

Rational. Rhapsody



IBM® Rational® Rhapsody®



IBM Rational Rhapsody TestConductor Add On Qualification Kit for DO-178B/C Overview

Version 1.11



License Agreement

No part of this publication may be reproduced, transmitted, stored in a retrieval system, nor translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of the copyright owner, BTC Embedded Systems AG.

The information in this publication is subject to change without notice, and BTC Embedded Systems AG assumes no responsibility for any errors which may appear herein. No warranties, either expressed or implied, are made regarding Rhapsody software including documentation and its fitness for any particular purpose.

Trademarks

IBM® Rational® Rhapsody®, IBM® Rational® Rhapsody® Automatic Test Generation Add On, and IBM® Rational® Rhapsody® TestConductor Add On are registered trademarks of IBM Corporation.

All other product or company names mentioned herein may be trademarks or registered trademarks of their respective owners.

© Copyright 2000-2017 BTC Embedded Systems AG. All rights reserved.

Table of Contents

1.How to Read this Document	4
2.Qualification of Tools in the Context of DO-178C and DO-330	5
3.Tool Qualification of IBM Rational Rhapsody TestConductor Add On according to the DO-178C and DO-330.....	6
4.Overview about documents relevant for the TestConductor Qualification Kit for DO-178C.....	11
5.List of References	15

1. How to Read this Document

This document shall provide guidance to qualify IBM Rational Rhapsody TestConductor Add On for DO178-B [4.] and DO-178C [5.] projects, respectively. Core of the approach is to follow the recommendations of the DO-330 “Software Tool Qualification Considerations” [7.]. In order to fulfill the obligations as described in DO-330 a TestConductor validation suite has been developed and been made available to customers. This enables customers to qualify TestConductor for their projects.

It is assumed that the reader of this document has access to documents DO-178C and DO-330. Further, it is assumed that the reader of this document understands the approach to qualify COTS tools for DO-178-C/B projects, as sketched in DO-330 chapter 11.3. In the sequel of this document we discuss the obligations regarding tool qualification as described in DO-330 chapter 11.3, and how these are supported with the “IBM Rational Rhapsody TestConductor Add On Qualification Kit for DO-178B/C”.

Additional Information:

There IBM Rational provides a “IBM Rational Rhapsody Kit for DO-178B/C”. While this document focuses on the *qualification* of TestConductor, the mentioned IBM Rational Rhapsody Kit for DO-178B/C provides additional guidance documentation to *safely work* with tools like IBM Rational Rhapsody and IBM Rational Rhapsody TestConductor Add On in DO-178B/C type projects.

2. Qualification of Tools in the Context of DO-178C and DO-330

The DO-178C fully named “Software Considerations in Airborne Systems and Equipment Certification” published by RTCA, Inc. [1.], in a joint effort with EUROCAE [2.], replaces the DO-178B as the primary document by which the certification authorities such as FAA, EASA and Transport Canada will approve all commercial software-based aerospace systems. The DO-178C comes with several companion documents addressing specific application areas or technologies, the so-called supplements.

If software is to be developed and certified according to the DO-178C then this applies – in a variable extend – for all tools used in the software life cycle. In the example of IBM Rational Rhapsody (“Rhapsody”) and IBM Rational Rhapsody TestConductor Add On (“TestConductor”) as a modeling tool (Rhapsody) and a testing tool (TestConductor) these tools might have to be qualified according to the guidelines stated in the DO-178C and supplement DO-330. These guidelines also fulfill the tool qualification requirements according to DO-178B.

As for all other tools, Rhapsody and TestConductor may not be qualified for general use against the DO-178C, but must be qualified for use on each project.

This document is intended to give the user the means at hand to qualify TestConductor for a certain project.

Section 3. of this document gives an overview that describes how TestConductor and the Qualification Kit comply with the objectives for COTS tools (Commercial off the shelf) stated by the DO-178C and DO-330. In section 4. the quoted documentation and data is shortly described with a location where to find it.

