








Q-CHECKER FOR V5
RELEASE 5.4

INSTALLATION GUIDE



Conventions in this Manual

Markings

-  CAUTION marks an advice that performing an action may lead to unwanted consequences such as loss of data.
-  NOTE marks important information, such as dependencies or requirements of a function.
-  INFO marks supplementary information, e.g. with respect to a wider context.
-  TIP marks information for the practical use of the software.
-  This symbol marks step-by-step instructions.

Syntax notation for command line input and configuration files

| | |
|---------------|-------------------------------------------------------------------------------------------------------------------|
| Bold | Boldface marks text to be typed as shown. |
| <i>Italic</i> | Italics mark text to be replaced by the required parameter value. |
| [] | Square brackets mark an optional parameter. Include this parameter, if required. |
| { } | Braces mark a list of parameters, which are separated by vertical bars . Include one of the given parameters. |
| ... | The parameter before the ellipsis may be included more than once. |

Copyright Dassault Systèmes 2000–2016

Q-Checker is a trademark or registered trademark of Dassault Systèmes or its subsidiaries, in the US and/or other countries.

Contents

| | | |
|-----|----------------------------------------------------------------|-----------|
| | Conventions in this Manual | 2 |
| 1 | Rechtliche Hinweise / Legal Notices | 5 |
| 2 | Conventions in this manual | 7 |
| 3 | System Requirements | 8 |
| 4 | Installing on Windows using the installer | 10 |
| 5 | Silent installation | 15 |
| 6 | Uninstalling on Windows using the installer | 17 |
| 7 | Installing manually on Windows | 18 |
| 8 | Installing on UNIX | 19 |
| 8.1 | Overview | 19 |
| 8.2 | Unpacking the installation package | 19 |
| 8.3 | Creating a new CATIA environment | 20 |
| 8.4 | Extending an existing CATIA environment | 21 |
| 9 | Environment Variables | 23 |
| 9.1 | Q-Checker application path variables | 23 |
| 9.2 | Q-Checker application settings variables | 24 |
| 9.3 | Database connection variables | 27 |
| 9.4 | PDM Save variables | 28 |
| 10 | Directory structure of a Q-Checker installation | 29 |
| 11 | Q-Checker startup script | 31 |
| 12 | Enrolling licenses | 32 |
| 13 | Adding Q-Checker environments to the installation | 33 |
| | Installing an environment | 33 |

1 Rechtliche Hinweise / Legal Notices

Q-Checker is © 2000–2016 Dassault Systèmes.

This page specifies the patents, trademarks, copyrights, and restricted rights for the Q-Checker for V5, Q-PLM for V5 and Q-Monitor for V5 Release 5.4:

Trademarks

Q-Checker, Q-PLM and Q-Monitor, 3DEXPERIENCE, the Compass logo and the 3DS logo, CATIA, SOLIDWORKS, ENOVIA, DELMIA, SIMULIA, GEOVIA, EXALEAD, 3D VIA, BIOVIA, NETVIBES, 3DSWYM and 3DEXCITE are commercial trademarks or registered trademarks of Dassault Systèmes, a French "société européenne" (Versailles Commercial Register # B 322 306 440), or its subsidiaries in the U. S. and/or other countries. All other trademarks are owned by their respective owners. Use of any Dassault Systèmes or its subsidiaries trademarks is subject to their express written approval.

DS Offerings and services names may be trademarks or service marks of Dassault Systèmes or its subsidiaries.

Third-Party Copyrights Notices

Certain portions of the Q-Checker for V5, Q-PLM for V5 and Q-Monitor for V5 Release 5.4 contain elements subject to copyright owned by the following entities:

@ Digia

@ IBM

The Q-Checker for V5, Q-PLM for V5 and Q-Monitor for V5 Release 5.4 include open source software components. Source code for these components is available upon request. The original licensors of said open source software components provide them on an "as is" basis and without any liability whatsoever to customer (or licensee).

| IP Asset Name | IP Asset Version | Copyright notice |
|-----------------------|------------------|------------------------------|
| Under Apache 2.0 | | |
| AXIS2C | 1.6 | © Apache Software Foundation |
| AXIS2J | 1.4 or higher | © Apache Software Foundation |
| Under OpenSSL license | | |
| OpenSSL | | © OpenSSL Development team |

Restricted Rights

This clause applies to all acquisitions of Dassault Systèmes Offerings by or for the United States federal government, or by any prime contractor or subcontractor (at any tier) under any contract, grant, cooperative agreement or other activity with the federal

government. The software, documentation and any other technical data provided hereunder is commercial in nature and developed solely at private expense. The Software is delivered as "Commercial Computer Software" as defined in DFARS 252.227-7014 (June 1995) or as a "Commercial Item" as defined in FAR 2.101(a) and as such is provided with only such rights as are provided in Dassault Systèmes standard commercial end user license agreement. Technical data is provided with limited rights only as provided in DFAR 252.227-7015 (Nov. 1995) or FAR 52.227-14 (June 1987), whichever is applicable. The terms and conditions of the Dassault Systèmes standard commercial end user license agreement shall pertain to the United States government's use and disclosure of this software, and shall supersede any conflicting contractual terms and conditions. If the DS standard commercial license fails to meet the United States government's needs or is inconsistent in any respect with United States Federal law, the United States government agrees to return this software, unused, to DS. The following additional statement applies only to acquisitions governed by DFARS Subpart 227.4 (October 1988): "Restricted Rights - use, duplication and disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (l) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252-227-7013 (Oct. 1988)

Q-Checker is © 2000–2016 Dassault Systèmes.

