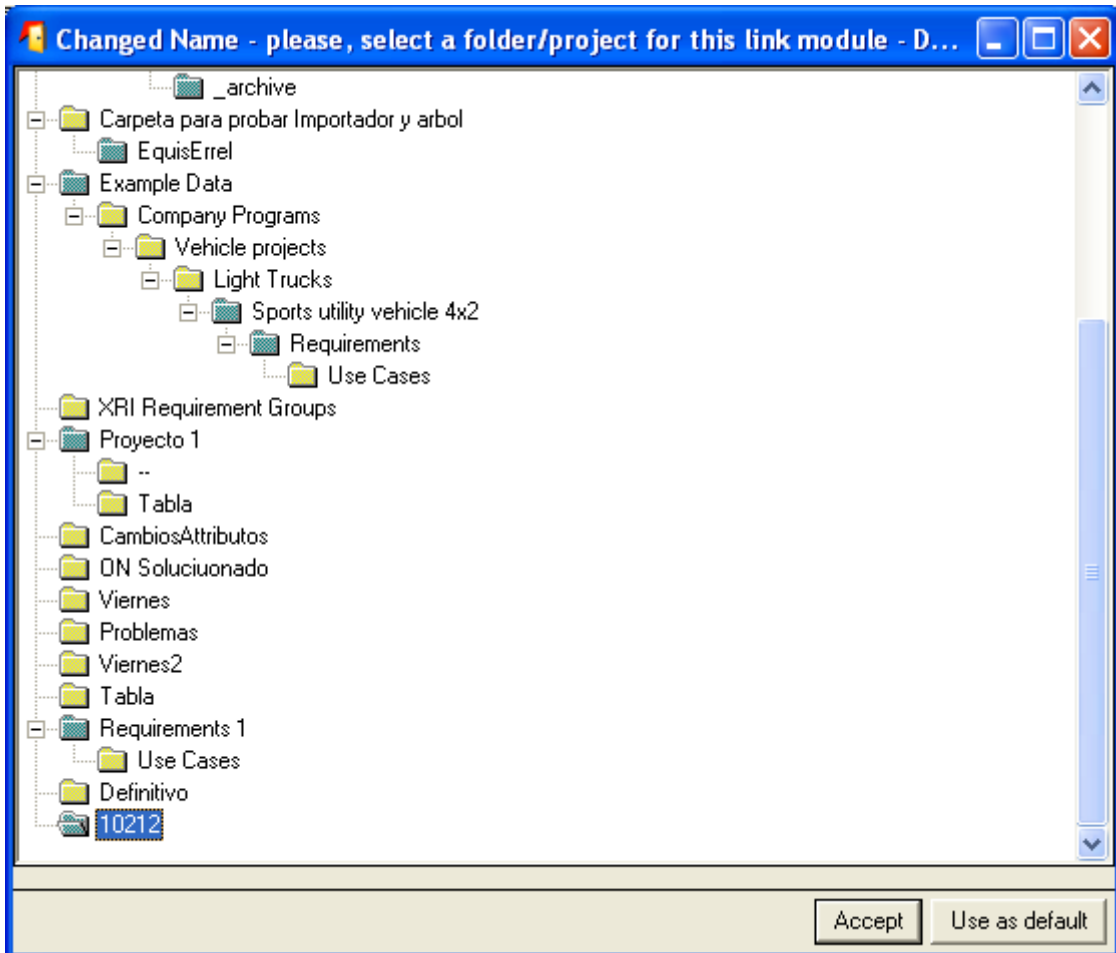
From this window the user will be able to see the status of the modules. This status can take three values:

- New: if the module is new and does not exist in DOORS.
- Existing: if the module already exists in DOORS.
- Deleted: If the module exists in DOORS but has been deleted.

After choosing the desired modules to be imported, the user must click the “Import into DOORS Database” button. This action can lead to different steps, depending on the status of the module:

- Deleted: If one or more of the selected modules have been deleted, the import tool will ask the user to select a different module.
- New: The module will be created and all its objects inserted in the module path.
- Existing: In this case, the user will be prompted to decide what to do with the modified elements.