3. Tool Qualification of IBM Rational Rhapsody TestConductor Add On according to the DO-178C and DO-330

3.1 Scope of using TestConductor in the Software Life Cycle

IBM Rational Rhapsody TestConductor is an add-on for IBM Rational Rhapsody. Rhapsody is a model based development environment for system and software engineering. TestConductor is a testing tool supporting test architecture and test creation, automatic test execution and report generation.

The DO-178C defines a list of software testing objectives and demands several software testing activities to fulfill those objectives. The objectives and recommended activities are described in section DO-178C/6.4 and are summarized in the tables DO-178C/A-6 and DO-178C/A-7.

Of the listed objectives TestConductor can be used to automate the following testing activities:

- Executable Object Code complies with high-level requirements (Table A-6.1)
- Executable Object Code is robust with high-level requirements (Table A-6.2)
- Executable Object Code complies with low-level requirements (Table A-6.3)
- Executable Object Code is robust with low-level requirements (Table A-6.4)
- Executable Object Code is compatible with target computer (Table A-6.5)
- Test coverage of high-level requirements is achieved (Table A-7.3)
- Test coverage of low-level requirements is achieved (Table A-7.4)
- Test coverage of software structure (modified condition/decision) is achieved (Table A-7.5)
- Test coverage of software structure (decision coverage) is achieved (Table A-7.6)
- Test coverage of software structure (statement coverage) is achieved (Table A-7.7)

TestConductor, being a software tool, might contain defects which theoretically could lead to misinterpreted tests (false negatives, or even worse, false positives), or wrong computation of requirements coverage, model coverage or code coverage. To reduce these risks to a minimum there are the following counter measures:

- TestConductor supports out-of-the-box a feature 'result verification'. For tests complying with specific requirements the result verification executes an independent consistency check on the test results.
- A TestConductor Validation Suite is available which consists of a rich set of tests for critical features of TestConductor. With this Validation Suite TestConductor functionality can be extensively tested for different environments. The tests of the Validation Suite can be traced back to features on the TestConductor specification. More information on the Validation Suite can be found in section 4.2.
- TestConductor and the development process of TestConductor is continuously audited by TÜV Süd Germany, and TestConductor is classified as T2 offline support tool according to IEC 61508-4 and EN 50128:2011. According to those standards of the other domains TestConductor is qualified to be applied in a safety related software development for all SIL levels according to IEC 61508, EN 50128 and for all ASIL levels according to ISO 26262.

3.2 Tool Developer Responsibilities and TQL Levels

The DO-330 describes the process and requirements for tool qualification in the context of development of software which is subject to the DO-178C and is therefore the main resource for this qualification guideline.

For our intent, to enable the qualification of TestConductor as a COTS testing tool, the chapters 11.3 (Qualifying COTS Tools) and ANNEX A (Tool Qualification Objectives) of DO-330 are of special interest. DO-330, section 11.3.1 says:

“For COTS tools, the tool qualification activities are typically performed by two entities:

- *Tool developer: The entity that developed the tool.*
- *Tool user: The entity that uses the tool in the scope of a given software project.*

Activities are defined below for both the COTS tool developer and tool user in order to satisfy all of the tool qualification objectives. It should be noted that for TQL-5 qualification, objectives may be satisfied without any tool qualification data from the tool developer.”

In the sequel of this section we focus on the obligations of the tool developer:

- Chapter 11.3 deals with the qualification of COTS tools. Specifically with objectives and activities typically applicable to the tool developer. Subsection 11.3.2 lists the tool developer activities, which are in essence providing a developer-TOR and additional preliminary tool qualification data. Table 11-1 (see Table Table 1 in this document) summarizes the tool developer objectives, to be addressed by performing the listed activities, in regard to the overall objectives in DO-330 ANNEX A. Since the nature of the DO-178 is objective based we can concentrate on the objectives in the table.
- ANNEX A contains a set of comprehensive tables which summarize all objectives of the DO-330 document by showing the applicability of each objective by TQL (Tool Qualification Level) for each tool life cycle data item.