2 Conventions in this manual



CAUTION: The signal word “CAUTION” indicates that a certain action may cause unintended results and the loss of data.

Syntax notation for command line input and text-based input

Fett:

Boldface marks text to be typed as shown.

Kursiv:

Italics mark text to be replaced by the required parameter value.

[]

Square brackets mark an optional parameter. Include this parameter, if required.

{ | }

Braces mark a list of parameters, which are separated by vertical bars. Include one of the given parameters.

...

The parameter before the ellipsis may be included more than once.

3 System Requirements



NOTE: Q-Checker is supported only on systems certified for the corresponding CATIA release. For detailed software requirements, refer to the CATIA Program Directory that is included with CATIA software.

Supported operating systems

- Microsoft Windows XP Professional x64 Edition SP2 (CATIA V5-6R2012)
- Microsoft Windows Vista Enterprise / Business Edition 64-bit SP1 or SP2 (CATIA V5-6R2012)
- Microsoft Windows 7 Enterprise / Pro Edition 64-bit RTM or SP1
- Microsoft Windows 8 Desktop Enterprise / Pro Edition 64-bit
- IBM AIX 7.1 64-bit (V5-6R2014)

The Q-Checker environment editor is only available for Microsoft Windows.

CATIA versions and releases

Q-Checker is available for CATIA V5-6R2012 oder later.

Supported CATIA platforms and required packages

All CATIA platforms are supported: P1, P2, P3, PLM Express.

Depending on the CATIA platform, the following CATIA configuration packages must be installed and the following licenses must be available:

| Platforms | Required configurations | Required licenses |
|-----------|--------------------------------------------------------|-------------------|
| CATIA P1 | All configuration packages | At least MD1 |
| CATIA P2 | At least E12+MD2+PM2+SA2 configuration and PX1 product | At least MD2 |
| CATIA P3 | On request | On request |

To use certain composites criteria, a CATIA CD3 license is required.

To use criterion *MML (Multi-Model Link Not Published)*, a CATIA PX1 license is required.

License server

Q-Checker requires Dassault Systèmes License Server (DSLS) R2016x or later.

To download DSLS, visit this website:

↗ <http://www.3ds.com/support/download/dassault-systemes-license-server/>

Other required software

- HTML browser for viewing check reports in HTML format
- PDF viewer for viewing and printing the documentation in PDF format

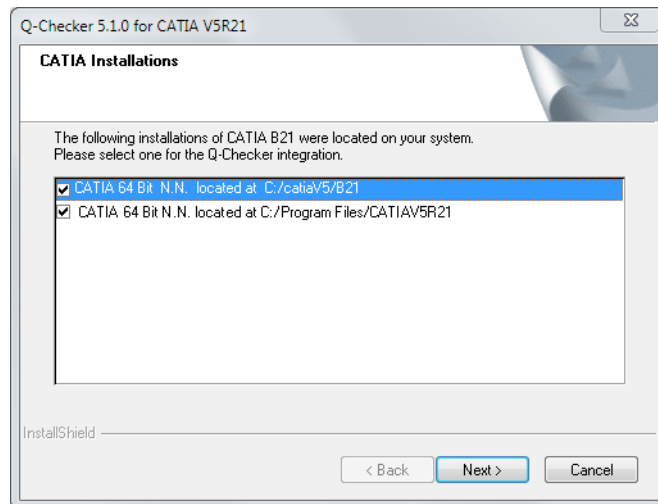
4 Installing on Windows using the installer



NOTE: In some cases, the installer cannot be used, for example when installing Q-Checker on a server. Instead, you can manually install Q-Checker from an archive.

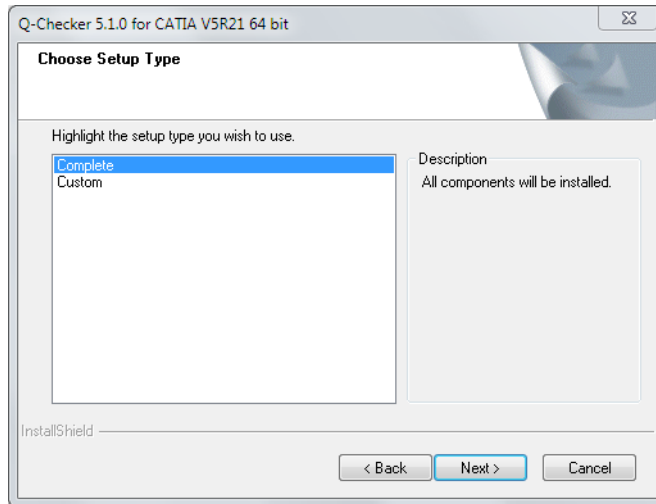
➤ Chapter 7 *Installing manually on Windows*, page 18

- 1 Q-Checker installation packages are specific to CATIA releases. Choose the correct Q-Checker installation package for the installed version of CATIA.
- 2 Run the installer.
 - To proceed, make the required settings, then click “Next”.
 - To change settings in a previous step, click “Back” repeatedly until you reach the respective dialog.
 - To abort the installation, click “Cancel”.
- 3 The installer looks up installations of the respective CATIA release on your computer. Select the installation of CATIA for which Q-Checker will be installed.



