RAPITA Systems
A DANLAW Company

Product preview

RapiCover tool qualification for ISO 26262 projects

Introduction

Tool qualification is essential in the production of software designed for safety-related embedded systems. In the tool qualification process, the tool developer provides evidence that the tool meets its functional and safety requirements. Typically, this qualification is only valid within certain conditions (*Conditions of Use*), which the tool developer must also document.

When you use Rapi**Cover** for an ISO 26262 project, we can provide tool qualification evidence. We can also provide engineering effort, tests and further documentation to help you validate the *Conditions of Use*.

Tool qualification evidence

The objective of ISO 26262 tool qualification is to provide evidence that software tools are suitable for use in developing safety-related software. We perform software tool criteria evaluation and tool qualification to produce test evidence, resulting in the work products described in Table 1.

Table 1. Tool qualification work products for ISO 26262 projects

Document	ISO 26262 reference	Description
Software Tool Criteria Evaluation Report	8-11.4.5	Assesses the detectability and consequences of tool failure. Determines tool confidence level and the tool qualification process.
Software Tool Qualification Report	8-11.4.6 to 10	Describes tool functional and robustness behaviour. Describes requirements development and testing of the tool. Presents test results, tool deviations and limitations.

RapiCover tool qualification

Internally, our requirements-based testing process follows the aerospace standard DO-330, and we derive the *Conditions of Use* from software tool-chain analysis within that process.

Rapi**Cover** may be used with off-the-shelf or custom target integrations, so the *Conditions of Use* include validation of the correctness of the target integration.

When you purchase a license for Rapi**Cover** (RPC), we can provide documents and tests that you need to validate the *Conditions of Use*. These are part of a fully qualified installation of Rapi**Cover**, which includes the following:

- RPC: the Rapi**Cover** tool.
- TIS (Target Integration Service): integration of RPC into your build and target environments (see our *Target Integration Service* product brief for more details).
- QK-RPC (Rapi**Cover** Qualification Kit): documents describing qualification for the version of Rapi**Cover** you use in your project.
- QTIS (Qualified Target Integration Service): documents describing qualification for the integration of Rapi**Cover** into your build and target environments including tests of that Rapi**Cover** integration.

Tool Qualification Kit

The tool qualification kit contains documents that describe the version of Rapi**Cover** you are using. This forms part of the evidence you need to qualify the tool, and contains the following documents:

- STCER (Software Tool Criteria Evaluation Report)
- STQR (Software Tool Qualification Report)
- Safety Manual
- Template IQR (Integration Qualification Report)

The documents in this kit must be supported with documents describing the integration of Rapi**Cover** into your development environment, and tests of the integration.

Qualified Target Integration Service

To complete your Rapi**Cover** qualification, you must provide evidence describing the integration of Rapi**Cover** into your build and target environments, and validation of the *Conditions of Use*. We consider three types of condition:

- Conditions discharged by review of the integration code. For example, the correct use of macros and appropriate memory initialization.
- Conditions discharged by tests of the integration. For example, checking that data is streaming correctly from the target.
- Conditions discharged by the process of using Rapi**Cover**. For example, validating the tool coverage settings.

When you order a qualified target integration service (QTIS), we provide the following documentation and tests:

- Full IQR (Integration Qualification Report) describing validation of the *Conditions of Use* pertaining to integration code review.
- On-site tests validating the *Conditions of Use* pertaining to tests of the integration code.
- A list of expected results.

This reduces your effort to validation of a small number of process conditions.

Qualification options and licensing

You have a number of options when you purchase Rapi**Cover**, as shown in Table 2.

Each *use* of qualification materials in a project requires a separate license. A *use* is defined as a tool installation specific to one target and test environment and typically represents a single submission or application for certification.

We offer multiple license discounts if you want to *use* our tools multiple times on the same system or project.