ANNEX Table	A Objectives applicable to Tool Developer	Notes
T-0	Objectives 2,4 and 5: Applicable Others: Not applicable	<ul style="list-style-type: none"> • Objective 2: Tool developer provides a developer-TOR • Objectives 4 and 5 are performed using the developer-TOR • Other objectives: These are the tool user's responsibility.
T-1	All: Applicable	<ul style="list-style-type: none"> • Objectives 3 and 5: Fully applicable for the tool developer. • Other objectives: The scope of these objectives is limited to the specific activities of the tool development process.
T-2	All: Applicable	<ul style="list-style-type: none"> • The tool developer uses the developer-TOR to perform the activities of the tool development process.
T-3	All: Applicable	<ul style="list-style-type: none"> • Objectives 1 and 8 are performed using the developer-TOR
T-4 to T-7	All: Applicable	<ul style="list-style-type: none"> • The tool developer performs all activities.
T-8	All: Applicable	<ul style="list-style-type: none"> • Scope is limited to the data produced during the activities performed by the tool developer.
T-9	All: Applicable	<ul style="list-style-type: none"> • Scope is limited to the activities performed by the tool developer.
T-10	All: Not applicable	<ul style="list-style-type: none"> • Tool user performs these activities.

Table 1: Tool Developer Objectives, DO-330/T-11-1

Since the objectives for qualification of a tool depend on the assigned TQL of the tool, the TQL level of TestConductor has to be determined. The lower the TQL (being TQL 5 the lowest level), the smaller is the number of objectives which have to be fulfilled. The DO-178C elaborates on the TQL level of tools in section 12.2.2. Tools are classified according to three different criteria:

- **Criteria 1:** A tool whose output is part of the airborne software and thus could insert an error
- **Criteria 2:** A tool that automates verification process(es) and thus could fail to detect an error, and whose output is used to justify the elimination or reduction of:
 - Verification process(es) other than that automated by the tool, or
 - Development process(es) that could have an impact on the airborne software.
- **Criteria 3:** A tool that, within the scope of its intended use, could fail to detect an error.

The appropriate TQL level depends both on the software level (ranging from A to D) and the criteria as shown in Table *Table 2* (DO-178C/Table 12-1).

Software Level	Criteria 1	Criteria 2	Criteria 3
A	TQL-1	TQL-4	TQL-5
B	TQL-2	TQL-4	TQL-5
C	TQL-3	TQL-5	TQL-5
D	TQL-4	TQL-5	TQL-5

Table 2: Tool Qualification Level Determination, DO-178C Table 12-1

TestConductor being a Criteria 3 tool (TestConductor output is not part of the airborne software; no processes other than the automated ones are affected, resp. reduced or eliminated), is assigned TQL-5, the least demanding TQL¹.

3.3 Objectives for Tool Developer and Fulfillment by the TestConductor Qualification Kit

In this section we identify the tool developer objectives declared in the DO-330 and explain how the TestConductor Qualification Kit for DO-178C helps to fulfill these objectives to ease the qualification of TestConductor for a certain project.

ANNEX A in DO-330 contains a set of tables which summarize the objectives declared in the DO-330. We apply the table DO-330/11-1 (Tool Developer Objectives for COTS tools) to extract the subset of objectives which are in the responsibility of the tool developer. Of these objectives we select the even smaller subset of objectives which are relevant for TQL-5. Each objective references one or more related activities and one or more output documents. The activities lead to output documents we later focus on. For each output document we establish a match with one or more documents of the qualification kit to show the fulfillment of a particular objective.

Also note that TestConductor was audited by an independent certification authority (TÜV SÜD Product Service GmbH, Germany [3.]) in the context of a tool pre-qualification according to ISO 26262, IEC 61508, IEC 62304 and EN 50128. The audit included an evaluation of the development process as is in place for TestConductor.

2.2.1 DO-330, ANNEX A, Table T-0

¹ The DO-330 actually states in 11.3.1 that for TQL-5 qualification, objectives may be satisfied without any tool qualification data from the tool developer.