If no matching CATIA installation is found, the installer displays an error message and aborts the installation.

- 4 Choose the type of installation:
 - “*Complete*”: Install all components of Q-Checker
 - “*Custom*”: Install only selected components of Q-Checker (see the next step).

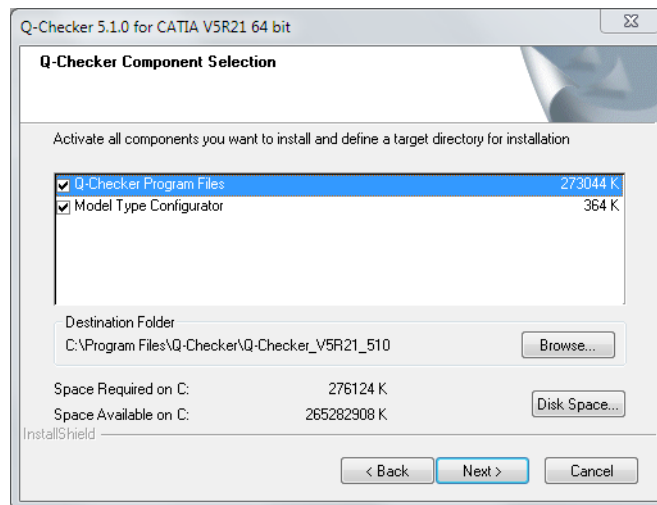


5 If you have selected a “*Custom*” installation:

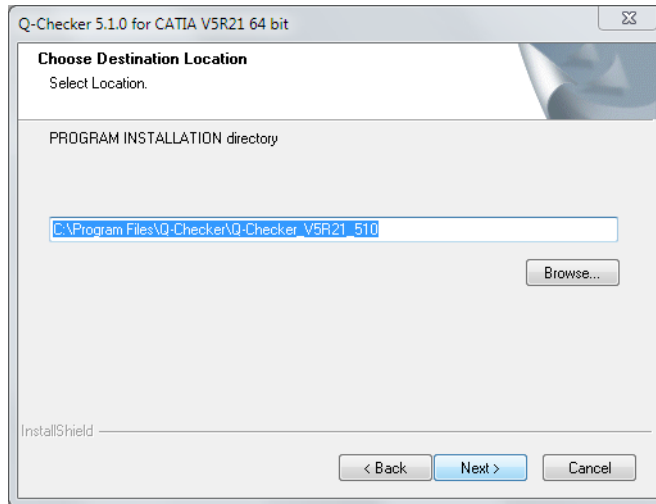
Select which components to install:

- “*Q-Checker program files*”: Main Q-Checker application
- “*Model type configurator*”: Utility for creating a root-feature attribute for model type recognition by Q-Checker.

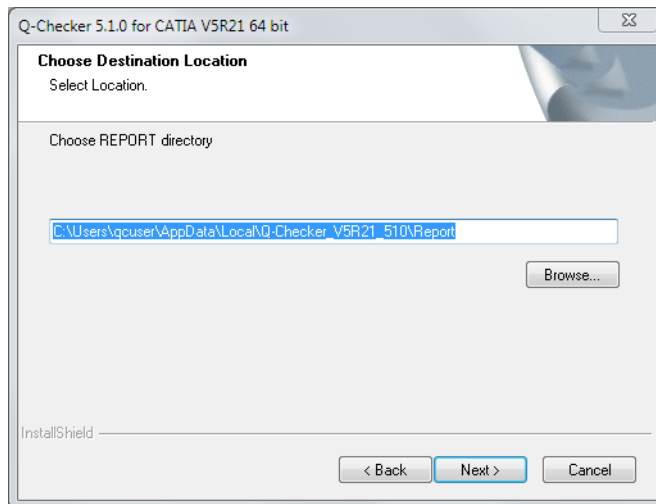
In the “Destination folder” field, select the path where Q-Checker will be installed.



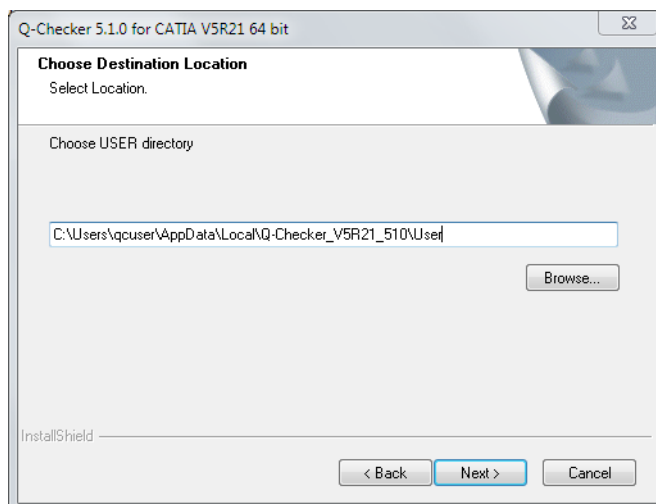
6 In case of a complete installation, select the path where Q-Checker will be installed.



