

RTCA, Inc.  
1150 18th Street, NW, Suite 910  
Washington, DC 20036

**Model-Based Development and Verification  
Supplement to DO-178C and DO-278A**

Copies of this document may be obtained from

RTCA, Inc.

Telephone: 202-833-9339

Facsimile: 202-833-9434

Internet: [www.rtca.org](http://www.rtca.org)

Please visit the RTCA Online Store for document pricing and ordering information.

## FOREWORD

This document was prepared by RTCA Special Committee 205 (SC-205) and EUROCAE Working Group 71 (WG-71) and approved by the RTCA Program Management Committee (PMC) on December 13, 2011.

RTCA, Incorporated is a not-for-profit corporation formed to advance the art and science of aviation and aviation electronic systems for the benefit of the public. The organization functions as a Federal Advisory Committee and develops consensus-based recommendations on contemporary aviation issues. RTCA's objectives include but are not limited to:

- coalescing aviation system user and provider technical requirements in a manner that helps government and industry meet their mutual objectives and responsibilities;
- analyzing and recommending solutions to the system technical issues that aviation faces as it continues to pursue increased safety, system capacity, and efficiency;
- developing consensus on the application of pertinent technology to fulfill user and provider requirements, including development of minimum operational performance standards for electronic systems and equipment that support aviation; and
- assisting in developing the appropriate technical material upon which positions for the International Civil Aviation Organization and the International Telecommunication Union and other appropriate international organizations can be based.

The organization's recommendations are often used as the basis for government and private sector decisions as well as the foundation for many Federal Aviation Administration Technical Standard Orders.

Since the RTCA is not an official agency of the United States Government, its recommendations may not be regarded as statements of official government policy unless so enunciated by the U.S. government organization or agency having statutory jurisdiction over any matters to which the recommendations relate.

This Page Intentionally Left Blank

## TABLE OF CONTENTS

MB.1.0	INTRODUCTION.....	1
MB.1.1	Purpose.....	1
MB.1.2	Scope.....	1
MB.1.3	Relationship to Other Documents.....	2
MB.1.4	How to Use This Document.....	2
MB.1.5	Document Overview.....	2
MB.1.6	Characteristics of Model-Based Development and Verification.....	3
MB.1.6.1	Requirements From Which the Model is Developed.....	3
MB.1.6.2	Specification Models and Design Models.....	3
MB.1.6.3	Examples of Model Usage.....	4
MB.2.0	SYSTEM ASPECTS RELATING TO SOFTWARE DEVELOPMENT.....	5
MB.2.1	System Requirements Allocation to Software.....	5
MB.2.2	Information Flow Between System and Software Life Cycle Processes.....	5
MB.2.2.1	Information Flow from System Processes to Software Processes.....	7
MB.2.2.2	Information Flow from Software Processes to System Processes.....	8
MB.2.2.3	Information Flow between Software Processes and Hardware Processes.....	8
MB.2.3	System Safety Assessment Process and Software Level.....	8
MB.2.4	Architectural Considerations.....	8
MB.2.5	Software Considerations in System Life Cycle Processes.....	9
MB.2.5.1	Parameter Data Items.....	9
MB.2.5.2	User-Modifiable Software.....	9
MB.2.5.3	Commercial-Off-The-Shelf Software.....	9
MB.2.5.4	Option-Selectable Software.....	10
MB.2.5.5	Field-Loadable Software.....	10
MB.2.5.6	Software Considerations in System Verification.....	10
MB.2.6	System Considerations in Software Life Cycle Processes.....	10
MB.3.0	SOFTWARE LIFE CYCLE.....	11
MB.3.1	Software Life Cycle Processes.....	11
MB.3.2	Software Life Cycle Definition.....	11
MB.3.3	Transition Criteria Between Processes.....	11
MB.4.0	SOFTWARE PLANNING PROCESS.....	13
MB.4.1	Software Planning Process Objectives.....	13
MB.4.2	Software Planning Process Activities.....	14
MB.4.3	Software Plans.....	15
MB.4.4	Software Life Cycle Environment Planning.....	16
MB.4.4.1	Software Development Environment.....	16
MB.4.4.2	Language and Compiler Considerations.....	16
MB.4.4.3	Software Test Environment.....	17
MB.4.4.4	Simulation Environment.....	17
MB.4.5	Software Development Standards.....	18
MB.4.6	Review of the Software Planning Process.....	18
MB.5.0	SOFTWARE DEVELOPMENT PROCESSES.....	19
MB.5.1	Software Requirements Process.....	20
MB.5.1.1	Software Requirements Process Objectives.....	20
MB.5.1.2	Software Requirements Process Activities.....	20
MB.5.2	Software Design Process.....	22
MB.5.2.1	Software Design Process Objectives.....	22

