

## TABLE OF CONTENTS

<b>LIST OF ILLUSTRATIONS .....</b>	<b>9</b>
<b>LIST OF TABLES.....</b>	<b>12</b>
<b>ACRONYMS .....</b>	<b>13</b>
<b>ABSTRACT .....</b>	<b>16</b>
<b>CHAPTER 1: INTRODUCTION .....</b>	<b>18</b>
<b>1.1 Problem Definition .....</b>	<b>24</b>
<b>1.2 Thesis Organization .....</b>	<b>28</b>
<b>CHAPTER 2: RELATED TECHNOLOGIES AND EARLIER WORK.....</b>	<b>29</b>
<b>2.1 Model-Based Software Engineering Process .....</b>	<b>29</b>
<b>2.2 Model-Based Testing Methodologies .....</b>	<b>32</b>
<b>2.3 Automated Test Case Generation using UML Constructs .....</b>	<b>36</b>
<b>2.4 DEVS-Based Bifurcated Model-Continuity Process.....</b>	<b>41</b>
<b>2.5 Distributed Modeling and Simulation .....</b>	<b>45</b>
<b>CHAPTER 3: DEVS MODELING AND SIMULATION FRAMEWORK.....</b>	<b>49</b>
<b>3.1 DEVS System Specifications .....</b>	<b>51</b>
3.1.1 Hierarchy of System Specifications.....	51
3.1.2 Framework for Modeling & Simulation .....	54
3.1.3 Model Continuity .....	55
<b>3.2 Model/View/Controller (MVC) Paradigm and DEVS Framework.....</b>	<b>56</b>
3.2.1 Real-Time Control and Visualization Limitations of Existing Network Simulators.....	57
3.2.2 Enhanced MVC.....	59
<b>3.3 Dynamic Model and Simulation Reconfiguration .....</b>	<b>61</b>
3.3.1 Variable Structure DEVS.....	61
3.3.2 Implementation of Variable Structure in Extended MVC.....	64
3.3.3 Notion of System Steady State.....	65

## **TABLE OF CONTENTS - *CONTINUED***

<b>3.4</b>	<b>Dynamic Simulation Control .....</b>	<b>67</b>
3.4.1	DEVS Simulation Engine .....	67
3.4.2	Interrupt Handling .....	69
3.4.3	The Notion of “Simulation Control” Explored.....	70
3.4.4	Parameter Control.....	72
3.4.5	Synopsis.....	73
<b>CHAPTER 4: REQUIREMENT SPECIFICATIONS AND AUTOMATED DEVS MODEL GENERATION .....</b>		
		<b>75</b>
<b>4.1</b>	<b>State-Based System Specifications .....</b>	<b>77</b>
4.1.1	Sample Example.....	80
<b>4.2</b>	<b>Message-Based System Specifications with Restricted Natural Language Processing .....</b>	<b>85</b>
4.2.1	Sample Example:.....	86
4.2.2	Transformation of Rules to universal Primitives: .....	89
4.2.3	Design of Entity Node model with multiple message streams: .....	90
<b>4.3</b>	<b>BPEL/BPMN-Based System Requirement Specifications .....</b>	<b>95</b>
<b>4.4</b>	<b>Scenario-Based Systems using DoDAF .....</b>	<b>101</b>
4.4.1	DODAF Specifications .....	103
4.4.2	Motivation for DoDAF-to-DEVS mapping.....	107
4.4.3	From OV-6 UML diagrams to DEVS component behavior specifications .....	110
4.4.4	Representing DoDAF within the System Entity Structure: Multiple Aspects .....	116
4.4.5	Deriving testable behaviors from DoDAF specification .....	118
<b>CHAPTER 5: AUTOMATED MODEL-BASED TEST CASE GENERATION ....</b>		
		<b>125</b>
<b>5.1</b>	<b>Automated Test Case Generator: Concept.....</b>	<b>126</b>
<b>5.2</b>	<b>Automated Testing Methodology .....</b>	<b>131</b>
5.2.1	Test Model Generator .....	132
5.2.2	Test Driver .....	134
<b>5.3</b>	<b>Synopsis .....</b>	<b>137</b>
<b>CHAPTER 6: NET-CENTRIC MODEL EXECUTION USING SERVICE ORIENTED ARCHITECTURE .....</b>		
		<b>140</b>
<b>6.1</b>	<b>DEVSML: Automating DEVS Execution over SOA Towards Transparent Simulators .....</b>	<b>140</b>
6.1.1	Overview of DEVSML .....	142
6.1.2	DEVS DTDs and their Standardization .....	146
6.1.3	Web Services Architecture for DEVSML.....	152