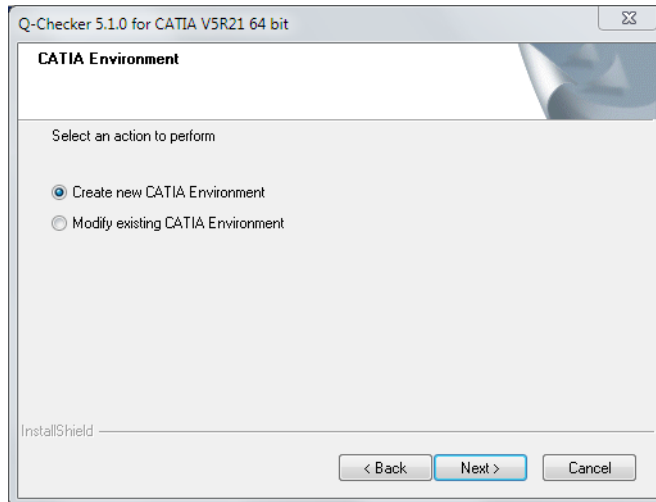
- 7 Select the path where Q-Checker will store the report files.



- 8 Select the path where Q-Checker will store user-defined environments and user settings files.



- 9 To make Q-Checker available in CATIA, a CATIA environment must exist where the application paths and other variables are defined. Select an option:

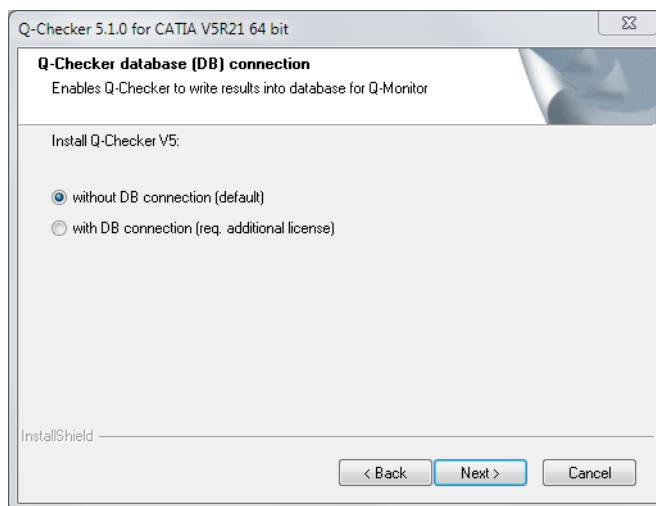


- *“Create a new environment”* (recommended): You are prompted to confirm the path and name of the new environment. The name of the environment must not contain blanks.
- *“Modify an existing environment”*. You are prompted to select a CATIA environment from the environment folder.

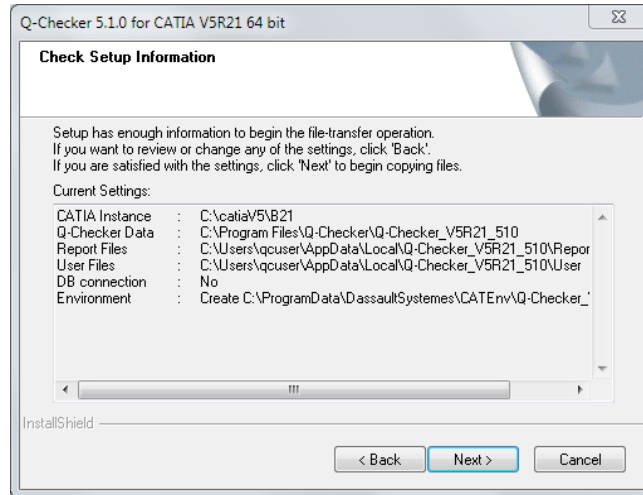
An existing environment must not contain Q-Checker variables; otherwise the installer cannot modify this environment.

10 Choose whether to use Q-Checker with or without database connection.

Database connection can send results to a database for analysis with QMonitor. A separate license is required to use database connection.



11 The installer is now ready to install Q-Checker with the settings that you have specified.



To install Q-Checker, click “Next”. To review a setting, click “Back”.

▶▶ When the installation has finished, shortcuts to CATIA with Q-Checker and to the Q-Checker environment editor are created on the Windows Desktop. Use these shortcuts to start the respective applications.

5 Silent installation

The Windows installer allows for silent installation. This means, the installation of the software can be recorded on a reference computer and repeated on further computers without user input.

Preparations

To prepare for silent installation, run the installer once on a reference computer to record the chosen settings in a response file.

To do this, start the installer at the command line.

Command line syntax

```
<install_exe> /r [/f1"<response_file>"]
```

<install_exe> Path and file name of the installer.

<response_file> Path and file name of the response file where the installation is recorded.

If you leave out parameter /f1, the response file is written to %SYSTEMROOT%\setup.iss.

Running a silent installation

To install the software with the recorded settings, run the installer from the command line and specify the location of the response file:

Command line syntax

```
<install_exe> /s [/f1"<response_file>"] [/f2"<log_file>"]
```

<install_exe> Path and file name of the installer.

<response_file> Path and file name of the response file to which the installation has been recorded.

If you leave out parameter /f1, the response file is written to %SYSTEMROOT%\setup.iss.

<log_file> Path and file name of the log file in which the results of the silent installation have been recorded.

If you leave out parameter /f2, the log file will be written to a file with the same path and name as the response file, but with the extension log.