After clicking on the “Import into DOORS database” button, the following dialog window will appear:

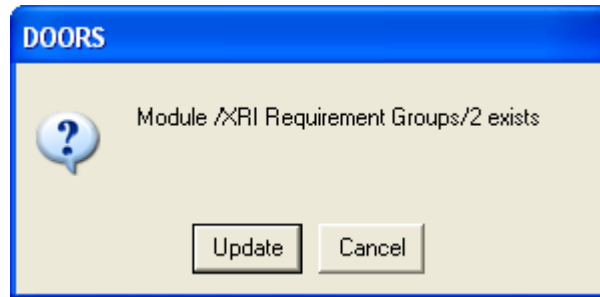


In this window, the user must select the path for the links modules contained in the file. At this step, the user can:

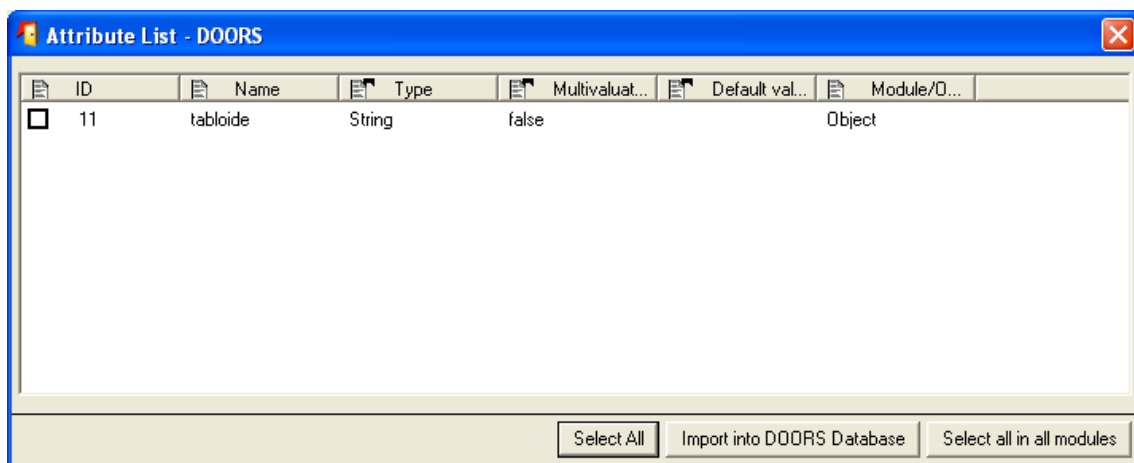
- Accept: Choosing the selected folder/project as the path for the current link module.
- Use as Default: Choosing the selected folder/project as the path for all the link modules in the file.

Importing Objects (requirements)

If the user has chosen to import an existing module, a confirmation dialog box will appear:



Clicking update will lead the user to the following window:



From this dialog box, the user will be able to choose the object attributes to be imported from the current module. Once again the user will be able to:

- Import into DOORS database. It will import the selected attributes for the current module.
- Select all in all modules. It will import all the attributes in all the modules.

After selecting one of the previous options, the following dialog will appear



ID	Status	Code	Name	Object Nu...	Description	Created By
13226	Unchanged	REQ-0002.01	Name_0002.01	0001	Desc_0002.01	Admin
13227	Unchanged	REQ-0002.01.01	Name_0002.01...	0001.0001	Desc_0002.01...	Admin
13228	Unchanged	REQ-0002.01.02	Name_0002.01...	0001.0002	Desc_0002.01.02	Admin
13229	Unchanged	REQ-0002.01.03	Name_0002.01...	0001.0003	Desc_0002.01.03	Admin
13230	Unchanged	0030	add	0001.0004	ADD	Admin

In this window the user will be able to select the desired requirements to be imported into DOORS. These requirements can have three statuses:

- **Changed:** If the requirement has changed in the XRI file. This status will be shown in the following cases:
 - New Attributes have been added to the module/requirement.
 - One or more existing attributes have changed their values.
 - The code of the requirement has changed.
 - New relationships involving these requirements (as target or source) have been added/changed.
- **Unchanged:** If the requirement is identical to one in the DOORS Database. In this case, nothing will be imported.
- **New:** if the requirement does not exist in the module.