## **TABLE OF CONTENTS - *CONTINUED***

<b>6.2</b>	<b>SOADEVS: Remote Execution of DEVS using Simulation Service .....</b>	<b>156</b>
6.2.1	WWW and Distributed Simulation.....	157
6.2.2	Abstraction of a Coupled model as an Atomic model with DEVS State Machine .....	162
6.2.3	Message Serialization .....	164
6.2.4	Details about the server architecture.....	166
6.2.5	DEVSMML and SOADEVS .....	172
<b>CHAPTER 7: DEVS UNIFIED PROCESS: PUTTING IT ALL TOGETHER .....174</b>		
<b>7.1</b>	<b>Automated DEVS Model Generation and DEVSMML .....</b>	<b>177</b>
<b>7.2</b>	<b>DEVSMML Collaborative Development.....</b>	<b>180</b>
<b>7.3</b>	<b>Automated Test-case Generation from DEVS models .....</b>	<b>182</b>
<b>7.4</b>	<b>SOADEVS: Net-centric Execution using Simulation Service.....</b>	<b>183</b>
<b>7.5</b>	<b>The Complete Process .....</b>	<b>185</b>
<b>CHAPTER 8: PROJECTS FROM WHICH DUNIP EVOLVED .....187</b>		
<b>8.1</b>	<b>Joint Close Air Support (JCAS) Model .....</b>	<b>190</b>
8.1.1	State-based approach .....	190
8.1.2	BPMN/BPEL based approach .....	193
8.1.3	Message-Based Restricted NLP-based approach.....	197
8.1.4	Automated test case generation for JCAS.....	198
8.1.5	Net-centric Execution of JCAS .....	199
<b>8.2</b>	<b>DoDAF-based Activity Scenario .....</b>	<b>202</b>
8.2.1	Example: Implementation of an Activity Component .....	202
8.2.2	Activity taken from Zinn as an example .....	204
8.2.3	DEVS Interpretation of Activity 6.....	207
8.2.4	Synopsis.....	217
<b>8.3</b>	<b>Link-16 ATC-Gen Project at JITC.....</b>	<b>218</b>
8.3.1	Auto Correlation Scenario .....	219
8.3.2	Auto Correlation Experiment Setup & Results.....	220
8.3.3	Testing Status.....	223
<b>8.4</b>	<b>GENETSCOPE Project at JITC.....</b>	<b>224</b>
8.4.1	SCOPE Command and DoDAF .....	228
8.4.2	SCOPE Architecture Implementation Using Enhanced MVC .....	234
8.4.3	Implications of the Example Above and NR-KPP.....	244

**TABLE OF CONTENTS - *CONTINUED***

<b>CHAPTER 9: DISCUSSION .....</b>	<b>248</b>
9.1 MDA and DUNIP .....	248
9.2 DUNIP and SCR .....	252
<b>CHAPTER 10: CONCLUSIONS AND FUTURE WORK .....</b>	<b>253</b>
10.1 Future Work .....	258
<b>REFERENCES .....</b>	<b>261</b>

## LIST OF ILLUSTRATIONS

<b>Figure 1.1:</b> Bifurcated Model-Continuity based System Life-cycle Process.....	25
<b>Figure 2.1:</b> Graphical process extended further from [Utt06] .....	36
<b>Figure 2.2:</b> Summarizing Model-based Testing .....	38
<b>Figure 2.3:</b> Test Scenario Generation based on requirement specifications.....	39
<b>Figure 2.4:</b> Bifurcated DEVS-to-DODAF System Lifecycle Development Process.....	43
<b>Figure 3.1:</b> Framework entities and relationships .....	54
<b>Figure 3.2:</b> Enhanced MVC paradigm with DEVS M&S framework.....	60
<b>Figure 3.3:</b> DEVS simulation protocol .....	68
<b>Figure 3.4:</b> Hierarchical simulator assignment for a hierarchical model .....	68
<b>Figure 3.5:</b> Automated test suite execution .....	70
<b>Figure 4.1:</b> DEVS state machine Document Type Description (statemachine.dtd).....	78
<b>Figure 4.2:</b> XML transformation of JTAC state machine described in tabular format ...	82
<b>Figure 4.3:</b> Generated DEVSJAVA code from valid jtac.xml in Figure 4.2.....	84
<b>Figure 4.4:</b> Rules for Restricted NLP based Requirement Specifications.....	86
<b>Figure 4.5:</b> Simon Says in English language .....	87
<b>Figure 4.6:</b> Universal State Machine (USM) for Rule-base Requirement Specifications	90
<b>Figure 4.7:</b> Graphical structure of internals of node entity with two message streams ...	92
<b>Figure 4.8:</b> Constructor for Node entity of the node diagram in Figure 4.7 .....	93
<b>Figure 4.9:</b> Various library functions supporting automated node coupling relations ....	94
<b>Figure 4.10:</b> Sample BPMN diagram .....	96
<b>Figure 4.11:</b> View of Web Service implemented as Web Service (courtesy: IBM) .....	97
<b>Figure 4.12:</b> Overview of BPEL-to-DEVS process.....	98
<b>Figure 4.13:</b> BPEL-to-DEVS transformation.....	99
<b>Figure 4.14:</b> WSDL-to-DEVS transformation.....	100
<b>Figure 4.15:</b> Snapshot of a BPMN-to-DEVS Transformation tool.....	101
<b>Figure 4.16:</b> Linkages among Views .....	105
<b>Figure 4.17:</b> DoDAF/DEVS execution roadmap .....	107
<b>Figure 4.18:</b> Development of DEVS Description model from UML Timing-Sequence Thread .....	111
<b>Figure 4.19:</b> Representing DoD AF within the SES framework.....	118
<b>Figure 4.20:</b> SES for enhanced DoDAF with a focus on OV .....	119
<b>Figure 4.21:</b> DEVS Model generation from various types of Requirement Specifications .....	124
<b>Figure 5.1:</b> ATC-Gen Development .....	128
<b>Figure 5.2:</b> IF-THEN rule format .....	128
<b>Figure 5.3:</b> XML RuleSet .....	130
<b>Figure 5.4:</b> Overview of ATC-Gen Tool Development .....	132
<b>Figure 5.5:</b> Test Model Generator .....	134
<b>Figure 5.6:</b> Enhanced MSVC paradigm with multiple controllers.....	136
<b>Figure 6.1:</b> DEVS Transparency and model interoperability using DEVSML .....	143