The result of a silent installation can be determined from the result code in the log file. Result code 0 means that the installation has finished successfully:

```
...  
[ResponseResult]  
ResultCode=0  
...
```

➤ A complete list of result codes can be found in the Installshield documentation. Search the internet for “Installshield Checking for Errors Using the Setup.log File ”

Example

- Install and write settings to a response file:

```
product_setup.exe /r /f1"c:\temp\setup.iss"
```

- Installation with identical settings (command in a single line):

```
product_setup.exe /s /f1"c:\temp\setup.iss"  
/f2"c:\temp\setup.log"
```

6 Uninstalling on Windows using the installer

- 1 In Windows, go to “Start” > “Settings” > “Control Panel” > “Add or Remove Programs”.
- 2 In the “Software” dialog, select the version of Q-Checker to uninstall.
- 3 Click “Remove” and follow the instructions on screen.
- 4 After the installer has finished, you can delete manually directories and files created by Q-Checker.

7 Installing manually on Windows

Manual installation on Windows is intended for cases in which the installer cannot be used, for example on a server where no local CATIA environment files are available.

To integrate Q-Checker with CATIA, you can create a new CATIA environment or modify an existing CATIA environment.

The following instruction assumes that you use the CATIA Environment Editor. Alternatively, you can edit an existing CATIA environment file manually.

- 1 Q-Checker installation packages are specific to CATIA releases. Choose the correct Q-Checker installation package for the installed version of CATIA.
- 2 Unpack the installation archive and move the unpacked directory to the desired location.
- 3 Start the CATIA Environment Editor. In Windows, click “Start” > “Programs” > “CATIA Pn” > “Tools” > “Environment Editor V5Rxx”.
- 4 In the CATIA Environment Editor, select the environment that you want to modify.
- 5 Click “Environment” > “New from existing...”.
- 6 In the “Name” field, enter a name for the new environment.
- 7 Select the “Add a new path” option.
- 8 In the “Install Path” field, append a semicolon (;) and the full path to the Q-Checker load directory.

Example

```
C:\catiaV5\r21;C:\Q-Checker_V5R21_5xx\load
```

- 9 Click “OK”.
- 10 Add the required environment variables to the CATIA environment.

To add a new variable, right-click the list of variables, then click “New Variable” on the context menu.

↗ Environment variables: Chapter 9 *Environment Variables*, page 23
- 11 When you have finished, click “Environment” > “Quit”.

8 Installing on UNIX

8.1 Overview

Installation on UNIX comprises the following steps:

1. Unpack the installation package.
2. Integrate Q-Checker into a CATIA environment. You can either create a new CATIA environment or modify an existing CATIA environment.

8.2 Unpacking the installation package

- 1 Q-Checker installation packages are specific to CATIA releases. Choose the correct Q-Checker installation package for the installed version of CATIA.
- 2 If the installation package file has the extension `.tgz`, change the extension to `.tar.gz`:

```
mv *.tgz *.tar.gz
```

If the installation package file has the extension `.taz`, change the extension to `.tar.Z`.

```
mv *.taz *.tar.Z
```

- 3 Move the installation package to the desired location:

Example

```
cp QCheckerV5_5xx.tar.gz /catdat/qchecker
```

- 4 Change to the installation directory:

Example

```
cd /catdat/qchecker
```

- 5 Unpack the archive.

- Unpack a `.tar.gz` archive:

```
gzip -d -c qcheckerV5Rxx_yyy.tar.gz | tar -xvf -
```

- Unpack a `.tar.Z` archive:

```
uncompress -d -c qcheckerV5Rxx_yyy.tar.Z | tar -xvf -
```

When unpacked, all files are located in a directory named `qcheckerV5Rxx_yyy`.



NOTE: Unpack UNIX installation packages on UNIX. When unpacking these files elsewhere, installation files might become unreadable.

8.3 Creating a new CATIA environment

- 1 The `QCheckerSetEnv.sh` shell script helps you to create a new CATIA environment. Before you run this script, adapt the paths and variables defined in the script as required.

To add environment variables to the new CATIA environment, use the following script command:

```
chcatenv -e $QCHECKER_ENV -d $QCHECKER_ENV_PATH -var varname=value -new
```

Where `varname` is the name and `value` is the value of the environment variable.

Excerpt from the `QCheckerSetEnv.sh` script

```
#  
# Name of the new environment  
QCHECKER_ENV = "QCHECKER_V5R21_yyy"  
#  
# Store the new environment in this directory  
QCHECKER_ENV_PATH = "/CATIAV5R21/CATEnv"  
#  
# Installation directory of Q-Checker  
QCHECKER_INSTALLATION = "/catdat/qcheckerV5R21_yyy"  
#  
# CATIA installation directory  
CATIA_INSTALLATION = "/CatiaV5/R21"  
#  
# This value is important for AIX. The value can be aix_a or aix_a64  
AIX_OSDS = "aix_a"  
...
```



NOTE: When using several versions of Q-Checker in parallel, use separate user and report directories for each Q-Checker installation. This avoids Q-Checker user settings to be overwritten.

- 2 Change to the directory of the `QCheckerSetEnv.sh` script.

- 3 Run the script using the following syntax:

```
./QCheckerSetEnv.sh -new {YES | NO} -desktop {YES | NO} -a {GLOBAL | USER}
```

| Option | Description |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| -new | YES: Replace an existing environment with the same name. NO: Do not replace an existing environment with the same name. |
| -desktop | YES: Create a desktop icon to run Q-Checker with CATIA NO: Do not create a desktop icon The desktop icon will be displayed after the next login. |
| -a | GLOBAL: Create a global environment (recommended) USER: Create a user environment |

8.4 Extending an existing CATIA environment

- 1 The QCheckerEnv.sh and QCheckerEnv.csh scripts help you to create a new environment. Before you run this script, adapt the paths and variables defined in the script as required.