Table T-11-1 (DO-330) states that from the objectives listed in T-0 (DO-330/ANNEX-A) for tool developers only objectives 2, 4 and 5 are applicable. In T-0 we see that objective 4 is not relevant if the TQL is 5 (as is the TQL of TestConductor), leaving objectives 2 and 5.

Objective 2 (“Tool Operational Requirements are defined”, 5.1.1.a) references the activities 5.1.2.a, 5.1.2.b, 5.1.2.c and specifies as output the “Tool Operation Requirements” as described in 10.3.1.

This objective is fulfilled by providing a developer-TOR (also see note in T-11-1 regarding objective 2 of T-0). The TestConductor developer-TOR consists of several documents and is described in section 4.1.

Objective 5 (“Tool operation complies with the Tool Operational Requirements”, 6.2.1.b) references the activity 6.2.2.c and specifies as output the “Tool Operational Verification and Validation Cases and Procedures” (as described in 10.3.3) and “Tool Operational Verification and Validation Results” (as described in 10.3.4).

This objective is fulfilled by providing a test specification (as part of the TestConductor requirements document in the Developer-TOR), a Validation Suite (4.2) and a QA report (4.4).

2.2.2 DO-330, ANNEX A, Table T-1

Table T-11-1 (DO-330) states that from the objectives listed in T-1 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-1 which are required for TQL 5.

2.2.3 DO-330, ANNEX A, Table T-2

Table T-11-1 (DO-330) states that from the objectives listed in T-2 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-2 which are required for TQL 5.

2.2.4 DO-330, ANNEX A, Table T-3

Table T-11-1 (DO-330) states that from the objectives listed in T-3 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-3 which are required for TQL 5.

2.2.5 DO-330, ANNEX A, Table T-4

Table T-11-1 (DO-330) states that from the objectives listed in T-4 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-4 which are required for TQL 5.

2.2.6 DO-330, ANNEX A, Table T-5

Table T-11-1 (DO-330) states that from the objectives listed in T-5 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-5 which are required for TQL 5.

2.2.7 DO-330, ANNEX A, Table T-6

Table T-11-1 (DO-330) states that from the objectives listed in T-6 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-6 which are required for TQL 5.

2.2.8 DO-330, ANNEX A, Table T-7

Table T-11-1 (DO-330) states that from the objectives listed in T-7 (DO-330/ANNEX-A) for tool developers all objectives are applicable. However there are no objectives in T-7 which are required for TQL 5.

2.2.9 DO-330, ANNEX A, Table T-8

Only objective 1 and objective 4 are applicable for TQL 5.

Objective 1 (“Configuration Items are defined”) references the activities 7.2.1 (DO-330) and specifies as output the “Tool Configuration Management Records” as described in 10.1.13 (DO-330).

This objective is fulfilled since TestConductor releases have a unique Version number and build number for TestConductor itself and all contained sub tools. This version number is documented in the TestConductor Release Notes (4.4), which are part of every TestConductor installation.

Additionally TestConductor was audited by an independent certification authority (TÜV SÜD Product Service GmbH, Germany [3.]) in the context of a tool pre-qualification according to ISO 26262, IEC 61508, IEC 62304 and EN 50128. The audit included an evaluation of the development process as is in place for TestConductor. Configuration management was part of this audit. REF

Objective 4 (“Archive, retrieval, and release are established”) references the activities 7.2.7 (DO-330) and specifies as output the “Tool Configuration Management Records” as described in 10.1.13 (DO-330).

This objective is fulfilled since TestConductor development data (source code and development data) is maintained in a configuration management system with user authorization and backup functionality which conforms to the requirements stated in DO-330/7.2.7. The configuration information of a TestConductor release is contained in the TestConductor Release Notes, which are part of every TestConductor installation.

Additionally TestConductor was audited by an independent certification authority (TÜV SÜD Product Service GmbH, Germany [3.]) in the context of a tool pre-qualification according to ISO 26262, IEC 61508, IEC 62304 and EN 50128. The audit included an evaluation of the development process as is in place for TestConductor. Configuration management was part of this audit.

2.2.10 DO-330, ANNEX A, Table T-9

Only objective 2 and objective 5 are applicable for TQL 5.