## LIST OF ILLUSTRATIONS - *CONTINUED*

<b>Figure 6.2:</b> Operations leading to model composability using DEVSMML.....	145
<b>Figure 6.3:</b> an SOA object capable of DEVS modeling .....	148
<b>Figure 6.4:</b> Automated XML snippet for a DEVS atomic model. ....	149
<b>Figure 6.5:</b> DEVS atomic DTD .....	151
<b>Figure 6.6:</b> DEVS coupled DTD .....	151
<b>Figure 6.7:</b> Web service Architecture for DEVSMML Implementation .....	153
<b>Figure 6.8:</b> Client side implementation using interfaces. ....	155
<b>Figure 6.9:</b> DEVS/SOA distributed architecture.....	158
<b>Figure 6.10:</b> Hierarchical simulator assignment for a hierarchical model.....	163
<b>Figure 6.11:</b> Hierarchical simulator assignment with Digraph2Atomic adapter .....	163
<b>Figure 6.12:</b> Communication among services.....	165
<b>Figure 6.13:</b> Execution of DEVS SOA-Based M&S .....	166
<b>Figure 6.14:</b> Server’s package structure for DEVS SOA .....	167
<b>Figure 6.15:</b> Adapter package containing Digraph to Atomic adapters .....	168
<b>Figure 6.16:</b> devsmml Modeling package for DEVS SOA .....	168
<b>Figure 6.17:</b> simulation package in DEVS SOA.....	169
<b>Figure 6.18:</b> Service package in DEVS SOA .....	170
<b>Figure 6.19:</b> Proxy package in DEVS SOA.....	171
<b>Figure 6.20:</b> DEVSMML implementation over SOADEVSS.....	172
<b>Figure 6.21:</b> DEVSMML and SOADEVSS integrated.....	173
<b>Figure 7.1:</b> Bifurcated Model-Continuity based System Life-cycle Process.....	177
<b>Figure 7.2:</b> Netcentric collaboration and execution using DEVSMML and SOADEVSS ..	179
<b>Figure 7.3:</b> Client application snapshot implemented as an applet. ....	181
<b>Figure 7.5:</b> GUI snapshot of SOADEVSS client hosting distributed simulation.....	184
<b>Figure 7.6:</b> Server Assignment to Models .....	185
<b>Figure 7.7:</b> The Complete DEVS Unified Process.....	186
<b>Figure 8.1:</b> JCAS Operational Scenario.....	191
<b>Figure 8.2:</b> Coupled scenario for JCAS model .....	192
<b>Figure 8.3:</b> DEVS Execution of JCAS model on console .....	193
<b>Figure 8.4:</b> JCAS BPMN scenario description .....	194
<b>Figure 8.5:</b> Snapshot of a BPMN-to-DEVS Transformation tool.....	195
<b>Figure 8.6:</b> Message-based Restricted NLP description of JCAS scenario .....	197
<b>Figure 8.7:</b> State-based specification of model CAOC.....	198
<b>Figure 8.8:</b> State-machine for CAOC Observer.....	199
<b>Figure 8.9:</b> SOADEVSS client running the JCAS model using Simulation services .....	200
<b>Figure 8.10:</b> Simulation output at client’s application using SOADEVSS client .....	201
<b>Figure 8.11:</b> OV-5 diagram for “select contractor” in IDEF0 notation .....	203
<b>Figure 8.12:</b> OV-6a diagram for “select contractor” in IDEF3 notation .....	203
<b>Figure 8.13:</b> Pseudo Code as per Zinn’s interpretation and integration procedure.....	203
<b>Figure 8.14:</b> Activity Report Model for Activity 6 generated thru Popkin SA .....	204