To add environment variables to the new CATIA environment, use the following script command:

```
varname=value  

\export varname
```

Where varname is the name and value is the value of the environment variable.



NOTE: When using several versions of Q-Checker in parallel, use separate user and report directories for each Q-Checker installation. This avoids Q-Checker user settings to be overwritten.

- 2 Change to the directory of the QCheckerEnv.sh script.
- 3 Run the script using the following syntax:

```
./QCheckerEnv.sh -new {YES | NO} -desktop {YES | NO} -a {GLOBAL | USER}
```

| Option | Description |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| -new | YES: Replace an existing environment with the same name. NO: Do not replace an existing environment with the same name. |
| -desktop | YES: Create a desktop icon to run Q-Checker with CATIA NO: Do not create a desktop icon The desktop icon will be displayed after the next login. |
| -a | GLOBAL: Create a global environment (recommended) USER: Create a user environment |

9 Environment Variables



NOTE: If an optional environment variable is not set, the value marked as “default” is applied.

9.1 Q-Checker application path variables

Environment variables for installation paths of the Q-Checker application and its components

| Variable name | Value | Description |
|-----------------|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCAQC_PATH | One directory path | Path where Q-Checker is installed. Required. |
| TCAQC_LOAD | One directory path | Path to the Q-Checker load module. Required. |
| TCAQC_ADMIN | One or more directory paths separated by semicolon. | Path where Q-Checker admin environments are stored. Required. When more than one path is specified, the names of environments and other subdirectories must be unique across all specified paths. |
| TCAQC_USER | One directory path | Path where Q-Checker user settings and user-defined check environments are stored, for example in the home directory of the respective user. Required. |
| TCAQC_RESOURCES | One or more directory paths separated by semicolon. | Path where Q-Checker resource files are stored. Required. Resource files includes translation files for the environment editor GUI and report templates. If more than one path is specified, Q-Checker will use the first path or file where the required information is available. All subsequent paths and files are ignored. |
| TCAQC_DOC | One directory path | Path to the Q-Checker manuals. Required. |
| TCAQC_HTML | One or more directory paths separated by semicolon. | Path to the Q-Checker HTML online help. |
| TCAQC_REPORT | One directory path | Path where Q-Checker check reports are stored. Required. |

9.2 Q-Checker application settings variables

Environment variables for Q-Checker application settings

| Variable name | Value | Description |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCAQC_LANG_CAT | YES, NO | <p>This setting controls the dependency of the Q-Checker language setting from the CATIA session language.</p> <p>YES: Use the CATIA session language (only English, German or French). For other CATIA session languages, use English in Q-Checker.</p> <p>NO: Users can select the Q-Checker language setting in the Q-Checker user settings.</p> |
| TCAQC_SHOW_TOOLS | One or more of the following keywords, separated by comma: PLUGINSTOQCENV, SPECTREETOXML | <p>Show buttons on the Q-Checker toolbars that are hidden by default. Specify the keywords for the buttons to be shown, separated by comma.</p> <p>PLUGINSTOQCENV: “Install current plugin set to a Q-Checker environment” button</p> <p>SPECTREETOXML: “Convert spec tree to XML” button</p> |
| TCAQC_HIDE_TOOLS | One or more of the following keywords, separated by comma: InternalReport, AnalysisViewer, SealViewer, DeleteQCElements, ModelTypeConfigurator | <p>Hide buttons on the Q-Checker toolbars that are shown by default. Specify the keywords for the buttons to be hidden, separated by comma.</p> <p>InternalReport: “Display browser for internal check reports” button</p> <p>AnalysisViewer: “Start analysis viewer” button</p> <p>SealViewer: “Start check seal viewer” button</p> <p>DeleteQCElements: “Delete all permanent marking elements” button</p> <p>ModelTypeConfigurator: “Start model-type configurator” button</p> <p>Example To hide the “Start analysis viewer” button and the “Start model-type configurator” button, define the variable like this:</p> <p>TCAQC_HIDE_TOOLS=AnalysisViewer,SealViewer</p> |

Environment variables for Q-Checker application settings (continued)

| Variable name | Value | Description |
|-------------------------------------|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCAQC_CALL_BROWSER_FOR_BATCH_OUTPUT | YES, NO | <p>YES: After a batch check, open the batch check survey in HTML format in a browser.</p> <p>NO (default): After a batch check, open the batch check survey in plain text format.</p> <p>This setting only applies to checks in batch mode with the Q-Checker user interface. For batch checks started at the command line, see OVERVIEW MANUAL, <i>Running a batch check at the command line</i>.</p> |
| TCAQC_REPORTS_IN_SPECTREE | YES, NO, SPECTREE_ONLY | <p>YES: The internal check report is available in the CATIA specification tree (for certain document types only) and in the “Internal checks reports” dialog.</p> <p>NO (default): The internal check report is not available in the CATIA specification tree, only in the “Internal check reports” dialog.</p> <p>SPECTREE_ONLY: The internal check report is only available in the CATIA specification tree. If the document type does not allow to display the check report in the specification tree, the check protocol is displayed in the “Internal check reports” dialog.</p> |
| TCAQC_LIC_EXPIRY_ALERT | Integer | Set this variable to display an alert when the Q-Checker license will expire soon. The value specifies the number of days before expiry when the alert will be displayed. To disable the alert, set the value to 0. |
| TCAQC_CATSTART | OFF | For AIX only: Run CATDUA utilities without the CATSTART command. This is required when using extended memory support for CATIA on AIX. |
| TCAQC_USE_DISPLAY_NAME | YES, NO | <p>YES: Setting for connection to VPM V4. Display the CATIA model name instead of the internal name from VPM V4.</p> <p>NO (default): Setting for connection to Smarteam. Display the CATIA model name instead of the internal name from Smarteam. For connections to VPM 4, the internal name from VPM V4 is displayed.</p> |