The user may select a list of changed or new requirements (if an unchanged requirement is selected, a warning message will appear) or all of them by clicking in the “Select All” button. After that, clicking the “Import into DOORS Database” will cause the selected requirements to be imported or updated in the DOORS database

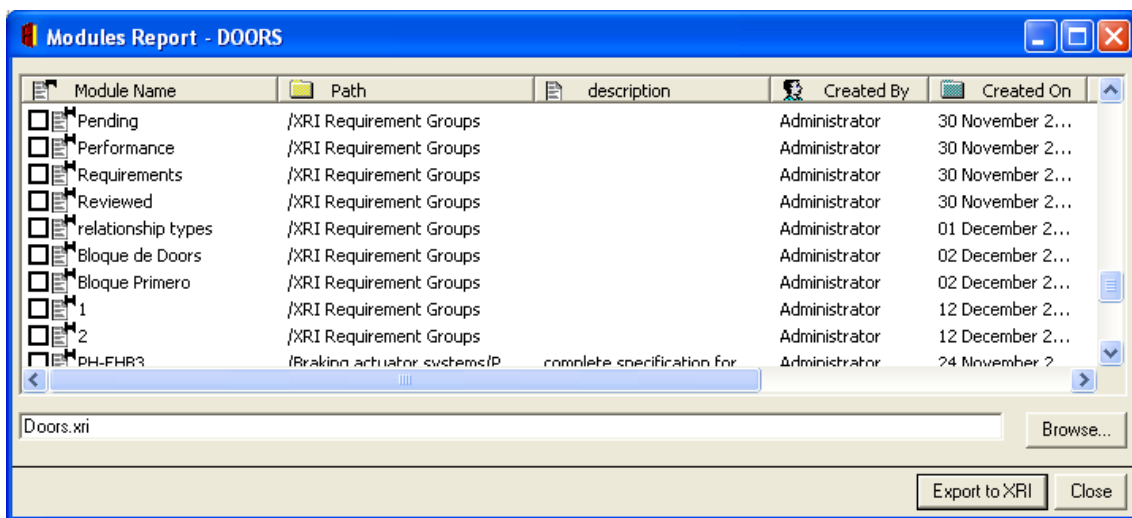
Note: In order to import new and changed relationships, the elements involved must exist in the destination. Therefore, either both elements are imported along with the relationship, or if they already exist in the destination, only the relationship along with one of the elements can be imported.

Note2: Although hierarchy changes will not mark an element as changed they will be imported if the element with hierarchy changes is selected to be imported (but just for the selected element)



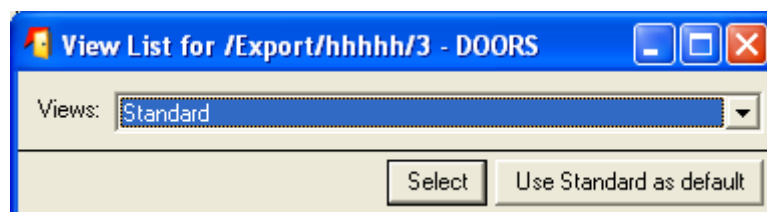
How to use DOORS Integration (export)

To run the DOORS integration to export, the user must load the “DOORS2XRIL.dxl” file and run it, as described in chapter 2. The following window will appear



This dialog will show only the modules belonging to the actual folder/project and all his folder/project (the one selected in the main window browser). If the user wants to view all modules, simply use the DOORS Database (“/”) as the actual folder. To do this, click on the DOORS database folder in the main window’s Browser

From this window, the user will be able to select the desired modules to be exported. After that, the user must select a destination file for the XRI and then click on the “Export to XRI” button. This will show the following dialog box





The user will then be able to choose the attributes to export. To do this, the user must select a view containing the desired attributes, clicking either on the *Select* button, to use a specific view, or click on *Use standard as default* to use the standard view for all modules.

Note: The standard view will export all the attributes in the module.



