



QS_Toolbox

Table

Introduction	3
What is QS_Toolbox	3
Demonstration version vs Commercial version	3
Component installation	4
Information structure in QS_Toolbox	5
General functions.....	6
Method management	7
Method Overview	7
Buttons bar	8
Search area	8
The magnifying glass	8
The pen.....	9
The printer.....	9
The page	9
Eyes, cursor and folder.....	10
Method Analyzer	12
4D Command	13
Methods Pallet.....	14
Form Explorer	15
Structure Analysis	17
Structure	17
Display table & field definition data	17
Specific options	18
Table Relations.....	18
Search & Control	19
Check & Update	21
4D Link Editor.....	23
Converter EN / FR.....	24
Version Control System	26
Reports	28
Setup	29
New DB Analysis	29
Restart 4D.....	29

Introduction

QS_Toolbox was developed with 4D v17 and compiled in 4D v17.3.

Because it follows the standard version branch, it can be used without any problem on the R version branch.

What is QS_Toolbox

It is a component for 4D developers to use when analyzing their code, transferring code or migrating.

QS_Toolbox allows in-depth analysis of the structure in order to help you detect possible errors and programming difficulties.

QS_Toolbox does not modify your data without your permission. Except for the method parameters and the replacement of "disapproved" commands, no changes can be made in your code.

Demonstration version vs Commercial version

QS_Toolbox is free, and the source code is available up to version 2.2 of the component.

Since version 3.0, QS_Toolbox has become a commercial component whose cost is excessively low compared to the services it is able to provide you.

The demo version is fully functional for 60 days.

Each time the component is launched, it connects to our servers to check the validity of the license. Aware that you may be on the move and without a temporary connection, the latter accepts 10 offline launches before "blocking" until a new connection is detected.

Component installation

The component (or its shortcut / alias) must be placed in the 'Components' folder of your application to scan.

To start it, you can set the following code in 'on startup database' method:

```
ARRAY TEXT($tTxt_Components;0)
COMPONENT LIST($tTxt_Components)
If (Not(Is compiled mode))

    // Help to get back the hand if an issue on startuo
    If ((Application type=4D Local mode) & (Is compiled mode=False) & (Shift down))
        TRACE
    End if







    // ASSERT on
    SET ASSERT ENABLED(True) // - Actif par défaut

    // QS_Toolbox
    If (Find in array($tTxt_Components;"QS_Toolbox@")>0)
        EXECUTE METHOD("QS_Toolbox")
    End if

Else
    // ASSERT off
    SET ASSERT ENABLED(False)
End if
```

At the first launch, a folder is created on the local disk and located:

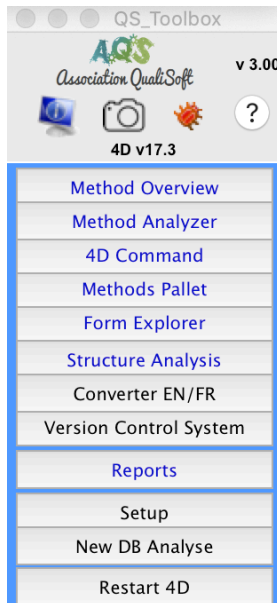
- PC : 'drive':\my documents\QS_Toolbox\<base name>
- Mac : /documents/QS_Toolbox/<base name>

-  All Project Methods
-  Export
-  Export JSON
-  Export_HTML
-  Forms
-  Report Validation

In the folder <base name>, several sub-folders are created to organize the output information.

A data file is also created in this folder and is used to manage parameters and save methods as part of version management. This data file is called 'QST_External_DB'.

Information structure in QS_Toolbox



After launch, the component is presented in the form of a pallet accessible in structure mode.

The entire interface is in English.

All different options are accessible through menus. Those in blue are only accessible once the startup database analysis is completed. The duration of this analysis depends on the complexity of the structure (number of tables, forms, methods, ...).

The themes of QS_Toolbox are:

- Methods Analysis
- Analysis of Forms
- Analysis of the database structure
- Code conversion (EN <-> FR)
- 4D method version management
- Automated reporting

Note: When the component is launched, it starts the analysis of your database. This analysis can temporarily slow down your application and allow a smooth navigation between the different elements of the component.

In the event of a major change, you can restart the analysis of your database at any time. To do this, we recommend that you close all windows of the component.

General functions



The upper part of the tool palette provides you with the following information:

- Name of the analyzed application (here QS_Toolbox itself)
- Version number of the component (here 3.00)
- 4D version number used (here v17.3)

The available tools are, from left to right:

- Storage of system information
- Screen copy automatically saved in the Log folder of the database
- Access to the site of the "bugs list" of the major version of 4D in progress
- The 'About' of the component.

The license number is also entered from this location

Generally speaking, a double click on the name of a method opens it in the 4D editor. In case of changes, they will only be included in the QS_Toolbox analysis after reanalysis of your database. This has been done voluntarily so as not to have to create a process that runs in the background and slows down your application.

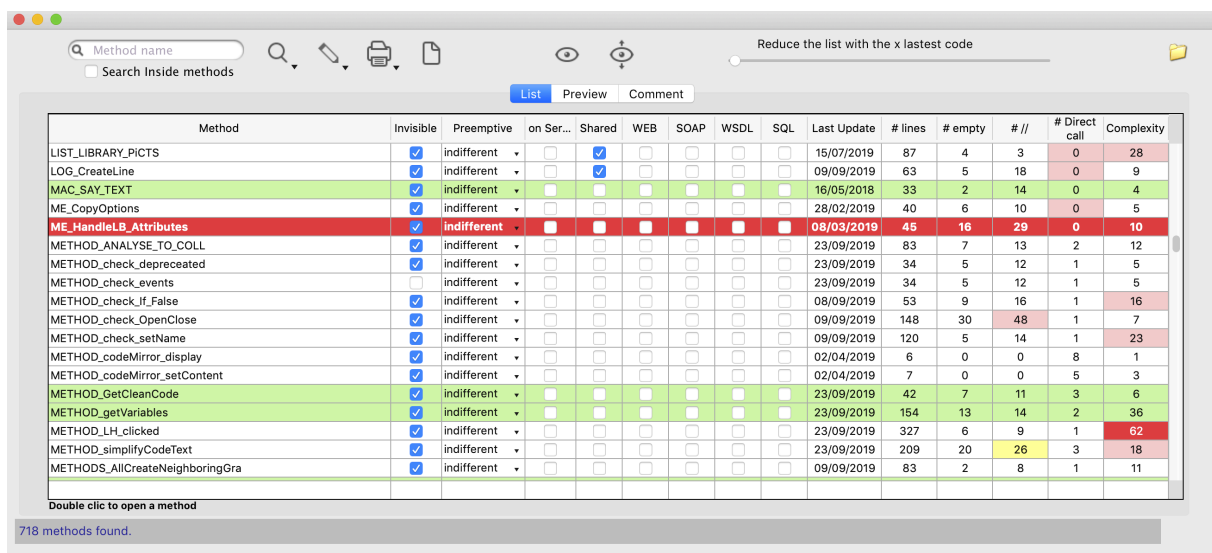
Method management

The management of methods is divided into 4 distinct parts that we will study below:

- Method Overview: view a set of statistics on methods and modify their properties
- Method Analyze: in-depth analysis of your methods
- 4D command: lists methods from 4D commands and also provides other information
- Pallet methods: quick access to all your code, including form widgets