Environment variables for Q-Checker application settings (continued)

| Variable name | Value | Description |
|----------------|------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCTRACE | Path and filename | <p>Set this variable to activate the Q-Checker trace mode. The value specifies path and filename of the trace file.</p> <p>When TCTRACE is set, significant processing steps executed during Q-Checker checks are written to the trace file. The trace file can help to identify problems occurring while a Q-Checker check is performed.</p> |
| TCAQC_SEAL_VAL | One or more of the following keywords, separated by comma: | <p>This variable controls in which modes of operation an internal check seal is validated, and which action is performed if a model contains no check seal or an invalid check seal. Set no more than one value for each mode of operation.</p> <p>Validation when starting an interactive check</p> <ul style="list-style-type: none"> • INTERACTIVE+RECHECK: If no valid check seal exists, the model is checked again. • INTERACTIVE+INTERRUPT: If no valid check seal exists, the validation result is displayed. • None of these values: No validation in this mode of operation. <p>Validation when starting a batch check with the Q-Checker user interface:</p> <ul style="list-style-type: none"> • INTERACTIVE_BATCH+RECHECK: If no valid check seal exists, the model is checked again. • INTERACTIVE_BATCH+INTERRUPT: If no valid check seal exists, the validation result is displayed. • None of these values: No validation in this mode of operation. <p>Validation when starting a batch check at the command line:</p> <ul style="list-style-type: none"> • COMMAND_LINE_BATCH+RECHECK: If no valid check seal exists, the model is checked again. • COMMAND_LINE_BATCH+INTERRUPT: If no valid check seal exists, the validation result is displayed. • None of these values: No validation in this mode of operation. |

Environment variables for Q-Checker application settings (continued)

| Variable name | Value | Description |
|-------------------------------------------------------|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCAQC_MODELTYPE_ SAVE_ACTIVATE_ SUPPRESS_DIALOG | YES, NO | <p>Q-Checker saves the checked model depending on the “Save current model” setting in the model type manager. Users are prompted to confirm saving the model. This variable can suppress the confirmation.</p> <p>YES: Save the model without confirmation.</p> <p>NO (default): Prompt the user for confirmation.</p> <p>If the “Save current model” setting is selected in the profile options, users are also prompted to confirm saving the model. This prompt cannot be suppressed using this variable.</p> <p>↗ Setting in the model type manager for saving the checked model: ADMINISTRATION MANUAL, “Save checked model” check box</p> <p>↗ Setting in the profile options for saving the checked model: ADMINISTRATION MANUAL, “Save model after the check run” check box</p> <p>↗ User setting for suppressing the message: OVERVIEW MANUAL, “Save model’ message when starting a check” check box</p> |

9.3 Database connection variables

Environment variables for database connection

| Variable name | Value | Description |
|--------------------|---------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCAQC_DB_CONNECT | One or more of the following keywords, separated by comma: INTERACTIVE, INTERACTIVE_BATCH, COMMAND_LINE_BATCH | <p>INTERACTIVE: Connect to database for checks in interactive mode.</p> <p>INTERACTIVE_BATCH: Connect to database for batch checks started from the Q-Checker user interface.</p> <p>COMMAND_LINE_BATCH: Connect to database for batch checks started at the command line.</p> <p>Database connection requires a special license.</p> |
| TCAQC_DB_FILE_PATH | Directory path | Path to the QCHECKER.db database definition file. |

9.4 PDM Save variables

Environment variables for PDM Save

| Variable name | Value | Description |
|------------------------------------------|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| TCAQC_PDM_SAVE_ACTIVATE | YES, NO, SUPPRESS_DIALOG_AND_SAVE | <p>Only for checks in interactive mode:</p> <p>YES: Display a list of all checked models. The user can confirm to save all checked models to the PDM database, or cancel.</p> <p>If a path has been set with TCAQC_PDM_SAVE_ACTIVE_FOR_PATHkeyword, the prompt is displayed only if the model was taken from the specified path.</p> <p>If no path has been set with TCAQC_PDM_SAVE_ACTIVE_FOR_PATH, the prompt is displayed in any case.</p> <p>NO (default): Do not display the prompt and do not save the models.</p> <p>SUPPRESS_DIALOG_AND_SAVE: Save all checked models to the PDM database without confirmation.</p> <p>↗ For checks in batch mode, see OVERVIEW MANUAL, *PDM_SAVE_BATCH_ACTIVATE.</p> |
| TCAQC_PDM_SAVE_ACTIVE_FOR_PATH | Pathname | Path used by TCAQC_PDM_SAVE_ACTIVATE |
| TCAQC_PDM_SAVE_ACTIVE_AFTER_MODIFICATION | YES, NO | <p>Only for checks in interactive mode:</p> <p>YES (default): Display a list of all checked models. The user can confirm to save all checked models to the PDM database, or cancel.</p> <p>NO: Display a list of models for which a healing has been performed, which will not be saved.</p> <p>The user can confirm to save the other models, for which no healing has been performed, or cancel.</p> |
| TCAQC_PDM_TMP_SAVE_PATH | Path | Set this variable to save unsaved models as temporary files before the check. The value specifies the path for temporary files. |
| TCAQC_QPLM_LIB | See the Q-PLM documentation. | Q-PLM module which is called from Q-Checker PDM Save. |