Objective 2 (“Assurance is obtained that tool processes comply with approved plans”) references the activities 8.2.a, 8.2.c, 8.2.d, 8.2.f, 8.2.h, 8.2.i and specifies as output the “Tool Quality Assurance Records or Software Quality assurance Records” as described in 10.1.14.

According to DO330/T-9 for TQL-5 Tool Quality Assurance Records may be part of the Software Quality Assurance Records.

TestConductor is developed according to strict processes defined with IBM concerning change management and bug reporting. This includes reviews and quality assurance by IBM. These procedures were also audited in the TÜV Süd audit (4.8).

Objective 5 (“Tool conformity review is conducted”) references the activities 8.2.g, 8.2.h, 8.3 and specifies as output the “Tool Quality Assurance Records or Software Quality assurance Records” as described in 10.1.14.

According to DO330/T-9 for TQL-5 Tool Quality Assurance Records may be part of the Software Quality Assurance Records.

TestConductor is developed according to strict processes defined with IBM concerning change management and bug reporting. This includes reviews and quality assurance by IBM. These procedures were also audited in the TÜV Süd audit (4.8).

2.2.11 DO-330, ANNEX A, Table T-10

No objectives applicable for tool developers.

4. Overview about documents relevant for the TestConductor Qualification Kit for DO-178C

This section contains a list and description of documents which are part of the Qualification Kit or are referenced by this document.

4.1 Developer-TOR

The general content of a developer-TOR is described in DO-330/11.3.2.1. For TestConductor the developer-TOR consists of the TestConductor specification. The TestConductor specification contains the specification for the key features and related test scenarios. The TestConductor specification is part of the TestConductor Qualification Kit for DO-178C.

According to DO-330/11.3.2.1.f the user manuals may contribute to the developer-TOR, too. TestConductor provides a core user manual named “IBM Rational TestConductor User Guide”, which is part of the Rhapsody installation (4.5). Additionally there are tutorials for usage of RhapsodyInC++ and RhapsodyInC (4.6) and a Testing Cookbook (4.7) with advanced information.

4.2 TestConductor Validation Suite

In order to verify the TestConductor Tool Operational requirements, as described in detail in the TestConductor specification, a TestConductor Validation Suite is available under a Non-Disclosure Agreement free of charge for customers wishing to qualify the tool themselves.

The TestConductor Validation Suite consists of:

- An executable test environment written in Java to execute and evaluate tests.
- A set of tests (implementations of scenarios in the Test Specification)
- A manual how to execute the tests and read the results
- A guideline how to adapt the Validation Suite to specific operational environments.
- A QA-Report summarizing the expected and as is test execution results (4.3)

By executing the validation suite (for the designated environment) it can be verified that TestConductor tool operation complies with the Tool Operational Requirements. Each test can be traced back to feature test scenarios for all the tool operational requirements as described in the TestConductor specification (developer-TOR), hence establishing full traceability between tool operational requirements, tool features and tests.

The Validation Suite supports out of the box the following environments (OS/Compiler) and can be adjusted to support further environments:

Windows 7 (32/64): gcc (cygwin 32/64), MSVC9, MSVC 10, MSVC 11

Windows 8.1 (32/64): gcc (cygwin 32/64), MSVC9, MSVC 10, MSVC 11

Redhat Linux 6 (32/64): gcc

4.3 QA-Report

The QA-Report is part of the Validation Suite and contains a summary of all executed tests for the release the report was created for.

The QA-Report can be found in the Validation Suite (4.2) at:

[ReferenceResults/TC-QAReport-ValidationSuite.pdf]

4.4 TestConductor Release-Notes

The TestConductor Release notes are part of every installation of TestConductor and contain the version and build number a list of the new features and bug fixes for this and prior releases and a list of limitations.

The TestConductor Release-notes can be found at this location:

[<RhapsodyInstallationFolder>/Doc/pdf_docs/RTC_Release_Notes.pdf]

4.5 TestConductor User Guide

The TestConductor User Guide explains the features and usage of the tool and is part of the documentation of the Rhapsody installation.