Method Overview



Method	Invisible	Preemptive	on Ser...	Shared	WEB	SOAP	WSDL	SQL	Last Update	# lines	# empty	# //	# Direct call	Complexity
LIST_LIBRARY_PICTS	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	15/07/2019	87	4	3	0	28
LOG_CreateLine	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019	63	5	18	0	9
MAC_SAY_TEXT	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	16/05/2018	33	2	14	0	4
ME_CopyOptions	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	28/02/2019	40	6	10	0	5
ME_HandleLB_Attributes	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	08/03/2019	45	16	29	0	10
METHOD_ANALYSE_TO_COLL	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	83	7	13	2	12
METHOD_check_deprecated	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	34	5	12	1	5
METHOD_check_events	<input type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	34	5	12	1	5
METHOD_check_if_False	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	08/09/2019	53	9	16	1	16
METHOD_check_OpenClose	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019	148	30	48	1	7
METHOD_check_setName	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019	120	5	14	1	23
METHOD_codeMirror_display	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	02/04/2019	6	0	0	8	1
METHOD_codeMirror_setContent	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	02/04/2019	7	0	0	5	3
METHOD_GetCleanCode	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	42	7	11	3	6
METHOD_getVariables	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	154	13	14	2	36
METHOD_LH_clicked	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	327	6	9	1	62
METHOD_simplifyCodeText	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019	209	20	26	3	18
METHODS_AllCreateNeighboringGra	<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019	83	2	8	1	11

Double clic to open a method

718 methods found.

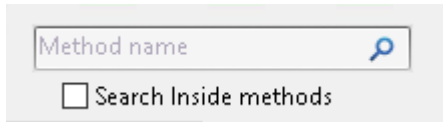
This is the basic module of the component for method management.

This module allows you to, among other things:

- View and modify the parameters of your methods
- To visualize the statistics carried out on the methods
- To visualize the method
- To edit the comments of the method
- Export the catalogue of your methods in HTML format

Buttons bar

Search area



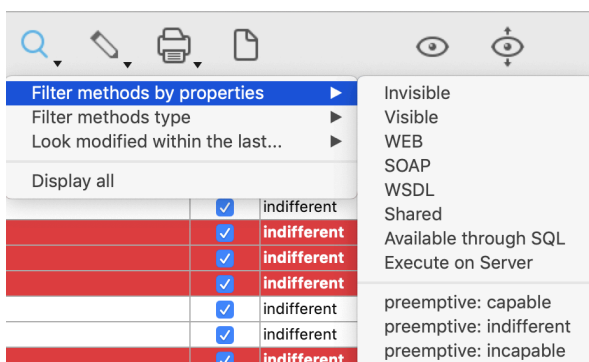
Method name

☐ Search Inside methods

Allows you to reduce the list of displayed methods by searching on the name and content of the method, if the option is enabled.

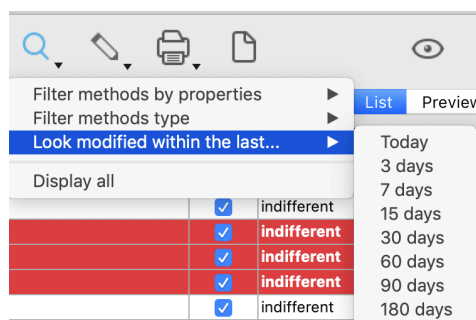
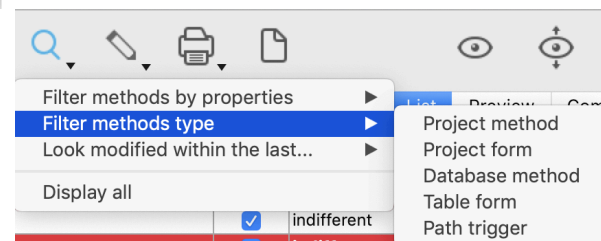
The magnifying glass

The button bar associated with this module allows you to quickly make selections:



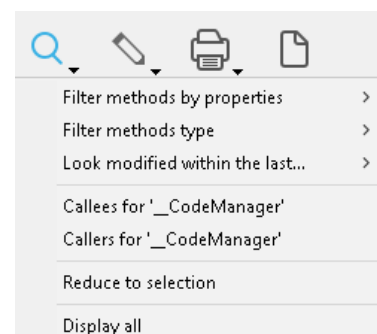
Selection based on method properties

Selection based on the nature of the method



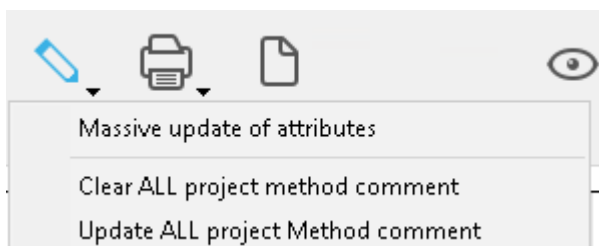
Selection based on the age of the methods

Note that if a method is selected, access to "Called methods" and "Calling methods" is possible. In this case, the content of the list is updated.



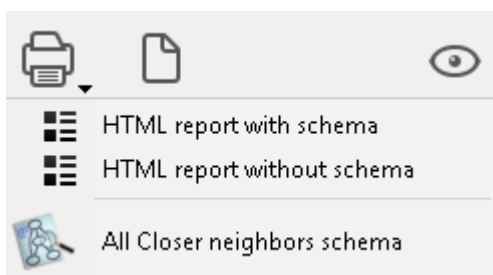
The pen

The pen allows you to make a generalized modification of the properties of the selected methods or to act directly on the comments of the methods.



- Deleting comments from all methods
- Integration of comments located at the beginning of the code as method comments. By default, only the first 10 lines are taken into account (adjustable option in the Setup menu).

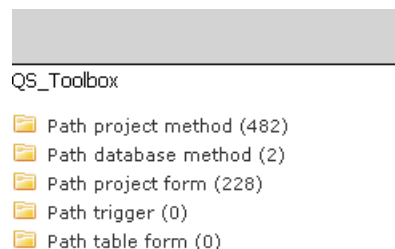
The printer



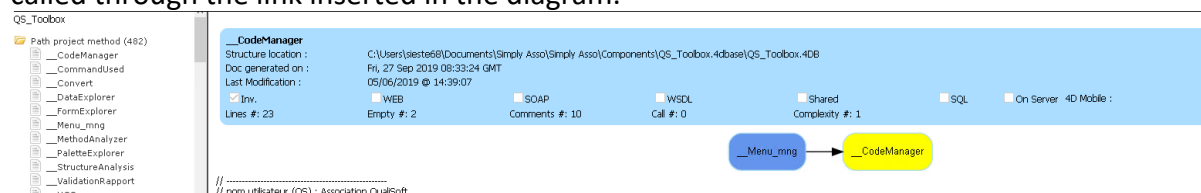
Allows the HTML export of the catalog of methods with or without schema generated using Graphviz that must be installed on your computer.

At the end of the export, you will be able to navigate through the methods between them.

The export creates a menu and shows you the number of methods available depending on their location.



Clicking on the name of the method displays the information about it and the callers and called through the link inserted in the diagram.