## LIST OF ILLUSTRATIONS - *CONTINUED*

<b>Figure 8.15:</b> IDEF3 representation of Activity 6 (“Conduct Dynamic Assessment of Target” TCT 2005 Architecture, 2003: OV-6a) [Zin04] .....	205
<b>Figure 8.16:</b> Pseudocode for Activity 6 – based on IDEF3 diagram .....	206
<b>Figure 8.17:</b> DEVS interrelationships of Activity 6 with other Activities. ....	210
<b>Figure 8.18:</b> DEVS description of Activity 6 in relation to Table 6 components.....	211
<b>Figure 8.19:</b> Automated Testing.....	219
<b>Figure 8.20:</b> Auto Correlation Sequential Diagram .....	220
<b>Figure 8.21:</b> Minimal Testable I/O pairs for Auto Correlation.....	220
<b>Figure 8.22:</b> Test Drivers Setup Diagram .....	221
<b>Figure 8.23:</b> Test Model Test Driver successful Auto Correlation scenario .....	222
<b>Figure 8.24:</b> SUT Test Driver successful Auto Correlation scenario.....	222
<b>Figure 8.25:</b> Geographic locations of fixed stations.....	224
<b>Figure 8.26:</b> Communication flow diagram for SCOPE command .....	224
<b>Figure 8.27:</b> System entity structure for SCOPE command system showing the fixed and mobile (aircraft) stations .....	227
<b>Figure 8.28:</b> GENETSCOPE simulation architecture for SCOPE command.....	228
<b>Figure 8.29:</b> DEVS M&S and the existing SCOPE command system.....	230
<b>Figure 8.30:</b> OV-5 for activity sounding .....	232
<b>Figure 8.31:</b> Simulation architecture for the SCOPE command network .....	235
<b>Figure 8.32:</b> Experimental frame for GENETSCOPE.....	237
<b>Figure 8.33:</b> Ground station configuration screen for Naval Air Station Sigonella.....	239
<b>Figure 8.34:</b> Mobile station configuration screen where the total count is bounded by the Experimental frame .....	240
<b>Figure 8.35:</b> Callsign entry for a mobile station .....	240
<b>Figure 8.36:</b> Flight path of mobile aircraft and other details .....	240
<b>Figure 8.37:</b> Experimental frame and ICEPAC data configuration .....	241
<b>Figure 8.38:</b> Run-time simulation visualization screen for rapid feedback.....	242
<b>Figure 10.1:</b> The Complete DEVS Unified Process.....	256

## LIST OF TABLES

<b>Table 3.1:</b> DEVS on addressing M&S issues.....	51
<b>Table 3.2:</b> Hierarchy of system specifications .....	52
<b>Table 4.1:</b> Tabular structure for State-based specifications .....	79
<b>Table 4.2:</b> State-based specifications for entity JTAC .....	80
<b>Table 4.3:</b> Mapping of Rules 1-8 to universal primitives in Universal State Machine (USM) .....	89
<b>Table 4.4:</b> Mapping of DoDAF with UML and DEVS M&S Elements.....	116
<b>Table 4.5:</b> Summarizing the contribution of OV-8, 9 to DEVS M&S .....	123
<b>Table 8.1:</b> Overview of DUNIP application in available case-studies .....	188
<b>Table 8.2:</b> State machine for component JTAC .....	191
<b>Table 8.3:</b> Activity-ID mapping for OV-8 and OV-9.....	207
<b>Table 8.4:</b> Sample OV-8 document .....	209
<b>Table 8.5:</b> Inner components within Operational Nodes and their mapping with ‘standardized’ DEVS models.....	214
<b>Table 8.6:</b> OV-9 description document mapping the Entity component inside Operational Node O1 with the Activity Components defined in OV-8 with port-interfaces .....	216
<b>Table 8.7:</b> Link 16 functionalities vs. Systems .....	223
<b>Table 8.8:</b> Activity 4ID mapping for OV-8 and OV-9 .....	231
<b>Table 8.9:</b> Sample OV-8 document .....	232
<b>Table 8.10:</b> Inner components within operational nodes and their mapping with “standardized” DEVS models .....	233
<b>Table 8.11:</b> Sample OV-9 Document .....	234
<b>Table 9.1:</b> Comparison of MDA and DUNIP .....	250