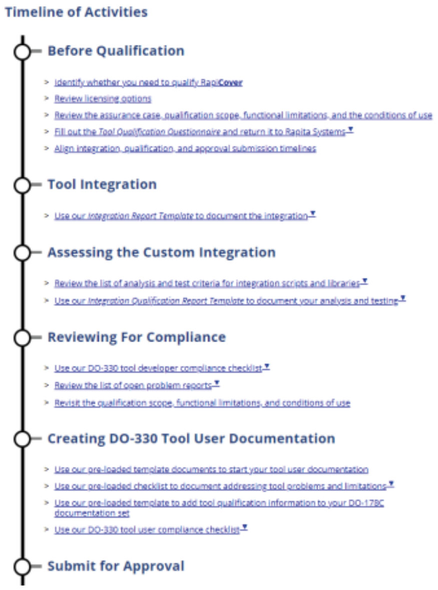
Table 2. Rapi**Cover** qualification options

		Tool		**Qualification**	
		Rapi**Cover** tool (RPC)	Target Integration Service (TIS)	Rapi**Cover** Qualification Kit (QK-RPC)	Qualified Target Integration Service (QTIS)
Option 1	Provided by:	**Rapita**	Customer	Customer	Customer
Option 2		**Rapita**	**Rapita**	Customer	Customer
Option 3		**Rapita**	Customer	**Rapita**	Customer
Option 4		**Rapita**	**Rapita**	**Rapita**	Customer
Option 5		**Rapita**	**Raipta**	**Rapita**	**Rapita**

Key features

Clear qualification guidance

Our qualification kits include clear guidance on what to do during your tool qualification, including a qualification timeline.



Timeline of Activities

Before Qualification

- Identify whether you need to qualify Rapi**Cover**
- Review licensing options
- Review the assurance case, qualification scope, functional limitations, and the conditions of use
- Fill out the *Tool Qualification Questionnaire* and return it to Rapita Systems
- Align integration, qualification, and approval submission timelines

Tool Integration

- Use our *Integration Report Template* to document the integration

Assessing the Custom Integration

- Review the list of analysis and test criteria for integration scripts and libraries
- Use our *Integration Qualification Report Template* to document your analysis and testing

Reviewing For Compliance

- Use our DO-330 tool developer compliance checklist
- Review the list of open problem reports
- Revisit the qualification scope, functional limitations, and conditions of use

Creating DO-330 Tool User Documentation

- Use our pre-loaded template documents to start your tool user documentation
- Use our pre-loaded checklist to document addressing tool problems and limitations
- Use our pre-loaded template to add tool qualification information to your DO-178C documentation set
- Use our DO-330 tool user compliance checklist

Submit for Approval

Figure 1. Qualification timeline

Streamlined qualification material

The documentation, requirements and tests included in our qualification kits are custom depending on your specific development environment, helping you minimize your review effort.

Compliance checklists

Checklists are included in our qualification kits that help you check your compliance progress.

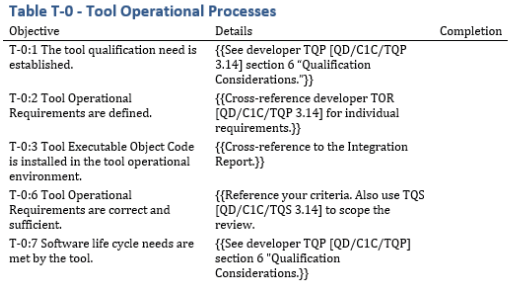


Table T-0 - Tool Operational Processes

Objective	Details	Completion
T-0:1 The tool qualification need is established.	{{See developer TQP [QD/C1C/TQP 3.14] section 6 "Qualification Considerations."}}	
T-0:2 Tool Operational Requirements are defined.	{{Cross-reference developer TOR [QD/C1C/TQP 3.14] for individual requirements.}}	
T-0:3 Tool Executable Object Code is installed in the tool operational environment.	{{Cross-reference to the Integration Report.}}	
T-0:6 Tool Operational Requirements are correct and sufficient.	{{Reference your criteria. Also use TQS [QD/C1C/TQS 3.14] to scope the review.	
T-0:7 Software life cycle needs are met by the tool.	{{See developer TQP [QD/C1C/TQP] section 6 "Qualification Considerations.}}	