The page

Allows you to copy the list in memory in order to paste it into a spreadsheet.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Method list from QS_Toolbox: 27/09/2019 @ 10:59:28													
2														
3	Method name	Attr. Invisibl	Attr. On Serv	Attr. Shared	Attr. Web	Attr. SOAP	Attr. WSDL	Attr. SQL	Last Update	Nb lines	Nb empty	Nb Commen	Nb Nested	Complexity
4														
5	_Menu_mng	VRAI	FAUX	FAUX	FAUX	FAUX	FAUX	FAUX	25/09/2019	218	9	12	16	54
6	_Start	VRAI	FAUX	FAUX	FAUX	FAUX	FAUX	FAUX	25/09/2019	47	5	12	1	3
7	FE_analyseElementClicked	VRAI	FAUX	FAUX	FAUX	FAUX	FAUX	FAUX	26/09/2019	224	5	8	2	35
8	QS_PARAM_AFFECTATION	VRAI	FAUX	FAUX	FAUX	FAUX	FAUX	FAUX	25/09/2019	22	2	10	2	1
9	QS_PARAM_SAVE	VRAI	FAUX	FAUX	FAUX	FAUX	FAUX	FAUX	25/09/2019	49	6	10	1	5
10	QST_DB_SetUp	VRAI	VRAI	FAUX	FAUX	FAUX	FAUX	FAUX	25/09/2019	91	8	11	3	8
11	[projectForm]/_FormExplorer/myLH								26/09/2019	53	2	1	0	13
12	[projectForm]/_Setup/RestoreSetUp								25/09/2019	4	0	0	0	1

Eyes, cursor and folder

The first eye is used to reduce the displayed list to the manual selection made in the method list. This allows you to compare properties of methods that must share the same methods more quickly.

The second eye is used to retrieve the list of methods.

The cursor allows you to reduce the list of methods based on the date they were modified. This makes it possible to identify very quickly the last methods affected.

The folder provides quick access to the information stored on the local hard disk:

- Windows: 'drive':\my documents\QS_Toolbox\<base name>
- Mac: /documents/QS_Toolbox/<base name>

Statistics' interpretation

Colour codes can be used to draw your attention to different elements:

# lines	# empty	# //	# Direct call	Complexity
87	4	3	0	28
63	5	18	0	9
33	2	14	0	4
40	6	10	0	5
45	16	29	0	10
83	7	13	2	12
34	5	12	1	5
34	5	12	1	5
53	9	16	1	16
148	30	48	1	7
120	5	14	1	23
6	0	0	8	1
7	0	0	5	3
42	7	11	3	6
154	13	14	2	36
327	6	9	1	62
209	20	26	3	18
83	2	8	1	11

The statistics correspond to:


- Total number of lines per method
- Number of empty lines
- Number of comment lines
- Number of method calls
- Relative complexity of the method

A cell highlighted by a color quickly allows you to visualize if a verification action is to be considered on your method. 3 color levels have been defined: yellow, pink and red.

Invisible	Preemptive	on Ser...	Shared	WEB	SOAP	WSDL	SQL	Last Update
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	15/07/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	16/05/2018
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	28/02/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	08/03/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019
<input type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	08/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	02/04/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	02/04/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	23/09/2019
<input checked="" type="checkbox"/>	indifferent	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	09/09/2019

The method properties are accessible and can be modified directly from the graphical interface.

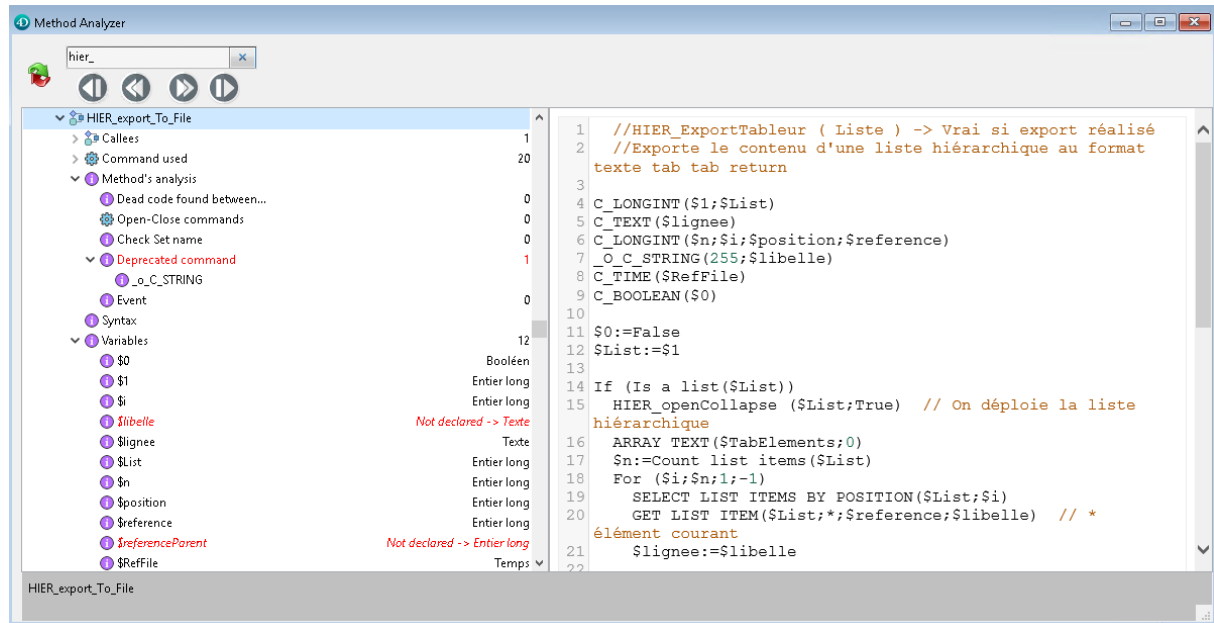
A simple click allows you to modify the desired property.

In case of modification, the icon  appears in the top left corner of the module to save the modifications.

A *green line* indicates that the 'Preemptive' property of the method can be set to 'capable'. This indication is only available if the typing file was generated during compilation. Similarly, a line not highlighted in green and whose "Preemptive" property is "capable" must be changed to "indifferent" unless you correct the code and restart the compilation for verification.

A *red line* indicates that the method is "useless" because the sum of the empty and commented lines is equal to the total number of lines of code of the method.

Method Analyzer



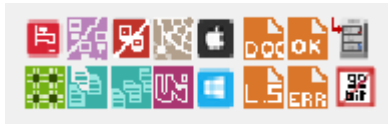
Second module on methods. The latter lists your methods using a hierarchical list, much like the method explorer, with the following differences:

- You have access to calling and called methods directly from the interface
- You have access to the list of commands used in the code
- You have access to the list of constants used in the code
- An analysis of your method is carried out in order to highlight points of attention such as the presence of dead code, the verification of the names of assemblies, the use of "end-of-life" commands...
- You have access to the syntax of your method*
- You have access to the list of variables of your method and their declaration. In case of non-reporting, QS_Toolbox offers you the type to report. *

* If the generation of the symbol file was performed during compilation, the information presented by QS_Toolbox is more relevant.