MB.5.2.2	Software Design Process Activities.....	22
MB.5.2.3	Designing for User-Modifiable Software.....	23
MB.5.2.4	Designing for Deactivated Code .....	24
MB.5.3	Software Coding Process.....	24
MB.5.4	Integration Process .....	24
MB.5.4.1	Integration Process Objectives .....	24
MB.5.4.2	Integration Process Activities.....	24
MB.5.5	Software Development Process Traceability.....	24
MB.6.0	SOFTWARE VERIFICATION PROCESS .....	25
MB.6.1	Purpose of Software Verification .....	25
MB.6.2	Overview of Software Verification Process Activities.....	26
MB.6.3	Software Reviews and Analyses .....	27
MB.6.3.1	Reviews and Analyses of High-Level Requirements.....	27
MB.6.3.2	Reviews and Analyses of Low-Level Requirements.....	28
MB.6.3.3	Reviews and Analyses of Software Architecture .....	29
MB.6.3.4	Reviews and Analyses of Source Code .....	29
MB.6.3.5	Reviews and Analyses of the Outputs of the Integration Process .....	30
MB.6.4	Software Testing.....	30
MB.6.5	Software Verification Process Traceability.....	30
MB.6.6	Verification of Parameter Data Items.....	30
MB.6.7	Model Coverage Analysis for Design Models .....	30
MB.6.7.1	Model Coverage Analysis Criteria .....	31
MB.6.7.2	Model Coverage Analysis Resolution .....	32
MB.6.8	Model Simulation.....	33
MB.6.8.1	Model Simulation for Verification of the Model.....	33
MB.6.8.2	Model Simulation for Verification of the Executable Object Code .....	35
MB.6.8.3	Simulation Cases, Procedures and Results.....	37
MB.6.8.3.1	Development of Simulation Cases, Procedures and Results.....	37
MB.6.8.3.2	Reviews and Analyses of Simulation Cases, Procedures and Results.....	37
MB.7.0	SOFTWARE CONFIGURATION MANAGEMENT PROCESS.....	39
MB.7.1	Software Configuration Management Process Objectives .....	39
MB.7.2	Software Configuration Management Process Activities.....	40
MB.7.2.1	Configuration Identification .....	40
MB.7.2.2	Baselines and Traceability.....	40
MB.7.2.3	Problem Reporting, Tracking, and Corrective Action.....	41
MB.7.2.4	Change Control.....	41
MB.7.2.5	Change Review.....	41
MB.7.2.6	Configuration Status Accounting .....	41
MB.7.2.7	Archive, Retrieval, and Release .....	41
MB.7.3	Data Control Categories .....	41
MB.7.4	Software Load Control .....	42
MB.7.5	Software Life Cycle Environment Control.....	42
MB.8.0	SOFTWARE QUALITY ASSURANCE PROCESS.....	43
MB.8.1	Software Quality Assurance Process Objectives.....	43
MB.8.2	Software Quality Assurance Process Activities .....	43
MB.8.3	Software Conformity Review.....	44