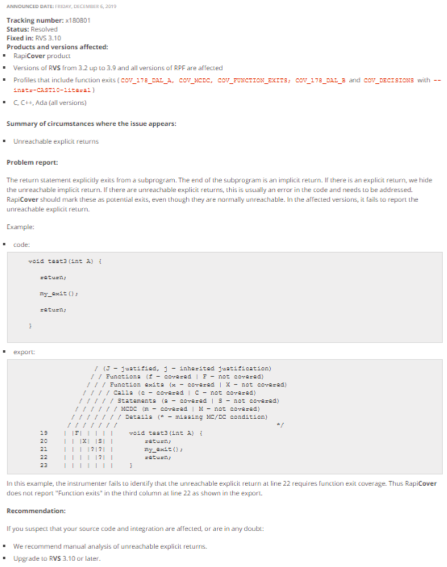
Figure 2. Compliance checklists help you check your compliance progress

Qualified instrumenters

The instrumenters used by RVS tools are qualified, so there's no need to manually qualify them.

Assurance issue notification

We notify you when we discover any assurance issues that might cause false positive results or introduce functional changes to your software. We keep you updated with the status of assurance issues regularly.



ANNOUNCED DATE: FRIDAY, DECEMBER 6, 2019

Tracking number: x180801
Status: Resolved
Fixed in: RVS 3.10
Products and versions affected:

- Rapi**Cover** product
- Versions of **RVS** from 3.2 up to 3.9 and all versions of RPF are affected
- Profiles that include function exits (`COV_178_DAL_A, COV_MCDC, COV_FUNCTION_EXITS; COV_178_DAL_B` and `COV_DECISIONS` with `--instr-CAST10-literal`)
- C, C++, Ada (all versions)

Summary of circumstances where the issue appears:

- Unreachable explicit returns

Problem report:

The return statement explicitly exits from a subprogram. The end of the subprogram is an implicit return. If there is an explicit return, we hide the unreachable implicit return. If there are unreachable explicit returns, this is usually an error in the code and needs to be addressed. Rapi**Cover** should mark these as potential exits, even though they are normally unreachable. In the affected versions, it fails to report the unreachable explicit return.

Example:

- code:

```
void test3(int A) {

   return;

   my_exit();

   return;

}
```

- export:

```
                   / (J - justified, j - inherited justification)
                  / / Functions (f - covered | F - not covered)
                 / / / Function exits (x - covered | X - not covered)
                / / / / Calls (c - covered | C - not covered)
               / / / / / Statements (s - covered | S - not covered)
              / / / / / / MCDC (m - covered | M - not covered)
             / / / / / / / Details (* - missing MC/DC condition)
            / / / / / / /                                        */
   19      | |F| | | | |    void test3(int A) {
   20      | | |X| |S| |        return;
   21      | | | |?|?| |        my_exit();
   22      | | | | |?| |        return;
   23      | | | | | | |    }
```

In this example, the instrumenter fails to identify that the unreachable explicit return at line 22 requires function exit coverage. Thus Rapi**Cover** does not report "Function exits" in the third column at line 22 as shown in the export.

Recommendation:

If you suspect that your source code and integration are affected, or are in any doubt:

- We recommend manual analysis of unreachable explicit returns.
- Upgrade to R**VS** 3.10 or later.

Figure 3. Example assurance issue

RAPITA SYSTEMS A DANLAW COMPANY

Rapita Systems Inc.
41131 Vincenti Ct.
Novi, MI 48375

Tel (USA):
+1 248-957-9801

Rapita Systems Ltd.
Atlas House, Osbaldwick Link Road
York , YO10 3JB
Registered in England & Wales: **5011090**

Tel (UK/International):
+44 (0)1904 413945

Email: **info@rapitasystems.com** | Website: **www.rapitasystems.com**

Document ID: MC-PP-401 RapiCover Tool Qualification ISO 26262 v1