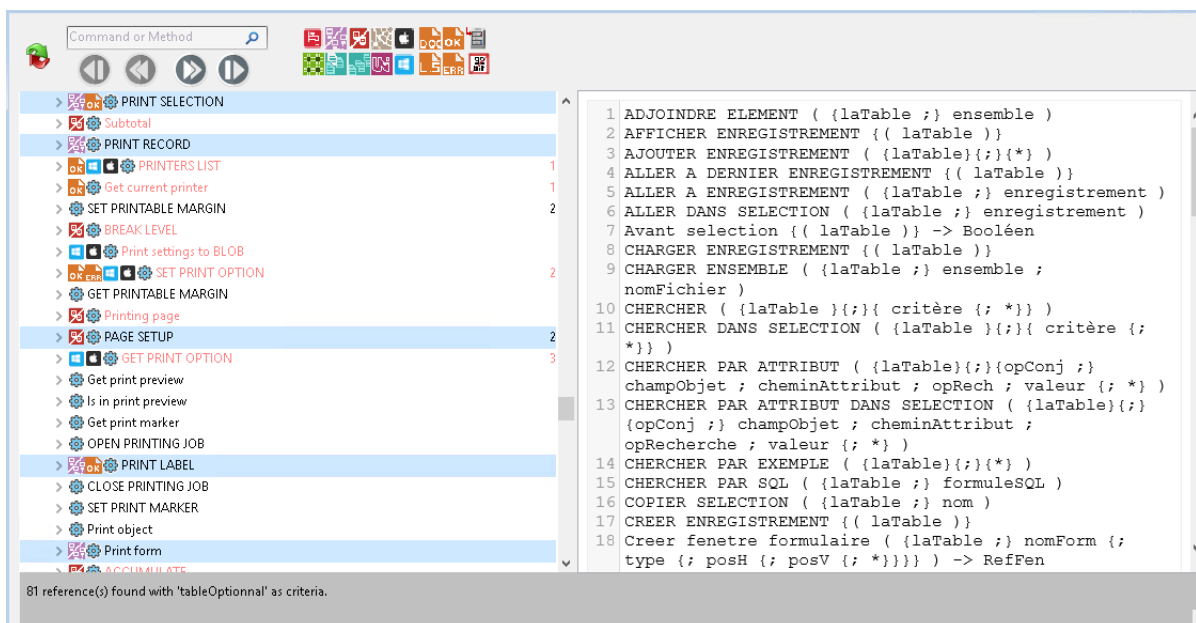
4D Command

This module allows you to list the methods used through the 4D command catalog, but also to list commands with a behavior such as a different client-server operation, those that modify the OK variable, ...

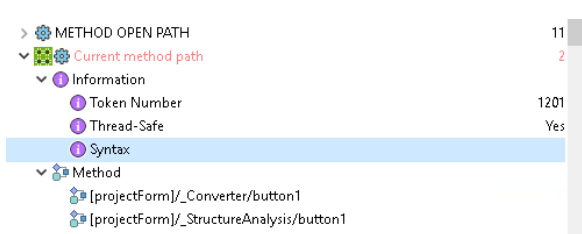


The buttons are "identical" to those found in 4D's documentation (<https://doc.4d.com/> or <https://livedoc.4d.com/>)

Clicking on one of the buttons displays the relevant commands and selects them for quick navigation between them.



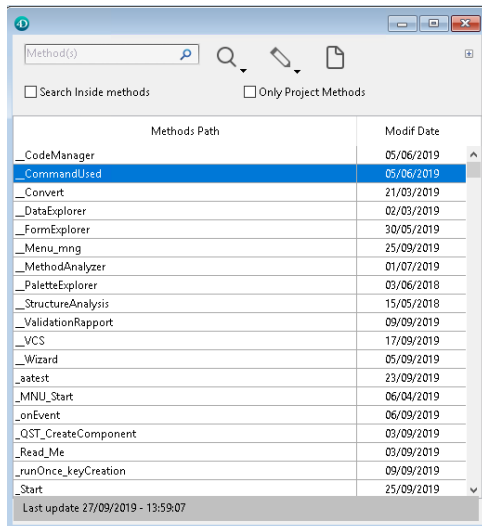
Example of selecting commands whose table specification is optional when using "Default table".



Selecting a command allows you to quickly know the number of methods using it and to extract:


- His token
- Its compatibility to be Thread-safe
- Its syntax
- The list of methods using this command, the code of which is displayed in case of selection

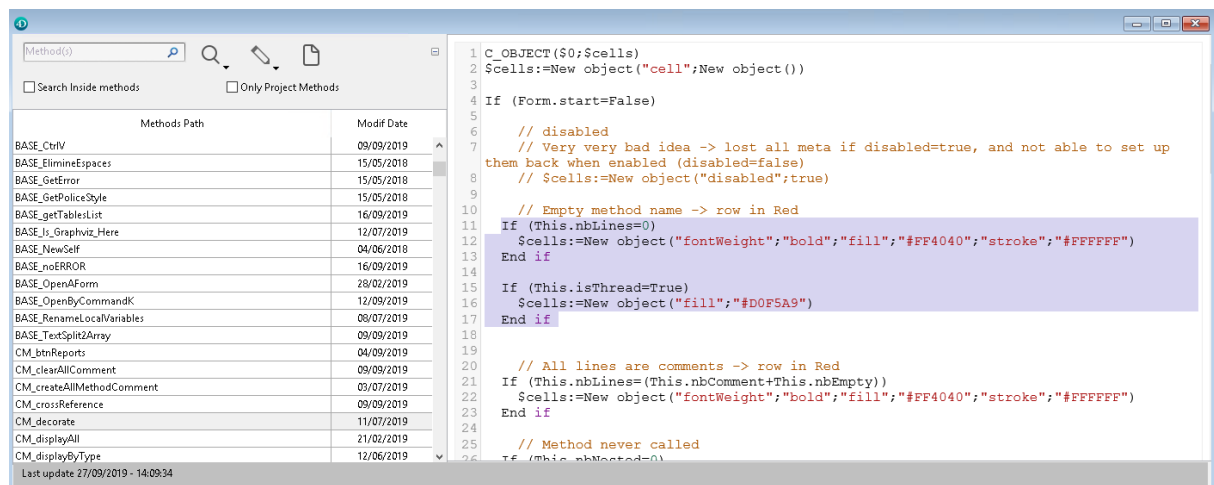
Methods Pallet



This module allows you to have the list of methods on hand to view or copy all or part of the code.

In addition to the selection tools identical to those in the Method overview module, you can search the code for project methods only, if the option is selected.

Clicking on it  displays the right panel where you will find the code of the selected method and from which you can extract a portion of the code before copying it.



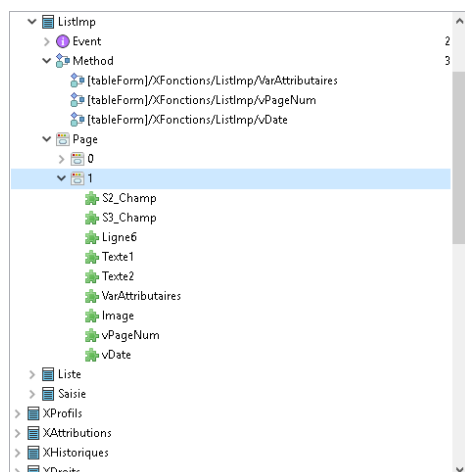
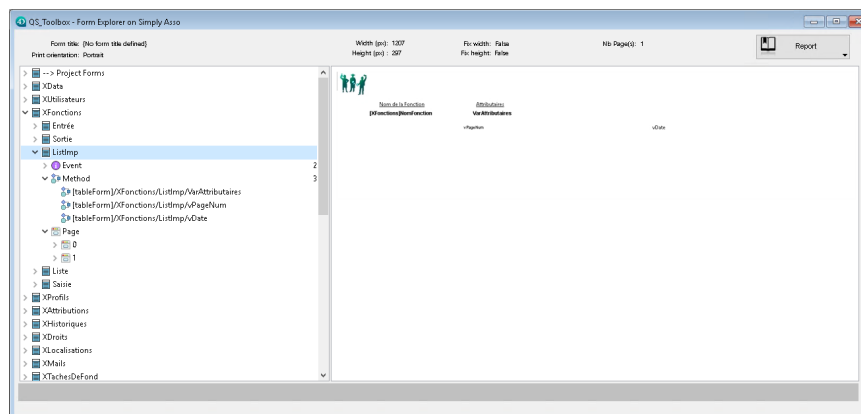
Form Explorer

This module allows you to view a set of characteristics of your forms in a different way than 4D's Form Explorer.

The top banner displays the main attributes of the form, such as the name of the form, heights and widths, orientation and number of pages.