MB.9.0	CERTIFICATION LIAISON PROCESS.....	45
MB.9.1	Means of Compliance and Planning.....	45
MB.9.2	Compliance Substantiation.....	45
MB.9.3	Minimum Software Life Cycle Data Submitted to Certification Authority .....	46
MB.9.4	Software Life Cycle Data Related to Type Design .....	46
MB.10.0	OVERVIEW OF CERTIFICATION PROCESS .....	47
MB.10.1	Certification Basis .....	47
MB.10.2	Software Aspects of Certification .....	47
MB.10.3	Compliance Determination .....	47
MB.11.0	SOFTWARE LIFE CYCLE DATA.....	49
MB.11.1	Plan for Software Aspects of Certification.....	50
MB.11.2	Software Development Plan.....	51
MB.11.3	Software Verification Plan.....	51
MB.11.4	Software Configuration Management Plan.....	53
MB.11.5	Software Quality Assurance Plan.....	54
MB.11.6	Software Requirements Standards .....	54
MB.11.7	Software Design Standards .....	54
MB.11.8	Software Code Standards .....	54
MB.11.9	Software Requirements Data.....	54
MB.11.10	Design Description .....	55
MB.11.11	Source Code.....	55
MB.11.12	Executable Object Code.....	55
MB.11.13	Software Verification Cases and Procedures.....	56
MB.11.14	Software Verification Results .....	56
MB.11.15	Software Life Cycle Environment Configuration Index.....	56
MB.11.16	Software Configuration Index.....	57
MB.11.17	Problem Reports .....	58
MB.11.18	Software Configuration Management Records.....	58
MB.11.19	Software Quality Assurance Records .....	58
MB.11.20	Software Accomplishment Summary .....	58
MB.11.21	Trace Data.....	58
MB.11.22	Parameter Data Item File .....	58
MB.11.23	Software Model Standards.....	58
MB.12.0	ADDITIONAL CONSIDERATIONS.....	61
MB.12.1	Use of Previously Developed Software.....	61
MB.12.1.1	Modifications of Previously Developed Software.....	61
MB.12.1.2	Change of Aircraft Installation.....	61
MB.12.1.3	Change of Application or Development Environment .....	62
MB.12.1.4	Upgrading a Development Baseline.....	63
MB.12.1.5	Software Configuration Management Considerations .....	64
MB.12.1.6	Software Quality Assurance Considerations .....	64
MB.12.2	Tool Qualification .....	64
MB.12.2.1	Determining if Tool Qualification is Needed.....	64
MB.12.2.2	Determining the Tool Qualification Level .....	64
MB.12.2.3	Tool Qualification Process .....	64
MB.12.3	Alternative Methods.....	65
MB.12.3.1	Exhaustive Input Testing.....	65
MB.12.3.2	Considerations for Multiple-Version Dissimilar Software Verification .....	65
MB.12.3.3	Software Reliability Models.....	65
MB.12.3.4	Product Service History.....	65