↗ For setting up a connection between Q-Checker and Q-PLM, see the Q-PLM Installation Guide.

10 Directory structure of a Q-Checker installation

Directory structure of a Q-Checker installation

| Relative path in a default installation | Environment variable | Description |
|-----------------------------------------|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| . | TCAQC_PATH | <p>Q-Checker installation directory.</p> <p>Depending on the operating system, this directory contains the following scripts:</p> <ul style="list-style-type: none"> • qcheckerV5: Unix startup script • qcheckerV5.vbs: Windows startup script • QCheckerEnv.csh: C-Shell script to extend an existing CATIA environment with variables for Q-Checker • QCheckerEnv.sh: Shell script to extend an existing CATIA environment with variables for Q-Checker • QCheckerSetEnv.sh: Shell script to create a new CATIA new environment |
| admin | TCAQC_ADMIN | <p>Admin environments.</p> <p>When more than one path is specified in the environment variable, the names of environments and other subdirectories must be unique across all specified paths. Otherwise the environment editor cannot start.</p> |
| doc | TCAQC_DOC | <p>Q-Checker online documentation in PDF format, accessible from the Q-Checker start window and from the environment editor. The criteria help is available in various languages. Each language has its own subdirectory.</p> |
| html | TCAQC_HTML | <p>Q-Checker criteria help in HTML format, accessible from the environment editor and from the analysis window. The criteria help is available in various languages. Each language has its own subdirectory.</p> |

Directory structure of a Q-Checker installation(continued)

| Relative path in a default installation | Environment variable | Description |
|-----------------------------------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| load | TCAQC_LOAD | <p>Q-Checker load modules (program components). Depending on the operating system, the load modules are located in one of the following subdirectories.</p> <ul style="list-style-type: none"> • aix_a64: AIX 64 bit • aix_a: AIX 32 bit • hpux_b: HP-UX • solaris_a: Solaris • win_b64: Microsoft Windows 64 bit • intel_a: Microsoft Windows 32 bit <p>In the documentation, these directories are referred to by the placeholder OSDS.</p> |
| Report | TCAQC_REPORT | <p>Report files generated by Q-Checker:</p> <ul style="list-style-type: none"> • Check reports in various formats • Batch check surveys • Product component surveys • External check seals <p>As an exception, the log file for batch checks using a master process (*.out.log) is written to the working directory from which the batch check has been started.</p> |
| resources | TCAQC_RESOURCES | Resource files, including report templates in the templates subdirectory and language files for custom localization of the Q-Checker environment editor user interface in the nls subdirectory. |
| samples | No variable | Sample files, including scripts and a command file for running batch checks |
| User | TCAQC_USER | User environments and user settings files. |

11 Q-Checker startup script

In the Q-Checker startup script, you can configure the following settings:

- Paths and data submission method for Q-Checker Database Connection.
- External applications that are called from within Q-Checker, for example the browser for check reports.
- The PDQ help resource linked to the “PDQ help” button in the Q-Checker start window.

↗ ADMINISTRATION MANUAL, 12 *Q-Checker startup script*

↗ Instructions for the configuration of Q-Checker Database Connection can be found in the Q-Monitor Installation Guide.

12 Enrolling licenses

For enrolling a license key for Dassault Systèmes License Server (DSLS), please note the instructions in the CATIA Program Directory.

↗ Requirements for a license server see *License server*, page 8

13 Adding Q-Checker environments to the installation

An environment is a complete configuration of Q-Checker. You can work with any number of environments in the same Q-Checker installation.

Installing an environment

- 1 If you receive the environment as an archive file, unpack the archive.
- 2 Place the environment folder in the admin or user directory of your Q-Checker installation.

Example In the image below, the folder named `admin` is the Q-Checker admin directory. The folder named `Default` contains the environment of the same name.

You can recognize an environment folder by a file named `qchecker.qcenv` located in the folder.

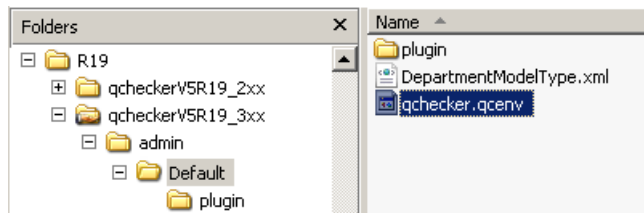


Fig. 1: An admin environment named “*Default*”, as shown in Windows Explorer

- 3 Open the Q-Checker start window and verify that the environment is available in the “Environment” list box.

New environments are available in the Q-Checker start window immediately after they have been added.



INFO: To check CAD data for conformity, suppliers can usually receive the current check environment from their contractor.