Form title: {No form title defined}	Width (px): 802	Fix width: False	Nb Page(s): 5
Print orientation: Portrait	Height (px): 679	Fix height: False	

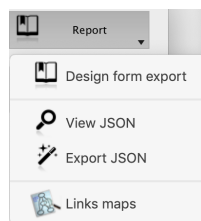
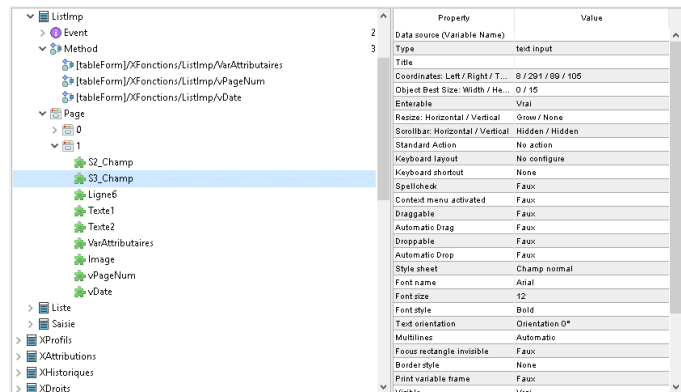
Selecting a form from the list allows you to retrieve, in real time, the events checked for that form, the list of widgets containing a method and the number of pages used in that form.



Clicking on the form name displays the screenshot of page 1 of the form.

Clicking on a page number not only displays the screen copy of the content of that page, but also lists the list of widgets that make up the page.

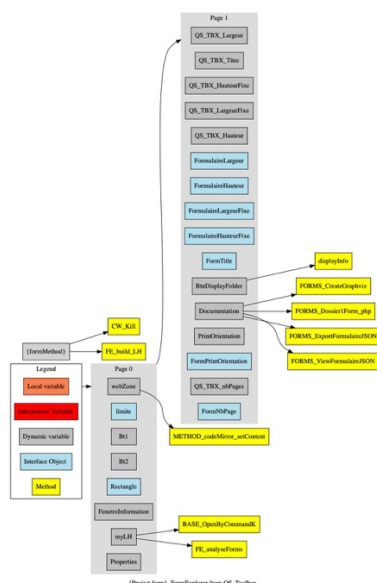
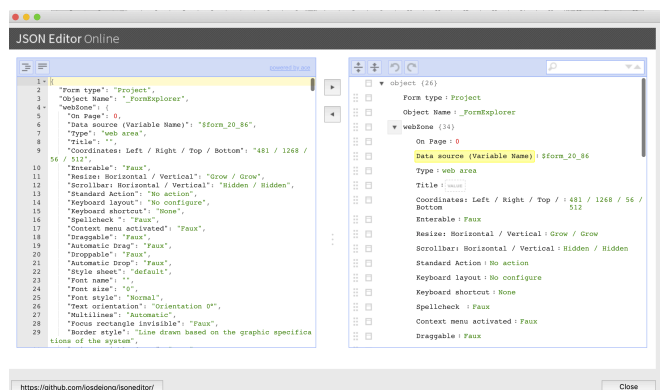
Selecting a widget allows you to display all the properties of the same widget.
The list consists of all the properties available using the 4D commands.



The 'Reports' button allows you to retrieve all the information in the form in the form:

- From an export to DOCX format
- An export in JSON format
- An export of a graphical representation of widgets, page by page
- The map of the form links

The visualization of the parameters of a form and its widgets in JSON format is available directly from QS_Toolbox.



The link map is an interesting tool.
It highlights the links between the widgets on the different pages of the form as well as the links with the project methods called from the widgets in a graphical form.

Structure Analysis

This module corresponds to the toolbox of QS_Toolbox. Indeed, it allows you to:

- View all field and table attributes
- View the definition of links
- To check, to look for what is wrong with your code.
- To execute SQL or 4D code under your sole responsibility
- Edit 4Dlink links

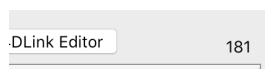
The transition from one functionality to another is managed by the following selection bar:



Structure

Display table & field definition data

When selecting a table, the list of fields that make it up is displayed. The background color is the one defined in the structure editor.



At the top right, QS_Toolbox indicates the number of records contained in this table.

When the selector is set to Tables, then the information about the Table is displayed.



Specific options

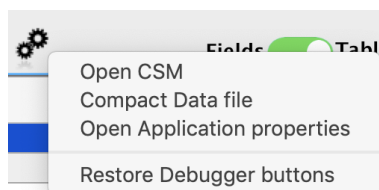
The specific options are available through the following icons:

- Copying information
- Displaying or exporting the structure schema
- Database management tools



Allows you to copy the definition of the fields of the selection table, all tables and all fields as well as to store the code for adding a record using the 4D code.

Display of the database structure and export of the structure image (PNG format).



The last icon allows you to launch the CSM dialogs, the application properties, compact the data file by command line or restore the debugger buttons when they have disappeared.

Table Relations

This feature allows you to view all the links defined in the database structure. It is not possible to define a link from this dialog. You must open the structure editor and add / modify the link from there.

Select a relation: ProfilsToFunction - FonctionToProfils

Name for N to 1: ProfilsToFunction

Name for 1 to N: FonctionToProfils

UUID: E2BF5B2DB94244ECA6613339EC7C0A93

Integrity: None

Choice field: 0

Fields:

Kind	Table name	Field name
source	XProfils	UUIDFonction
destination	XFonctions	PK_UUID

☐ Automatic load N to 1
☐ Automatic load 1 to N
☐ Auto Wildcard support
☐ Prompt if related one does not exist
☐ Auto assign related value in subform

Search & Control

This is the feature of QS_Toolbox that will allow you to perform various and varied searches on your methods

The specific options are available through the following icons:

- Copy of the result of the last search in the clipboard.
- The magnifying glass allows you to search, display tips, extract data
 - As in the previous menu, the wheels allow you to launch the CSM dialogs, the application properties, compact the data file by command line or export the image library.



The available menus are:

Check

- Check names of tables, fields and methods
- Command requiring to be "closed" after a call
- Declaration of the names of the assemblies

Search For

- Impaired orders
- Images referenced in the library
- Big tables
- A specific event
- A specific widget
- Improvements for use with 4D Server
- The use of data formatting
- the presence of dead code such as 'if (1=2)' ; 'if (false)'...

TIPS

- View all tips and tricks
- Show tips after analyzing project methods

Analyse

- Comparison of variable declarations

List

- Lists all objects in the database
- database parameters (SET DATABASE PARAMETER)
- lists the information of the WEB server
- lists the WEB server options

Get

- lists the indexes
- lists the primary keys
- lists the style sheets
- lists plug-ins
- lists the components

- copies the path of the application to memory
- copies the path of the data file to memory
- retrieves the external IP
- lists license information

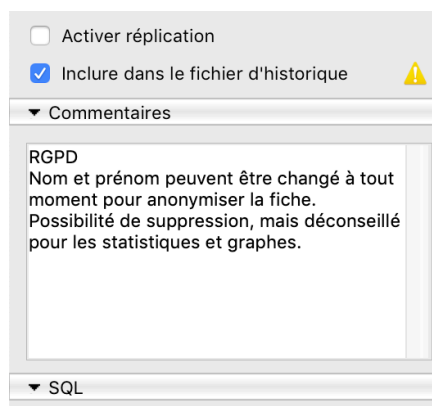
Copy system configuration


Copies all system information to memory:

- system configuration
- volume information
- network information
- printer list
- CODECS list
- component list
- plug-in list
- table fragmentation
- cache information
- ...