MB.12.3.4.1	Relevance of Service History .....	65
MB.12.3.4.2	Sufficiency of Accumulated Service History .....	65
MB.12.3.4.3	Collection, Reporting, and Analysis of Problem Reports Found During Service History .....	65
MB.12.3.4.4	Service History Information to be Included in the Plan for Software Aspects of Certification.....	66
ANNEX MB.A – PROCESS OBJECTIVES AND OUTPUTS BY SOFTWARE LEVEL IN DO-178C .....		67
ANNEX MB.B - ACRONYMS AND GLOSSARY OF TERMS .....		81
ANNEX MB.C – PROCESS OBJECTIVES AND OUTPUTS BY ASSURANCE LEVEL IN DO-278A .....		83
APPENDIX MB.A – COMMITTEE MEMBERSHIP.....		97
APPENDIX MB.B – FREQUENTLY ASKED QUESTIONS AND DISCUSSION PAPERS .....		107
MB.B.1	FAQ #1: What is the Data Control Category of a model?.....	107
MB.B.2	FAQ #2: Which verification objectives does model simulation, by itself support?....	107
MB.B.3	FAQ #3: What verification objectives can be met by combining model simulation with other tools or methods? .....	107
MB.B.4	FAQ #4: If a model is used to represent requirements and an autocode generator is not used, does the supplement apply? .....	108
MB.B.5	FAQ #5: How should the applicant develop and verify manually written Source Code, that is invoked by the code generated from the Design Model?.....	108
MB.B.6	FAQ #6: When Source Code is automatically generated from a Design Model, what type of testing should be performed to provide assurance of the correctness of the integration of this Source Code with manually written Source Code? .....	109
MB.B.7	FAQ #7: When using models for verification, should expected results be determined prior to test execution? .....	109
MB.B.8	FAQ #8: Is a model subject to Tool Qualification? .....	110
MB.B.9	FAQ #9: What is an example of a model that is considered part of the test environment and what activities are applicable to that model?.....	110
MB.B.10	FAQ #10: Can a single Software Model Standard be applied to both Specification Models and Design Models? .....	111
MB.B.11	FAQ #11: May the applicant use the model coverage analysis activity to achieve the structural coverage analysis objectives?.....	111
MB.B.12	FAQ #12: What are the independence issues when a Design Model is used for both code generation and test generation?.....	112
MB.B.13	FAQ #13: How does the supplement apply if a Design Model that is used in the software development process is part of the system-level life cycle data, as in example 5 of Table MB.1-1 (that is, the Design Model also contains system requirements allocated to software)?.....	113
MB.B.14	FAQ #14: How does the supplement apply if the requirements from which the Design Model was developed are part of the system life cycle data, as in example 4 and 5 of Table MB.1-1 (that is, high-level requirements per DO-178C and this supplement are at the system level in the form of system requirements allocated to software)? .....	113
MB.B.15	FAQ #15: Do data files associated with models need to be treated as parameter data items?.....	114
MB.B.16	FAQ #16: Can simulation support the assessment of test coverage of the low-level requirements contained in a Design Model? .....	115
MB.B.17	DP #1: Examples of model-based development and the relationship between a Design Model or a Specification Model and DO178C high-level requirements, low-level requirements, and software architecture.....	115

MB.B.18 DP #2: Information on the usage of libraries in a Model-Based Development and Verification processes. ....	122
--	-----

### LIST OF FIGURES

FIGURE MB.2-1	INFORMATION FLOW BETWEEN SYSTEM AND SOFTWARE LIFE CYCLE PROCESSES.....	6
FIGURE MB.DP1-1	KEY TO READING DP1 DIAGRAMS.....	116
FIGURE MB.DP1-2	EXAMPLE A: SEPARATE MODELS USED TO EXPRESS LLR AND SW ARCHITECTURE .....	117
FIGURE MB.DP1-3	EXAMPLE B: ONE MODEL USED TO EXPRESS HLR, LLR/SW ARCHITECTURE WITH HLR PROVIDED BY SRATS.....	118
FIGURE MB.DP1-4	EXAMPLE C: SEPARATE MODELS USED TO EXPRESS HLR AND LLR/SW ARCHITECTURE.....	119
FIGURE MB.DP1-5	EXAMPLE D: ONE MODEL USED TO EXPRESS HLR AND ONLY HLR.....	120
FIGURE MB.DP1-6	EXAMPLE E: ONE MODEL PROVIDED BY SRATS USED TO EXPRESS SYSTEM REQUIREMENTS, HLR, LLR/SW ARCHITECTURE .....	121