The TestConductor User Guide can be found at this location:

[<RhapsodyInstallationFolder>/Doc/pdf_docs/RTC_User_Guide.pdf]

4.6 TestConductor Tutorials

The TestConductor Tutorials are available for C and C++ and are part of the documentation of the Rhapsody installation.

The TestConductor Tutorials can be found at this location:

[<RhapsodyInstallationFolder>/Doc/pdf_docs/TestConductor_Tutorial_Cpp.pdf]

[<RhapsodyInstallationFolder>/Doc/pdf_docs/TestConductor_Tutorial_C.pdf]

4.7 TestConductor Testing Cookbook

The Testing Cookbook contains advanced information for special scenarios in a FAQ style is part of the documentation of the Rhapsody installation.

The Testing Cookbook can be found at this location:

[<RhapsodyInstallationFolder>/Doc/html_docs/TestingCookbook/index.html]

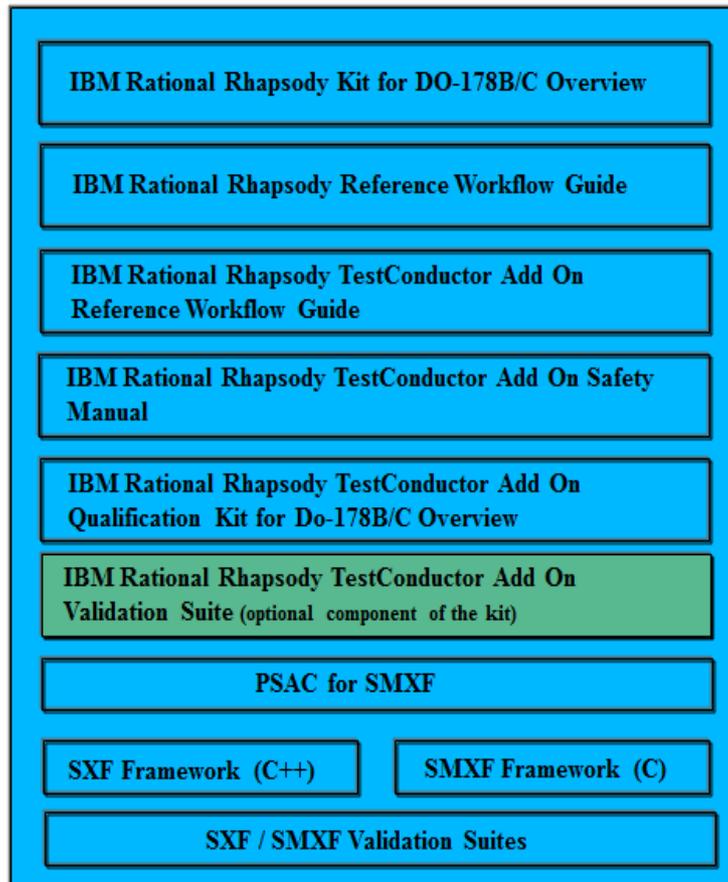
4.8 TÜV Süd Certificate and Certification Report for IBM Rational Rhapsody TestConductor

TÜV Süd Germany was asked to “pre-qualify” TestConductor according to safety standards outside of the Aerospace domain. Part of the “pre-qualification” work was an audit

- of the in place development and verification/validation procedures for TestConductor, and
- of the TestConductor validation suite to verify correct functioning.

The results of this “pre-qualification” work is documented in a report to the certificate. TestConductor customers can access this document. The “pre-qualification” is repeated on a regular basis, in fact for all new major versions of TestConductor.

4.9 IBM Rational Rhapsody Kit for DO-178B-C



The IBM Rational Rhapsody Kit for DO-178B/C is not dealing with the qualification of tools. Instead, the IBM Rational Rhapsody Kit for DO-178B/C includes guidance on how to capably develop safety-related software with IBM Rational Rhapsody by meeting the objectives described in the safety-related standards DO-178B [4.], DO-178C [5.], and supplement DO-331 [6.].