Get GRPD information

Retrieves information placed in the 'Comments' area and identified by GPDR



☐ Activer réplication
☒ Inclure dans le fichier d'historique 

▼ Commentaires

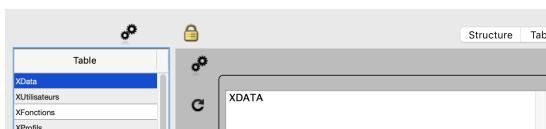
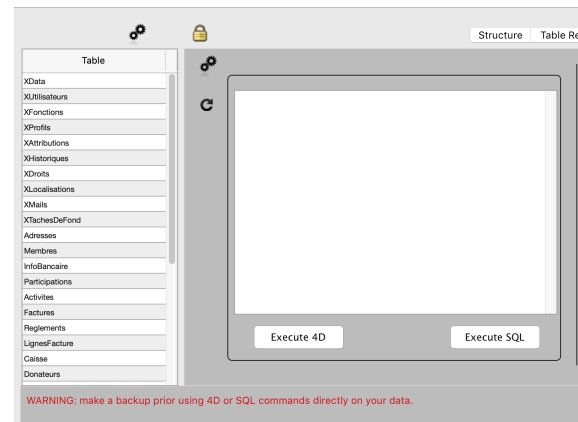
RGPD
 Nom et prénom peuvent être changé à tout moment pour anonymiser la fiche.
 Possibilité de suppression, mais déconseillé pour les statistiques et graphes.

▼ SQL

Check & Update

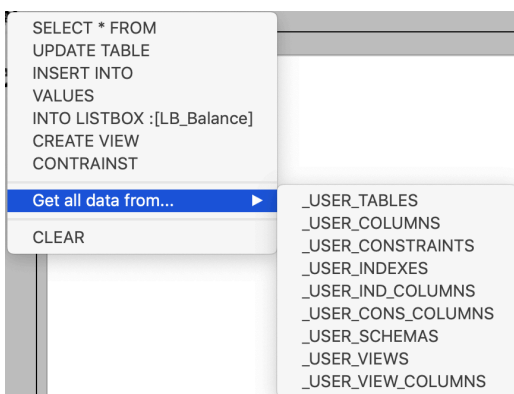
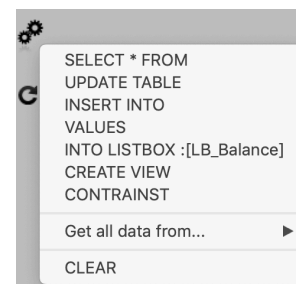
The features in this section may be dangerous for your data if the code executed makes a change to it.
An input field is available and allows you to execute either 4D code or SQL code directly from QS_Toolbox.

No verification of the entry is performed by the component.



A double click on the name of a table allows you to add the name of this table in the editor


To help you build a SQL query, the main usable keywords are available from the wheel icon.
The 'CLEAR' line allows you to delete any entry from the field.



The menu "This all data from..." allows you to obtain a list of the information of the system tables that are read-only.

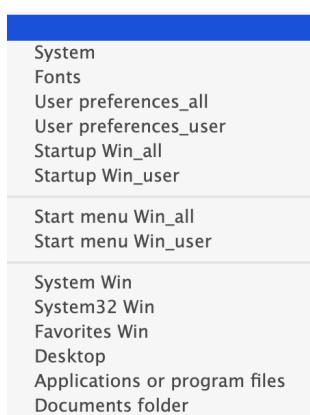
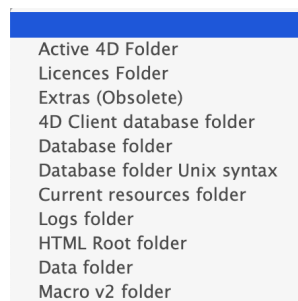
The result of a SQL query of type 'SELECT' is displayed in a listbox.

IN DE X I D	IN DE X N A M E	IN DE X T Y P E	AE Y W O R D	T A B L E N A M E	UN I Q U E N E S S	T A B L E I D	IN DE X I D
0FED0000000000000000000000000000	0FED0000000000000000000000000000	7		Adress	0	1/0	
0A000000000000000000000000000000	0A000000000000000000000000000000	7		Adress	0	1/1	
0C000000000000000000000000000000	0C000000000000000000000000000000	7		Adress	0	1/2	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	1/3	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/4	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/5	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/6	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/7	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/8	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/9	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/10	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/11	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/12	
0F000000000000000000000000000000	0F000000000000000000000000000000	7		Adress	0	2/13	

The icon  allows you to delete any entry from the field.

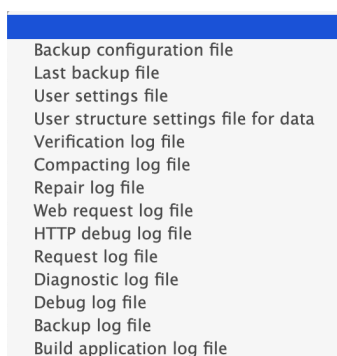
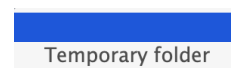
The right part of the Check & Update feature allows you to access the different folders of the system:

Access to all available folders thanks to the "4D Folder" command

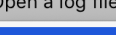
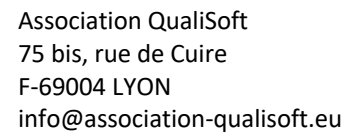


Access to all available folders through the 'System folder' command

Access to the temporary folder thanks to the command "Temporary folder".



Access to all available folders thanks to the "4D File" command



Open a log file...

- Verification log file
- Compacting log file
- Repair log file
- Build application log file

Open log with consideration of "LogStyleSheet.xsl".

[illegible]

4D Links

modelEmploi

Security Area

Q5.Toolbox

Facuration_eta

invoicesdemo

email

Global Settings

Database is Remote

Username

Q5.Toolbox

Facuration_eta

invoicesdemo

Normal

Structure

Settings for Local Database

Open in Custom

file:///Users/patrick.../Structure File...

file:///Users/patrick.../Data File Path

Skip OnStartup

Definition Imp...

Default Da... Data Opening

View of all properties used to create/update a 4DLINK File

Property List

Property Value

data_file

file:///Users/patrick.../Q5_SimplyAsso.4DD

structure_file

file:///Users/patrick.../soSimply Asso.4DD

structure_opening_mode

0

data_opening_mode

1

Create New...

Structure

Data

Settings for Remote Database

Server DB Name (without extension)

IP Address or DNS Name

Open Login Dialog

Reset Values

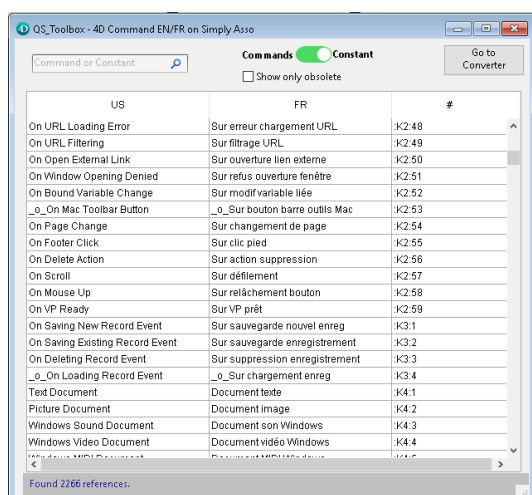
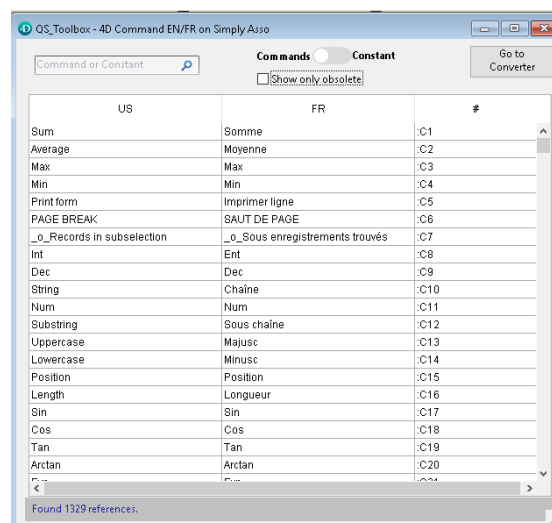
Save

This allows you, for example, to regain control over the start of an application by asking that the next time you open it, the "On Base Opening" method should not be executed.