### LIST OF TABLES

TABLE MB.1-1	MODEL USAGE EXAMPLES .....	4
TABLE MB.6-1	MODEL COVERAGE CRITERIA EXAMPLE .....	32
TABLE MB.7-1	SCM PROCESS ACTIVITIES ASSOCIATED WITH CC1 AND CC2 DATA .....	42
TABLE MB.A-1	SOFTWARE PLANNING PROCESS.....	68
TABLE MB.A-2	SOFTWARE DEVELOPMENT PROCESSES .....	69
TABLE MB.A-3	VERIFICATION OF OUTPUTS OF SOFTWARE REQUIREMENTS PROCESS .....	71
TABLE MB.A-4	VERIFICATION OF OUTPUTS OF SOFTWARE DESIGN PROCESS .....	72
TABLE MB.A-5	VERIFICATION OF OUTPUTS OF SOFTWARE CODING & INTEGRATION PROCESSES .....	74
TABLE MB.A-6	TESTING OF OUTPUTS OF INTEGRATION PROCESS .....	75
TABLE MB.A-7	VERIFICATION OF VERIFICATION PROCESS RESULTS .....	76
TABLE MB.A-8	SOFTWARE CONFIGURATION MANAGEMENT PROCESS.....	78
TABLE MB.A-9	SOFTWARE QUALITY ASSURANCE PROCESS .....	79
TABLE MB.A-10	CERTIFICATION LIAISON PROCESS .....	80
TABLE MB.C-1	SOFTWARE PLANNING PROCESS.....	84
TABLE MB.C-2	SOFTWARE DEVELOPMENT PROCESSES .....	85
TABLE MB.C-3	VERIFICATION OF OUTPUTS OF SOFTWARE REQUIREMENTS PROCESS .....	87
TABLE MB.C-4	VERIFICATION OF OUTPUTS OF SOFTWARE DESIGN PROCESS .....	88
TABLE MB.C-5	VERIFICATION OF OUTPUTS OF SOFTWARE CODING & INTEGRATION PROCESSES .....	90
TABLE MB.C-6	TESTING OF OUTPUTS OF INTEGRATION PROCESS.....	91
TABLE MB.C-7	VERIFICATION OF VERIFICATION PROCESS RESULTS .....	92
TABLE MB.C-8	SOFTWARE CONFIGURATION MANAGEMENT PROCESS .....	94
TABLE MB.C-9	SOFTWARE QUALITY ASSURANCE PROCESS .....	95
TABLE MB.C-10	SOFTWARE APPROVAL PROCESS.....	96

This Page Intentionally Left Blank

---

**MB.1.0 INTRODUCTION**

A model is an abstract representation of a set of software aspects of a system that is used to support the software development process or the software verification process. This supplement addresses model(s) that have the following characteristics:

- a. The model is completely described using an explicitly identified modeling notation. The modeling notation may be graphical and/or textual.
- b. The model contains software requirements and/or software architecture definition.
- c. The model is of a form and type that is used for direct analysis or behavioral evaluation as supported by the software development process or the software verification process.

The following are not considered as models within this supplement:

- Figures without syntax/semantics (these may be illustrative).
- Equations related to natural language sentences.

The use of models may bring the benefits and capabilities of:

- Providing unambiguous expression of requirements and architecture.
- Supporting the use of automated code generation.
- Supporting the use of automated test generation.
- Supporting the use of analysis tools for verification of requirements and architecture.
- Supporting the use of simulation for partial verification of requirements, architecture, and/or Executable Object Code.

Since the publication of DO-178B, advances and experience have been gained in model-based development and verification, their application, and supporting tools. As the use of this technology for critical software applications in avionics has increased, there are a number of issues that need to be considered to ensure the safety and integrity goals are met. Clarifying each of these issues will ease the application of model-based development and verification; therefore, this supplement provides guidance for applicants and certification or approval authorities to facilitate the use of this technology.

**MB.1.1 Purpose**

This supplement contains modifications and additions to DO-178C objectives, activities, explanatory text, and software life cycle data that should be addressed when model-based development and verification are used as part of the software life cycle. This includes the artifacts that would be expressed using models and the verification evidence that could be derived from them. Therefore, this supplement also applies to the models developed in the system process that define software requirements or software architecture.

**MB.1.2 Scope**

This supplement discusses the use of model-based development and verification in the software life cycle for software that is produced in accordance with DO-178C. If the applicant is planning to use model-based development and verification, then the applicant