[<RhapsodyInstallationFolder>/Doc/pdf_docs/DO178B-C_Kit/....pdf]

The kit contains the following artifacts.

4.9.1 IBM Rational Rhapsody Kit for DO-178B/C Overview

The document provides an overview about all the artifacts of the kit.

4.9.1 IBM Rational Rhapsody Reference Workflow Guide

The IBM Rational Rhapsody Reference Workflow Guide document focuses on developing safety-related projects with Rational Rhapsody. When developing safety-related software additional quality objectives have to be met in order to produce and deliver “safe” systems. These additional quality objectives essentially depend on

- a specific industrial domain where the product under development shall be deployed,
- an appropriate safety standard that must be applied for a particular domain.

In the IBM Rational Rhapsody Reference Workflow Guide document, focus is placed on UML model-based development and testing of safety-related software with IBM Rational Rhapsody. Also included

is the *IBM Rational Rhapsody Reference Workflow* which provides a broader view of the development process spanning requirements, available methods, solutions, and tools.

4.9.2 IBM Rational Rhapsody TestConductor Add On Reference Workflow Guide

The IBM Rational Rhapsody TestConductor Add On Reference Workflow Guide document serves as a reference for testing activities to perform in a model based development process using IBM Rational Rhapsody with the IBM Rational Rhapsody TestConductor Add On. It complements the document IBM Rational Rhapsody Reference Workflow Guide that focuses on the model based development with IBM Rational Rhapsody in safety-related projects. The IBM Rational Rhapsody TestConductor Add On Reference Workflow Guide document provides further information and describes variations of the IBM Rational Rhapsody Reference Workflow, focusing on testing methods as provided by IBM Rational Rhapsody TestConductor Add On.

4.9.3 IBM Rational Rhapsody TestConductor Add On Safety Manual

The IBM Rational Rhapsody TestConductor Add On Safety Manual provides guidance on using IBM Rational Rhapsody TestConductor for testing activities in a model based development process when developing safety-related software. This safety manual complements the previous documents, and provides additional information for installing and using IBM Rational Rhapsody TestConductor in safety-related projects.

4.9.4 IBM Rational Rhapsody TestConductor Add On Qualification Kit for DO-178B/C Overview

This document.

4.9.5 IBM Rational PSAC for SMXF

The IBM Rational Plan for Software Aspects of Certification for SMXF (PSAC for SMXF) provides information to the user regarding SMXF specification, implementation, testing, etc. This information supports customers during their product certification efforts.

4.9.6 IBM Rational Rhapsody SXF / SMXF Frameworks (C++ / C)

IBM Rational Rhapsody provides an Object eXecution Framework (OXF) library that is used for standard C and C++ code generation. For safety-related development IBM Rational Rhapsody provides two dedicated libraries called Simplified eXecution Framework (SXF) and Simplified MicroC eXecution Framework (SMXF). The SXF library is the safety-related C++ framework library. It's a comprehensive C++ library that is suitable to be used in safety-related production C++ code environments. The C counterpart of the SXF library is the SMXF library. This is a comprehensive C library that is suitable to be used in safety-related production C code environments.

Both libraries are delivered as part of the standard Rhapsody installation kit for Windows.

4.9.7 SXF / SMXF Validation Suites.

In order to verify the SXF / SMXF Tool Operational requirements Validation Suites are available free of charge for customers.

5. List of References

1. RTCA: <http://www.rtca.org>
2. EUROCAE, The European Organisation for Civil Aviation Equipment: <http://www.eurocae.net/>
3. TÜV-Süd http://www.tuev-sued.de/home_de
4. Software Considerations in Airborne Systems and Equipment Certification, RTCA Inc., RTCA DO-178B. 1992.
5. Software Considerations in Airborne Systems and Equipment Certification, RTCA Inc., RTCA DO-178C. 2011.
6. Model-Based Development and Verification – Supplement to DO-178C and DO-278A, RTCA Inc., RTCA DO-331. 2011.
7. Software Tool Qualification Considerations, RTCA Inc., RTCA DO-330. 2011.