Converter EN / FR

This module allows you to have a list of the commands and constants of the current version of 4D.

You can search by part of the name of an order or display only 'obsolete' orders with a single click.



To work with constants, simply switch the selector to the right side.

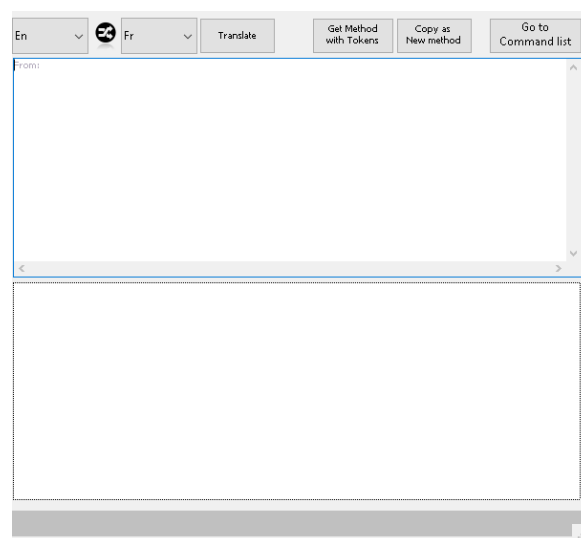
The searches are carried out in the same way as for orders.

To switch to the code translator, simply click on the 'Go to converter' button.

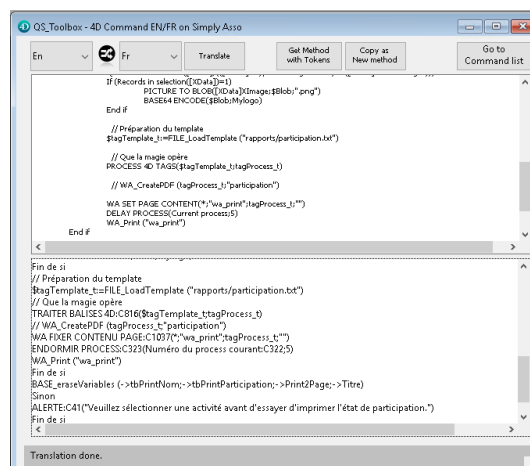
The screen that appears will allow you to enter your code (or paste it) in the From section.

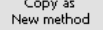
By default, the translation is: EN -> FR.

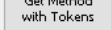
If you want to do FR -> EN, just click on the icon



Once the code to be translated is present in the upper part, simply click on the "Translate" button to obtain the translated code. Tokens are present in the translated code.



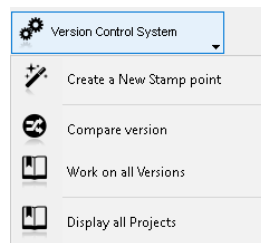
The button  allows you to save the translated code in a new project method. Despite all the controls performed for the translation, it is necessary that you proofread the code to ensure that it does not contain any errors related to the translation.

The button  allows you to read the code of a project method and to display the tokens associated with each command / constant.

Version Control System

This feature allows you to save and compare different versions of your methods, all types combined.

All methods are saved in the external database data file 'QST_External_DB'.

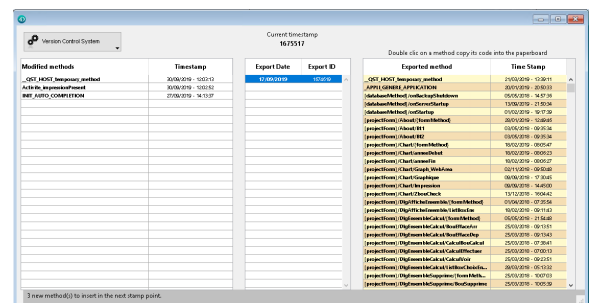


The System Control Version menu allows you to:

- To start the backup of the methods modified since the last backup
- To compare 2 versions of the same method
- Work on all backups of the current database (project)
- To work on all backups of all databases (projects) that use the VCS of QS_Toolbox

The main screen of the VCS consists of 3 distinct areas:

- List of methods to be saved
- List of exports already made
- List of methods exported when an export has been selected



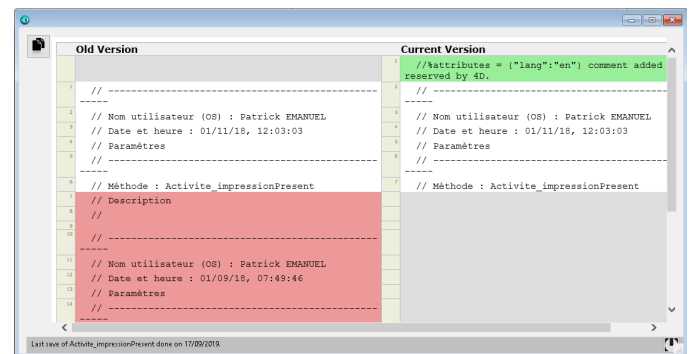
Modified methods	Timestamp	Export Date	Export ID	Exported method	Time Stamp
QST_HOCT_ImportMethod	30/09/2019 - 10:01:13	07/09/2019	1675518	Actio de assignationFormel	06/10/2018 - 13:05:10
Actio de assignationFormel	30/09/2019 - 10:01:13			Actio de assignationFormel	01/01/2019 - 09:41:12
IMP_AUTO_COMPLETION	27/09/2019 - 14:13:19			Actio de assignationFormel	01/01/2019 - 09:41:12
				Actio de assignationFormel	06/07/2018 - 16:53:30
				Adresser_Edite	30/10/2018 - 12:30:23
				Adresser_EditeForme	01/01/2019 - 14:10:14
				Adresser_EditeForme	06/05/2018 - 14:41:32
				Adresser_EditeForme	05/10/2018 - 07:15:12
				Adresser_EditeForme	23/01/2019 - 09:50:49
				Adresser_EditeForme	24/01/2019 - 10:00:02
				Adresser_EditeForme	07/09/2018 - 16:09:27
				Adresser_EditeForme	12/01/2019 - 11:43:40
				Adresser_EditeForme	11/01/2019 - 07:31:17
				Adresser_EditeForme	21/12/2018 - 11:45:55
				Adresser_EditeForme	20/12/2018 - 12:23:35
				Adresser_EditeForme	21/11/2018 - 11:45:34
				Adresser_EditeForme	04/08/2018 - 16:30:20
				Adresser_EditeForme	21/09/2018 - 17:36:55
				Adresser_EditeForme	26/01/2019 - 09:28:07
				Adresser_EditeForme	24/07/2018 - 07:05:50
				Adresser_EditeForme	06/07/2018 - 11:44:48
				Adresser_EditeForme	06/07/2018 - 12:34:47
				Adresser_EditeForme	05/11/2018 - 12:05:46
				Adresser_EditeForme	02/11/2018 - 12:01:11
				Adresser_EditeForme	21/12/2018 - 11:46:30
				Adresser_EditeForme	01/10/2018 - 13:01:59


When a click is made on a method that needs to be saved, QS_Toolbox searches all backups for the previous version of the same method and automatically selects it when it is found.

A new method backup will clear the first list and add a new line to the second.