Mapping between DOORS ERS and IRQA

DOORS ERS	XRI	IRQA
Field/Element	Field/Element	Field/Element
Formal Modules	Requirement Groups	Blocks
Prefix	Ignored	Ignored
Name	Name	Name
Description	Description	Description
Module version number	Ignored	Ignored
Created By	createdBy	Ignored
Created On	createdOn	Ignored
Last Modified By	Ignored	Ignored
Last Modified On	Ignored	Ignored
Formal Module Path within the DOORS Database	User Defined Group Attribute: DOORS_Path	User Defined Block Property: DOORS_Path (read-only)



Objects	Requirements	Requirements
Absolute Number	User Defined Element Attribute: DOORS_DBID_Absolute_Number	User Defined Element Attribute: DOORS_DBID_Absolute_Number
Object Number	User Defined Element Attribute: DOORS_Object_Number	User Defined Element Attribute: DOORS_Object_Number
Object Heading	name	Name
User Defined Element Attribute: XRICode	code	Code
Object Short Text	DOORS_Short_Text	DOORS_Short_Text
Object Text	Description	Description
Created Thru	Ignored	Ignored
Created By	createdBy lastVersionCreatedBy	Ignored
Created On	createdOn lastVersionCreatedOn	Ignored
Last Modified By	lastModifiedBy	Ignored
Last Modified On	lastModifiedOn	Ignored
Does not exist	DOORS_Sort_Number	DOORS_Sort_Number



Does not exist	DOORS_Table	DOORS_Table
TableBottomBorder	DOORS_TableBottomBorder	DOORS_TableBottomBorder
TableLeftBorder	DOORS_TableLeftBorder	DOORS_TableLeftBorder
TableRightBorder	DOORS_TableRightBorder	DOORS_TableRightBorder
TableTopBorder	DOORS_TableTopBorder	DOORS_TableTopBorder
<hr/>		
User defined attributes	Attributes	User defined attributes
User defined object attribute	Attribute	User Defined Element Attribute
User defined module attribute	Attribute	User Defined Block Property
User defined multi valued object attribute	Attribute	Attribute
User defined multi valued module attribute	Attribute	Not supported
DXL attribute	Ignored	Not supported
Default value	Default value	Default value
<hr/>		
Base Types	Types	Base Types
Integer	Integer	Integer
Real	Real	Real



String	Text	Text
Text	Text	Partially supported (it will become plain Text)
Date	Date	Date
Enumeration (color and related number supported)	Enumeration (color and related number supported)	Enumerate (only color is supported)
User defined Types		
User defined Types using base types	Types	User defined Types using base types
Link Modules		
Name	name	Name
Description	Ignored	Ignored
Mapping (one-to-one, many-to-many)	Ignored	Not supported
Attributes	Ignored	Not supported
Links	Relationships	Relationships by motive
Relationships Types		
Motives		



Additional Notes

The user should take into account the following consideration:

- The user importing the XRI files into IRQA must have write access rights over the attribute "DOORS_Absolute_Number". Additionally, this attribute must not be edited.
- An attribute "DOORS_DBID_Absolute_Number" will be created for each DB in which the modules existed. *DBID will be replaced by the Database ID*
- The user must not move requirements in IRQA from one block to another or rename blocks.
- Attributes starting with DOORS_ prefix are not intended to be edited.
- Users shouldn't modify the multi-valued attributes in DOORS, as it may affect the IRQA block structure.
- The same requirement must not be assigned to different modules in DOORS.
- The first time requirements are re-imported into DOORS, they will appear as *Changed*. A new attribute (XRI Requirements Group) will indicate that they are being part of an exchange process.
- Relationships involving requirements from different modules will only be imported if both modules have been selected for importing.
- All the relationship modules must be imported into the "/XRI Requirement Group" folder, otherwise, duplicated relationships may appear after importing an XRI file that was previously exported.
- The user exporting data from DOORS into XRI must have write access rights over the exported modules, even though the objects contained in the modules will not be changed.
- Users should not add new values to an enumerated type in IRQA if they intend to be export it again into DOORS.
- Users should not have two or more attributes with the same name and different types in different DOORS modules.
- Users importing data from XRI files to DOORS must have write access rights over the modules where the importation is taking place.
- XRI import in IRQA 3.3.1 release is only compatible with XRI files generated with the same release.
- Requirements with no name in DOORS will be renamed with a hyphen "-" in XRI. In the same way, requirements with a "-" name in XRI will be imported in DOORS with no name.
- Changes in hierarchy will be imported if the changed elements are marked to be imported but that changes will not mark the element as changed