Modified methods	Timestamp	Export Date	Export ID
		30/09/2019	1675518
		17/09/2019	1674619

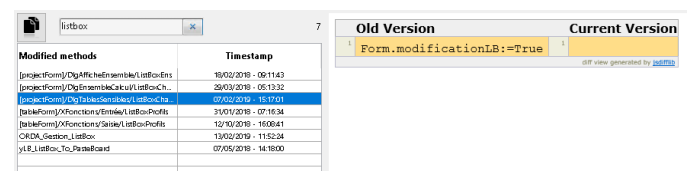
The comparison between the 2 methods uses a Java applet that displays the deletions in red and the additions in green



The icon  allows you to copy the content of OldVersion into memory in order to retrieve it in a text editor or to copy directly into the method editor.

Access to all backups displays all saved methods.

A search box allows you to narrow down the list in order to compare the code of the selected method with that of the current version. The number at the search level corresponds to the number of methods listed. Selecting a method automatically starts the comparison. In the following example, the code has simply been deleted.



Projects	Export Date	Export ID
Q5_Toolbox4DB	22/09/2019	2200339
Simply Asso4DB	18/09/2019	2186800
	18/09/2019	0
	18/09/2019	0
	18/09/2019	0
	18/09/2019	0
	17/09/2019	2184267
	28/08/2019	2074217
	04/03/2019	0
	26/02/2019	0

Exported method	Time Stamp
_saletest	21/09/2019 - 21:39:01
[projectForm]/_VCS/[formMethod]	19/09/2019 - 16:59:34
[projectForm]/_VCS/AllVersion_lb	20/09/2019 - 19:00:39
[projectForm]/_VCS/BtnGetOldVersion	18/09/2019 - 17:37:19
[projectForm]/_VCS/BtnGoPage1	19/09/2019 - 21:04:51
[projectForm]/_VCS/Exported_lb	20/09/2019 - 19:09:52
[projectForm]/_VCS/Exported_lb1	19/09/2019 - 21:42:24
[projectForm]/_VCS/Exports_lb1	20/09/2019 - 19:06:12
[projectForm]/_VCS/Modified_lb	19/09/2019 - 13:16:24
[projectForm]/_VCS/Projects_lb	20/09/2019 - 19:02:30
[projectForm]/_VCS/SearchPicker	19/09/2019 - 16:57:48
[projectForm]/_VCS/SearchPicker1	20/09/2019 - 19:06:30
CM_getCodeToDisplay	20/09/2019 - 18:49:42
CM_removeEtetLine	20/09/2019 - 18:47:07

Access to all projects displays the list of projects saved in the QS_Toolbox VCS and allows you to navigate through all the backups performed. This has a twofold benefit:

- Check that the code is the same on projects using "identical" methods
- Copy code from another project without having to open it

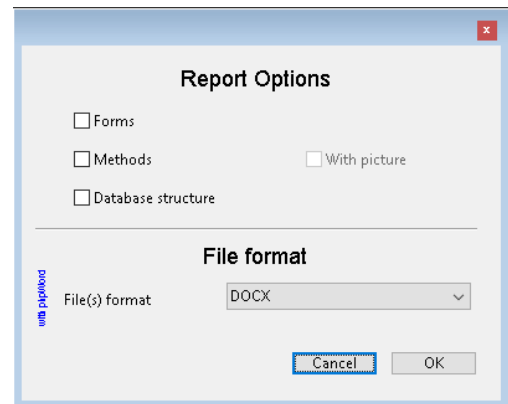
Reports

The reports created allow the export to text format of all the information concerning:

- The forms
- The methods
- The structure of the database

The export formats are:

- DOCX
- HTML
- RTF
- ODT



Nom	Modifié le	Type	Taille
Project Form - _Menu.docx	11/07/2019 13:29	Document Micros...	61 Ko
Path project method - 1.docx	02/07/2019 09:50	Document Micros...	183 Ko
Project Form - _Windows.docx	24/06/2019 15:22	Document Micros...	106 Ko
Project Form - _VCS.docx	24/06/2019 15:22	Document Micros...	256 Ko
Project Form - _StructureAnalysis.docx	24/06/2019 15:22	Document Micros...	634 Ko
Project Form - _Setup.docx	24/06/2019 15:21	Document Micros...	106 Ko
Project Form - _ObsoleteCommands.docx	24/06/2019 15:21	Document Micros...	122 Ko
Project Form - _Methods_Pallet.docx	24/06/2019 15:21	Document Micros...	111 Ko
Project Form - _MethodManager.docx	24/06/2019 15:21	Document Micros...	337 Ko
Project Form - _MethodAnalyzer.docx	24/06/2019 15:20	Document Micros...	85 Ko
Project Form - _LinkCreateStructureFile.d	24/06/2019 15:20	Document Micros...	53 Ko
Project Form - _LinkCreateDataFile.docx	24/06/2019 15:20	Document Micros...	51 Ko
Project Form - _joneditor_dialog.docx	24/06/2019 15:20	Document Micros...	62 Ko
Project Form - _FormExplorer.docx	24/06/2019 15:20	Document Micros...	150 Ko
Project Form - _FormatSortie.docx	24/06/2019 15:20	Document Micros...	75 Ko
Project Form - _DataExplorer.docx	24/06/2019 15:20	Document Micros...	58 Ko
Project Form - _dataByOrda_detail.docx	24/06/2019 15:20	Document Micros...	61 Ko
Project Form - _dataByOrda.docx	24/06/2019 15:19	Document Micros...	69 Ko
Project Form - _Copyrights.docx	24/06/2019 15:19	Document Micros...	241 Ko
Project Form - _Converter.docx	24/06/2019 15:19	Document Micros...	127 Ko
Project Form - _Commands.docx	24/06/2019 15:19	Document Micros...	128 Ko

The generated exports can provide a basis for documenting your application's sources. The content includes all the available information on each of the exported elements.

The export of forms systematically includes:

- A screen copy of each page of the form
- A table defining the properties for each of the widgets used

The method export includes the methods of all objects in the database.

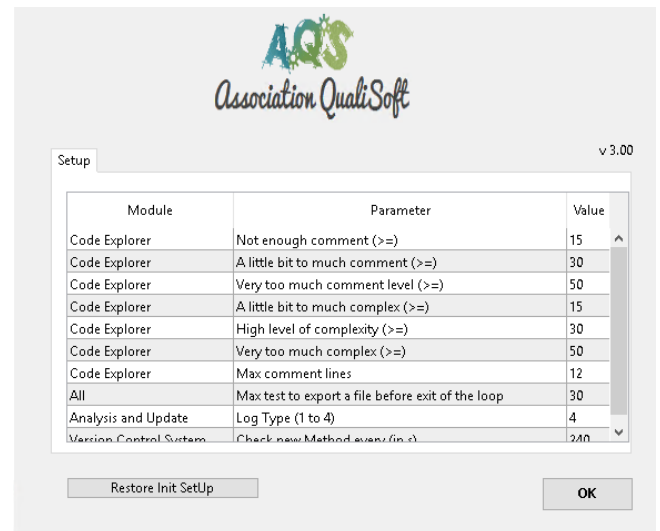
The export of the structure includes:

- The image of the database structure.
- The definition of each of the tables
- The definition of each of the fields

Setup

The different options of the application allow:

- To set the limits used for the definition of the colours used in the Method Overview section
- Set the number of times a file must be tested when it is written to disk. On VERY fast systems, this number must be increased
- To define (in second) the time between 2 modified method checks (only for the VCS).



New DB Analysis

Restarts an analysis of the database.

During the analysis, the application may be slightly slowed down. This is related to the use of resources.

Restart 4D

Restarts the application to consider the latest changes applied to the structure such as defining links for use by ORDA.